# Reducing Errors and Computational Load in Road Scene Text Recognition

T. R. Maximova[I], K. B. Bulatov[I, II]

[I] Smart Engines, Moscow, Russia
[II] Federal Research Center "Computer Science and Control", Russian Academy of Sciences, Moscow, Russia

**Abstract**. This paper focuses on the problem of reduction of the computation load for road scene text recognition by making a stopping decision which cuts off further recognition. The contribution of the paper is the construction of stopping rules for real-time text recognition systems with results combination, with an experimental evaluation on an open dataset RoadText-1k. We found that for fast-working systems the ROVER (Recognizer Output Voting Error Reduction) combination method and majority voting are best for Levenshtein and direct match metrics respectively, however, with an increase of per-frame processing time, ROVER becomes consistently better. Furthermore, while the selection of a single most focused frame is the worst strategy for fast-working systems, its comparative rank increases with the increase of processing time. Moreover, choosing one most focused frame and combining three most focused frames are preferable for fast-working systems when decreasing load on the device is needed.

**Keywords**: Combination method, Reducing computational load, Real-time recognition, Road scene analysis, Text recognition, Video stream recognition.

**DOI** 10.14357/20718632240301 **EDN** MMVTBM

## Introduction

Autonomous driving is a developing area of research. This technology is crucial since it can improve road safety and make driving more convenient and efficient [1]. Autonomous driving struggles to solve problems such as creating planning and control algorithms with particular regard to the urban setting [2], lane keeping [3], vehicle tracking [4] (it can be used to detect cars that are parked on streets [5]), estimation of the velocity of vehicles [6], traffic-sign detection and classification [7] etc. Convolutional neural networks and end-to-end models [3] are the most common methods that are used for solving most of these problems. There are more specific approaches to the mentioned problems such as multiple-object vehicle tracking systems by affinity matching using min-cost linear cost assignment [4] and dataset augmentation with synthetic traffic signs for rare traffic sign detection [8]. One of the vital problems in this area is the reduction of the load on the computational device. Since neural networks are widely used in self-driving vehicles [9], [10], it is important for computing devices that lack powerful processors to use smaller neural networks with negligible accuracy loss or computationally simplified neurons [11].

Autonomous navigation systems mostly use information from maps, sensory and visual feed [2, 3, 12] for route planning and safe navigation, while unplanned changes are very common on the road. These changes that a driver can learn from text warning boards might not be displayed on maps, but could be obtained, while analyzing video feed from an on-board camera, so text recognition is a very common research theme in this area [13]. The

recognition of such objects should be performed in real time (the best result could be available at any given time) that allows responsive decision making. In this paper we focus on road scene text objects.

In order to recognize an object, we have to detect it, track it, and apply some method which would obtain the best value prediction. We call "combination" a process of combining information obtained from multiple frames to produce a single prediction of the object value. For instance, one of such combination methods is majority voting [14], which selects the most frequent object value recognized in the set of processed frames. We need these processes in real time, so each processed frame increases a total load to the computational device. If during object tracking and combination a stopping decision can be made such that the average combination quality does not significantly decrease, the load would decrease.

The stopping problem for road scene text recognition was earlier discussed as a preliminary study in [15], where it was obtained that the general approach described in [16] works well for majority voting combination and ROVER-based (Recognizer Output Voting Error Reduction) combination [17], with majority voting being the best method of the two for maximizing mean recognition accuracy, but with ROVER-based method being better for mean Levenshtein distance minimization. The problem of real-time video stream recognition with stopping decision was introduced in [18] in scope of the identity document recognition problem, however, to the best knowledge of the authors it was never evaluated in the scope of road scene text recognition and reduction of the load on the computational device.

In this paper we are going to investigate a method of reducing the load on the computational device by making a dynamic decision when the video stream recognition should be stopped. If the stopping decision is made for the text object, it does not have to be recognized on subsequent frames, so the computational load can be reduced. Moreover, we are going to use recognition systems that do not recognize each obtained frame. The contributions of our paper are discovering whether it is worth applying the described combination methods, suggesting a working algorithm that solves stopping problems for these methods, analyzing in which conditions which method works best (in terms of quality degradation) and thus which method is preferable for our purposes.

## 1. Framework

In subsection 1-A we describe a basic model of the real-time text recognition process, in subsection 1-B we mention combination methods that we use in our experiments and in subsection 1-C we devise stopping rules corresponding to the recognition systems and their combination methods.

### A. General Model

In our paper we consider a text object recognition process with a given sequence of input frames. The recognition system S obtains a sequence of images $\{I_k\}_{k=1}^{n}$, where each image $I_k \in \bar{I}$ that denotes the set of all frames. The time of generating images is fixed and corresponds to a fixed frame rate of the device's camera. The time $t_0$ between registrations of two consecutive images $I_k$ and $I_{k+1}$ is constant for all k.

The purpose of processing a sequence of frames by the text object recognition system S is extracting the text object data. In our paper we assume that the time of the system S to process a single frame $I_k$ is constant, we denote it as $t_1$ and neglect the time of combination of per-frame recognition results. Assuming that generating and obtaining input frames are independent from the system S, the frame $I_k$ is instantly available for the system after its generation. We should note that only one instance of the recognition system S and its logical subsystem which manages acquisition of frames and processing strategy can work at any given time. Since we need to estimate the quality of the recognition result of the system S, we suppose that the final goal is recognizing a single text field of a text object.

The set of all possible text string recognition results are denoted as $\bar{X}$, and we assume that the text field of a text object that we need to recognize has a correct value $X^* \in \bar{X}$. If the system S processes an image $I_k$, it produces a text recognition result $S^{(1)}(I_k) \in \bar{X}$. If the system processes a sequence of images $I_1, I_2, \ldots, I_k$, the output is an accumulated text recognition result $S^{(k)}(I_1, I_2, \ldots, I_k) \in \bar{X}$ which is per-frame combined in some way. Furthermore, on X we define a metric function $\rho: \bar{X} \times \bar{X} \to R$, the value of which we interpret as recognition error.

We assume that the constructed process has the properties of anytime algorithms [26], such as being interruptible. It means that if at any moment we request an immediate interruption of the recognition process, the process stops and the result is returned as soon as possible. If the stopping decision is made at time t (by time t the system S has processed k frames), the loss can be expressed as follows:

$$L(t) = \rho(S(t), X^*) + c \cdot t, \quad (1)$$

where c > 0 is a cost of a unit of time in relation to the cost of the recognition error, and $S(t) = S^{(k)}(I_{j_1}, I_{j_2}, \ldots, I_{j_k})$ is the accumulated text recognition result during time t.

The aim of the paper is developing a stopping decision, corresponding to the parameters of the model $t_0$, $t_1$, c and to the per-frame combination method of the system S, in order to minimize the expected value of the loss function (1) when the process stops.

We assume that generally the recognition results become closer to the ground truths after the accumulation of several frames and that the rate of this improvement declines over time. An approximation of myopic stopping rule was proposed in [16]. The approximation was constructed calculating an expected distance to the next possible recognition result. Using expression of the loss function (1) considering the real-time model, we rewrite the approximation of a myopic stopping time as follows:

$$T_\Delta = min\{t \geq t_0 + t_1 : L(t) \leq E_t(L(t + \Delta t))\} = min\{t \geq t_0 + t_1 : \rho(X^*, S(t)) - E_t(\rho(X^*, S(t + \Delta t))) \leq c \cdot E_t(\Delta t)\} \leq min\{t \geq t_0 + t_1 : E_t(\rho(S(t), S(t + \Delta t))) \leq c \cdot E_t(\Delta t)\}, \quad (2)$$

where the condition $t \geq t_0 + t_1$ means that the process can stop after a minimum of one frame processed by the recognition system S, $E_t(\cdot)$ is a denotation of conditional expectation after time t and time t is a period between the current moment when we make the stopping decision and the next such moment.

With the assumption that the system S recognizes and accumulates each frame of the input sequence and after processing an image is finished the next frame in a sequence is instantly available, the ratio $t_1/t_0$ takes only integer values. It means that we can make the stopping decision after each new frame is processed and accumulated, so the time between two consecutive decisions is always equal to $t_1$. Thereby the approximation of a myopic stopping time (2) takes the following form:

$$T_\Delta = min\{t \geq t_0 + t_1 : E_t(\rho(S(t), S(t + t_1))) \leq c \cdot t_1\}. \quad (3)$$

This stopping rule is acceptable only if each input frame is fully processed by the combination method, so it means that the stopping rule (3) can be applied to per-frame combination methods such as ROVER [19]. The requirement is not so tough for combination methods in which one or several best results are selected using a criterion that is computed before processing frames [20].

Further we consider some criterion function $F : \bar{I} \to R$ that is defined on a set of images $\bar{I}$. We use this function in order to estimate the quality of the frame we are going to process. Moreover, we assume that the value $F(I_k)$ and the frame $I_k$ are observed simultaneously. Thus, now we consider the system S that recognizes a single frame (or several frames) that has the maximal value of the criterion function:

$$S^{(k)}(I_1, I_2, \ldots, I_k) = S^{(1)}(argmax_j F(I_j)). \quad (4)$$

In [21] it was shown that it is correct for such recognition systems which combination method chooses one best frame (or several best frames) to apply the same approximation (2). In this case we need to estimate the probability that the value of the criterion function F on the next step will become the maximum. However, we should note that for such systems if the value of the function F on the current step does not become greater than the current maximum, the system skips the current frame since it will not make any positive contribution to the result but it can only make the recognition result worse.

With this recognition system we need to change the application of the approximation of the stopping rule (2). We consider the probability P(t) at time $t \geq t_0 + t_1$ that the next frame $I(t + \Delta t)$ will have the value of the criterion function greater than the current maximum, so the approximation will be rewritten as:

$$T_\Delta = \min\{t \geq t_0 + t_1 : P(t) \cdot E_t(\rho(S(t), S^{(1)}(I(t + \Delta t)))) \leq c \cdot (t_1 \cdot P(t) + t_0 \cdot (1 - P(t)))\}, \quad (5)$$

where $E_t(\rho(S(t), S^{(1)}(I(t + \Delta t))))$ is a conditional expectation of the distance between the current

recognition result of the frame that has the greatest value F(I(t)) and the result at time $t + \Delta t$, in conditions of updating the maximum, (the best frame will certainly be $I(t + \Delta t)$). We estimate the probability P(t) by approximating the values of F of frames that have already been observed with some distribution, after that we use its distribution function to calculate the probability that a new value will become greater than the current maximum.

Although the methods of selecting a single (or several) best frame were compared to combining all frames in [20], a uniform frame processing scale was used there. Combination methods that choose one or several best frames can skip frames with less values of criterion function F, so these methods might effectively process more frames with the same amount of time. The comparison will be more correct if different values of the time between registration of two consecutive frames $t_0$ and the processing time $t_1$ will be taken into account.

### B. Combination Methods

There are many various approaches to solving the combination problem, for instance, selecting the best recognition result based on some criteria [22], applying classified ensemble techniques integrated with the result modification model [23] and many more.

A majority voting procedure [14] is a simple way to accumulate information from a sequence of frames by selecting the most frequently occurring recognition result. Another notable combination method is ROVER [19]. There are two steps in this method, firstly all input strings are aligned to determine the corresponding characters, and secondly the result is determined from the population of corresponding characters using some voting strategy, or using a summation of classification scores [17].

Furthermore, we use the combination system (4) in which the predictor value is the focus score. To calculate it we scale the frame $I_k$ so that its height is equal to h, then the frame is divided into squares with the side h. After that we calculate the focus score [24] of each square and count the arithmetic mean of these values. Since choosing one most focused frame (or several best frames) can skip frames that have less values of the criterion functions, these methods result in less load on the computational system.

### C. Stopping Rules

Using the theory from subsection 1-A, we rewrite the stopping rule in a general case:

$$N_\Delta = min\{n > 0 : \Delta_n \le c\},$$

where $\Delta_n = E_t(\rho(S(n), S(n + \Delta t))) \,/\, E_t(\Delta t)$. In order to calculate the expected distance between the current and the next result, we use an approach that was suggested in [16].

For ROVER method in direct match metric and for majority voting method in both metrics we calculate the approximation of $\Delta_n$ using the modification of the stopping rule, described in [15], [25]:

$$\Delta_n \approx \frac{1}{(n+1)\cdot t_1}\left(\sum_{k=1}^{n} \rho(X_n, S^{(n+1)}(I_1, \dots, I_n, I_k)) \; + \; \delta\right), \tag{6}$$

where $X_n$ is the result of the combination at the current step n and $\delta$ is a customizable parameter. The difference from the stopping rule in [15] is the division by $t_1$.

For the combination method in which we choose the most focused frame in both metrics we are in conditions of the approximation of the myopic stopping rule with predictor value (5):

$$\Delta_n \approx \frac{P_n\left(\sum_{k=1}^{m} \rho\left(X_n, S^{(1)}(I_{j_k})\right)\right) + \delta}{(n+1)\cdot(P_n \cdot t_1 + (1-P_n)\cdot t_0)}, \tag{7}$$

where $P_n$ is a probability that the focus of the next frame will become greater than the focus of the current best frame, $S^{(1)}(I_{j_k})$ is the result of recognition of the most focused frame at some step $j_k$.

To calculate the function $\Delta_n$ for the method of combining three most focused frames we also use the approximation (5), in this case $Q_n$ is a probability that the value of a focus of the next frame will become greater than the least of the three greatest values of the focuses:

$$\Delta_n \approx \frac{Q_n\left(\sum_{k=1}^{m} \rho\left(X_n, X_{j_k}\right)\right) + \delta}{(n+1)\cdot(Q_n \cdot t_1 + (1-Q_n)\cdot t_0)}. \tag{8}$$

In the rule $X_n$ is a result of the combination of three most focused frames and $X_{j_k}$ is the result of a combination of three most focused frames on some step $j_k$.

In our paper we discuss applying the described stopping rules, the most efficient methods and reducing the load on the computational device with the smallest degradation of the recognition accuracy.

## 2. Experiments

In order to evaluate the methods of text recognition results combinations and stopping methods for the task of road scene text recognition we used an open dataset RoadText-1K [14], which contains 1000 video clips captured from moving vehicles. Each frame of the dataset is annotated with the coordinates of bounding boxes and transcription ground truth of each text object.

To evaluate the methods let us investigate how each method behaves in relation to the others and to recognition without any combination. We are going to demonstrate it with a table of the average distance to the correct answer across all tracks in relation to the number of frames of all tracks on a given step. Besides, we are going to compare stopping rules (6), (7), (8) for combination methods with trivial stopping rules (the process stops after a fixed number of frames). The experiment shows that stopping rules are appropriate as they should on average show better results than the trivial rules. Furthermore, we are going to analyze how the increase of the processing time influences the results of recognition.

In our experiments we used two metrics. The first one is the direct match metric $\rho_D$, which has a value 0 for two identical recognition results, and a value 1 for different results. The second metric function that we are going to apply is a generalized Levenshtein distance, which measures the minimal number of elementary edit operations (substitutions, insertions, and deletions) required to transform one sequence of characters into another [27].

We used the following values for the parameters of the methods: the weight of an empty character for the ROVER algorithm [17] was taken to be 0.85. For choosing most focused frames we used the height h = 27 pixels that is the closest value to the average meaning for all frames 26.8. The value of the parameter $\delta$ for the Levenshtein metric we define as 1 and for the direct match metric – 0.5. The time of $t_0$ in rules (6), (7), (8) we consider equal to 0.03s in all the experiments since we process frames from video clips from the dataset [14] and not the video clips themselves. Each video clip lasts 10s and in each video clip there are approximately 300 frames so the approximate time between registrations of the images $I_k$ and $I_{k+1}$ is equal to 0.03s. The processing time of the recognition system $t_1$ is a constant value that is set depending on the experiment. The plots for $t_1 = 0.03$s, 0.12s, 0.3s are shown in Fig. 1, 2 and 3.

To compare combination methods we present Table 1 of mean distances to the correct answer in relation to the number of frames of all tracks at a given step. As a reference, the table lists the metric values for the recognition results without combination. In the direct match metric majority voting shows the best results while in Levenshtein distance the ROVER method is significantly better than other methods. In both metrics the method of combining three most focused frames does not perform as well as majority voting or ROVER, however, it achieves better mean distance values than taking the most focused frame.

In Fig. 1 we applied the devised stopping rules (6) – (8) on RoadText-1K [14], checked whether they are appropriate and compared combination methods. In the experiment we set the value $t_1 = 0.03$s. In order to check whether the stopping rules are appropriate we plotted with dotted lines trivial rules $N_K$ for each combination method (the recognition stops after exactly K frames). The plot of the stopping rule cannot be higher than the plot of the trivial rule. Considering the rule $N_K$ we made a decision whether we should use the devised stopping rule or not. In Fig. 1 all plots of combination methods are lower than the trivial rules. In direct match metric majority voting shows the best results. However, in Levenstein metric this method is higher than ROVER and even higher than combining three most focused frames.

Fig. 2 and 3 are the same plots as in Fig. 1 but with a higher processing time. They show how the average distance to the correct answer changes depending on the number of skipped frames. With the increase of the processing time the results of majority voting deteriorate, ROVER becomes the best method in both metrics and choosing three most focused frames becomes comparable to the full combination method.

Table 2 demonstrates the change of the mean distance to the ground truth for a fixed stopping time with an increase of the processing time. According to the Table 2 in Levenshtein metric in all cases ROVER produces results closer to the correct answer, although with the increase of processing time the method of choosing three most focused frames becomes comparable with ROVER. In direct match metric the method of choosing three most focused frames shows the best results for $t_1 = 0.09$s and 0.12s.

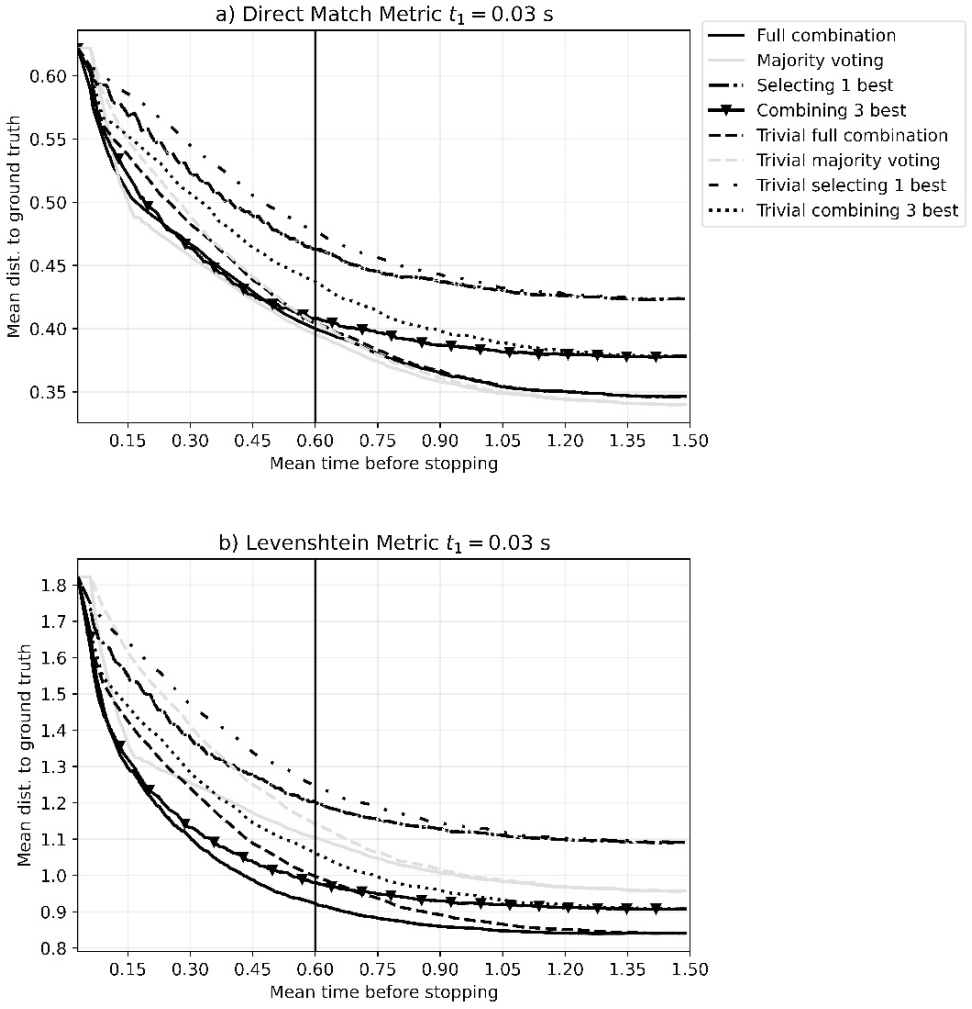


Fig. 1. Expected performance profiles for the analyzed combination methods and stopping rules. Time of frame processing and metrics differs for subplots: a) Direct match metric, $t_1 = 0.03$s, b) Levenshtein metric, $t_1 = 0.03$s. Lower is better. Markers do not show all the points, only some of them in order to simplify the representation

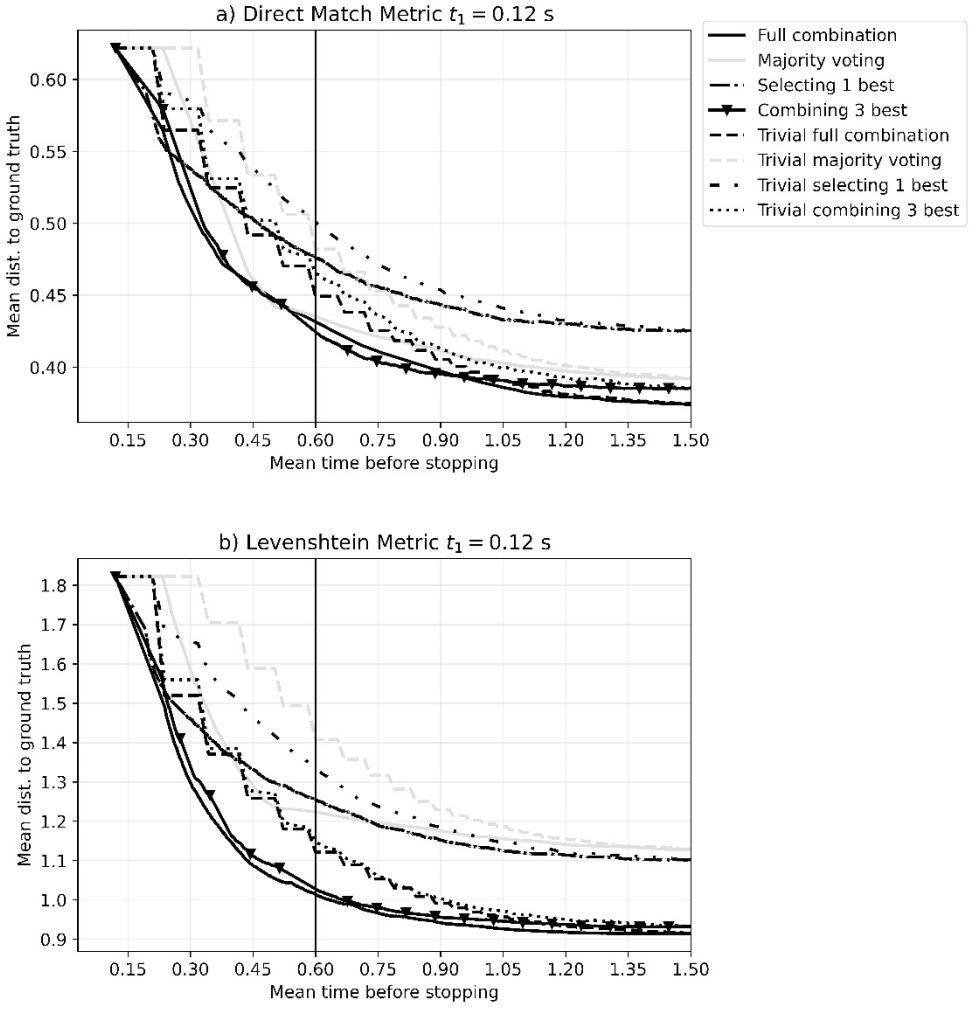


Fig. 2. Expected performance profiles for the analyzed combination methods and stopping rules.
Time of frame processing and metrics differs for subplots: a) Direct match metric, $t_1 = 0.12$s, b) Levenshtein metric, $t_1 = 0.12$s. Lower is better. Markers do not show all the points, only some of them in order to simplify the representation.

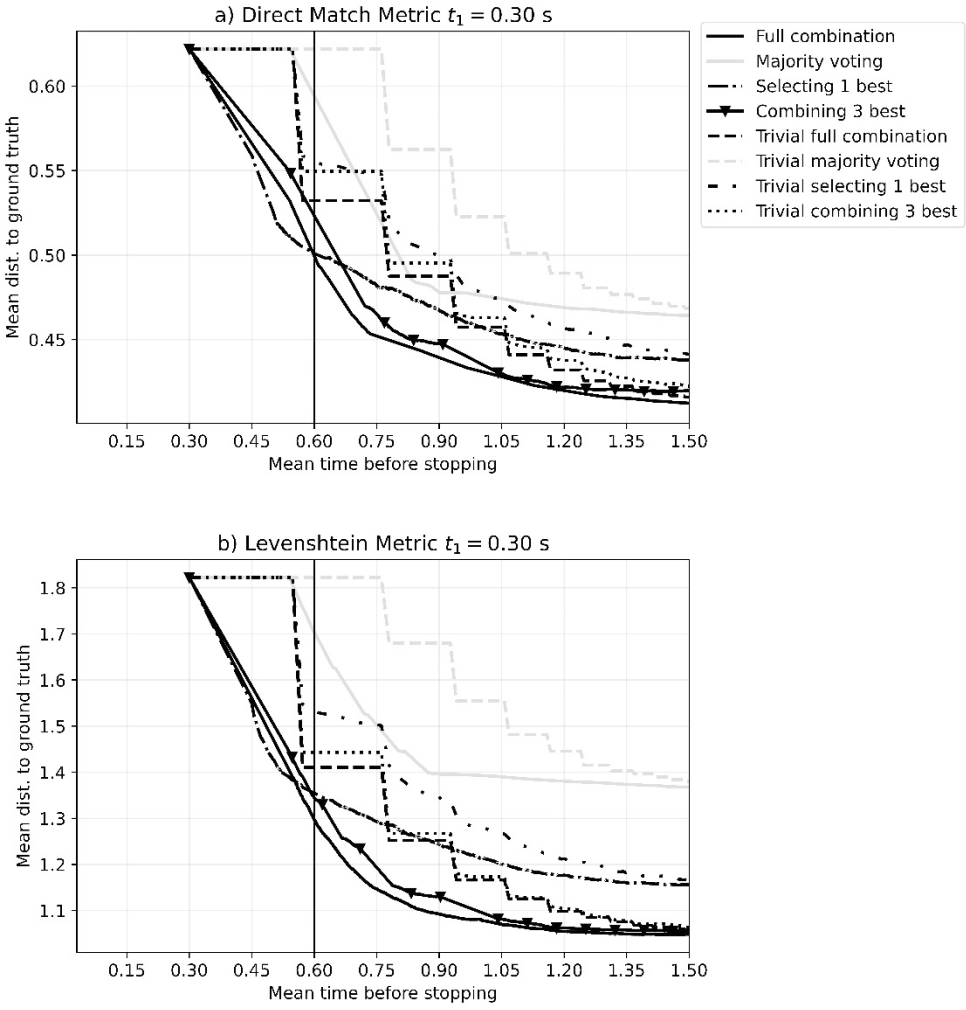


Fig. 3. Expected performance profiles for the analyzed combination methods and stopping rules.
Time of frame processing and metrics differs for subplots: a) Direct match metric, $t_1 = 0.30$s, b) Levenshtein metric, $t_1 = 0.30$s. Lower is better. Markers do not show all the points, only some of them in order to simplify the representation.

Table 1. The mean distances to the ground truth across all tracks in relation to the number of frames of all tracks at a given step. Lower is better

| | Direct Match Metric | | | | | Levenshtein Metric | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| Frames | ROVER | Majority Voting | Most Focused | Three Most Focused | No combination | ROVER | Majority Voting | Most Focused | Three Most Focused | No combination |
| 50 | 0.400 | **0.395** | 0.480 | 0.441 | 0.473 | **0.982** | 1.079 | 1.273 | 1.080 | 1.324 |
| 100 | 0.388 | **0.378** | 0.456 | 0.414 | 0.456 | **1.000** | 1.080 | 1.208 | 1.208 | 1.331 |
| 150 | 0.397 | **0.394** | 0.460 | 0.427 | 0.482 | **0.972** | 1.070 | 1.198 | 1.055 | 1.373 |
| 200 | 0.423 | **0.413** | 0.485 | 0.457 | 0.483 | **1.143** | 1.239 | 1.355 | 1.228 | 1.425 |
| 250 | 0.418 | **0.417** | 0.489 | 0.455 | 0.501 | **1.055** | 1.170 | 1.327 | 1.153 | 1.471 |
| 300 | 0.421 | **0.405** | 0.496 | 0.460 | 0.471 | **1.103** | 1.141 | 1.318 | 1.138 | 1.332 |

## 3. Discussion

Table 1 demonstrates that in direct match metric one most focused frame is comparable to recognition without a combination. However, in Levenshtein distance the results are better as the values are consistently smaller. Moreover, the combination of the three most focused frames in Levenshtein metric shows results similar to the results of majority voting while in direct match metric the results are much worse. We should note that the best results in Levenshtein metric are achieved using ROVER while in direct match metric it is worse than majority voting.

Analyzing Fig. 1 we concluded that the stopping rules are appropriate as their plots are lower than the plots of trivial rules. In Fig. 1 (b) ROVER gives the best results while in Fig. 1 (a) majority voting is the best. The most focused frame is the worst in both metrics and combining three most focused frames in Levenshtein metric is lower than majority voting but it is higher than the method of full combination.

Fig. 2 and 3 demonstrate that with the increase of processing time majority voting becomes worse than other methods in both metrics. With $t_1 = 0.30$s the plot of majority voting is noticeably higher than other methods, especially in Levenshtein metric (see Fig. 3 (b)). While in Fig. 1 combining three most focused frames is higher than ROVER, in Fig. 2 and 3 it is comparable to ROVER. In addition, Fig. 3 confirms the results of the experiments in [18]. With the increase of processing time, one most focused frame becomes better than other methods on the first steps of recognition.

Analyzing Table 2 we observed that the results of the experiments differ from the ones in the report [18]. In [18] choosing the most focused frame in several cases gives the best results while in our paper this method does not present any advantages over the others in most cases. We should note that in the report [18] in Levenshtein metric the method of combining three best frames gives best results in more than a half of cases. In our experiments in the case of Levenshtein distance this method is comparable to ROVER but the latter method is better. The best method for Levenshtein metric and the one for the Direct Match Metric differ strongly. We assume that it happens since Direct Match Metric unites all the wrong results in one equivalence class, while in Levenshtein metric different results have different values due to their "closeness" to the ground truth. That might mean that the combination of three most focused frames gives more correct results but when the result is wrong, it might differ from the ground truth a lot.

Thus, analyzing the experiments we concluded that the best results are achieved with ROVER and majority voting for fast-working systems. However, when the processing time increases, majority voting becomes significantly worse, so if we need to reduce the load on a computational device, we should apply selecting one most focused frame or combining three most focused frames for fast-working systems.

We should pay attention to some points that influence our experiments. Firstly, calculating the focus score we used scaling and division into squares as the value of focus depends on the size of the frame. If we do not apply them, the results of the

Table 2. The example of achieved mean distance from the combined result at stopping time to the ground truth for the stopping rules configured to yield mean stopping time E(T) = 0.6s. Lower is better

| | Direct Match Metric | | | | Levenshtein Metric | | | |
|---|---|---|---|---|---|---|---|---|
| $t_1$ | ROVER | Majority Voting | Most Focused | Three Most Focused | ROVER | Majority Voting | Most Focused | Three Most Focused |
| 0.03 | 0.401 | **0.396** | 0.463 | 0.408 | **0.922** | 1.105 | 1.202 | 0.980 |
| 0.06 | 0.410 | 0.415 | 0.462 | **0.407** | **0.948** | 1.157 | 1.207 | 0.975 |
| 0.09 | 0.423 | 0.423 | 0.468 | **0.410** | **0.987** | 1.197 | 1.225 | 0.995 |
| 0.12 | 0.432 | 0.435 | 0.476 | **0.421** | **1.014** | 1.224 | 1.256 | 1.023 |
| 0.15 | **0.434** | 0.444 | 0.481 | 0.438 | **1.046** | 1.258 | 1.264 | 1.082 |
| 0.18 | **0.444** | 0.462 | 0.489 | 0.446 | **1.085** | 1.339 | 1.302 | 1.116 |
| 0.21 | **0.455** | 0.470 | 0.494 | 0.460 | **1.130** | 1.373 | 1.319 | 1.149 |
| 0.24 | **0.462** | 0.473 | 0.500 | 0.474 | **1.169** | 1.486 | 1.337 | 1.243 |
| 0.27 | 0.492 | 0.622 | 0.500 | **0.472** | **1.227** | 1.613 | 1.346 | 1.261 |
| 0.30 | **0.494** | 0.622 | 0.500 | 0.548 | **1.297** | 1.704 | 1.350 | 1.344 |

experiments change due to the features of the dataset. We used the arithmetic mean of the values of the squares but there can be different methods of calculation. Secondly, we assumed a normal distribution of focus scores in order to calculate the probability of their increase. It is unknown how different methods for calculations of the probability will influence the results.

## Conclusion and Future Work

The paper described the method to reduce the load on computational devices in road text recognition. The reduction is achieved by the early stopping of recognition when the stopping decision is made after enough per-frame results are accumulated. We devised the stopping rules, and compared different combination strategies to analyze the degradation of the combined result after early stopping.

The main contributions of this work are determining whether described combination methods are worth applying, which ones are best for reducing the load on the computational device, proposing algorithms for making stopping decisions and researching the behavior of methods depending on the recognition system processing time. We came to a conclusion that using stopping rules can decrease the load on the device significantly. Moreover, we understood that while majority voting is the best combination method for maximizing direct result string match accuracy for fast-working systems, with an increase of time required to process a single frame ROVER and combination of the three most focused frames becomes preferable for both direct match and Levenshtein metrics. It is clear that while the selection of the most focused frame is the worst strategy for fast-working systems, its comparative rank sharply increases with the increase of frame processing time. Finally, choosing one most focused frame and combining three most focused frames are preferable when decreasing load on the computational device is needed.

In future work different methods of combination and their modifications can be investigated. For instance, other calculations of focus scores can be suggested that could have an influence on the process of recognition. Furthermore, we suggested only some of the possible rules for making a stopping decision, so we can construct new rules or approximations for the described ones.

## References


1. Yuan-Ying Wang, Hung-Yu Wei, "Road Capacity and Throughput for Safe Driving Autonomous Vehicles", IEEE Access, 2020, vol. 8, pp. 95779–95792, 10.1109/ACCESS.2020.2995312.
2. Paden B., Čáp M., Zheng Yong S., Yershov D., Frazzoli E., "A survey of motion planning and control techniques for self-driving urban vehicles", IEEE Transactions on Intelligent Vehicles, vol. 1, 1998, pp. 33–55, 10.1109/TIV.2016.2578706.
3. Chen Z., Huang X., "End-to-end learning for lane keeping of self-driving cars", 2017 IEEE Intelligent Vehicles Symposium (IV), 2017, pp. 1856–1860, 10.1109/IVS.2017.7995975.
4. Gündüz G., Acarman A. T., "A Lightweight Online Multiple Object Vehicle Tracking Method", 2018 IEEE Intelligent Vehicles Symposium (IV), 2018, pp. 427–432, 10.1109/IVS.2018.8500386.
5. Matsuda A., Matsui T., Matsuda Y., Suwa H., Yasumoto K., "A System for Real-time On-street Parking Detection and Visualization on an Edge Device'", 2021 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops), 2021, pp. 227–232, 10.1109/PerComWorkshops51409.2021.9431076.
6. Balamuralidhar N., Tilon S., Nex F., , 2021. "MultEYE: Monitoring system for real-time vehicle detection, tracking and speed estimation from UAV imagery on edge-computing platforms", Remote sensing, 2021, 13(4), p.573, 10.3390/rs13040573.
7. Zhu Z., Liang D., Zhang S., Huang X., Li B., Shimin Hu, "Traffic-Sign Detection and Classification in the Wild", IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, 10.1109/CVPR.2016.232.
8. Konushin A.S., Faizov B.V., Shakhuro V.I., "Road images augmentation with synthetic traffic signs using neural networks", Computer Optics, vol. 45, 2021, pp. 736–748, 10.18287/2412-6179-CO-859.
9. Rajesh R., Rajeev K., Suchithra K., Lekhesh V.P., Gopakumar V., Ragesh N.K., "Coherence vector of Oriented Gradients for traffic sign recognition using Neural Networks", The 2011 International Joint Conference on Neural Networks, 2011, 10.1109/IJCNN.2011.6033318.
10. Lobanov M., Sholomov D., "On the Acceleration of the Convolutional Neural Network Architecture Based on ResNet in the Task of Road Scene Objects Recognition", Journal of Information Technologies and Computing Systems, 2019, vol. 69, pp. 57–65.
11. Limonova E. E., Alfonso D. M., Nikolaev D. P., Arlazarov V. V., "Bipolar Morphological Neural Networks: Gate-Efficient Architecture for Computer Vision", IEEE Access, vol. 9, pp. 97569–97581, 2021, doi: 10.1109/ACCESS.2021.3094484.
12. Bojarski M., Testa D., Dworakowski D., Firner B., Flepp B., Goyal P., Jackel L. D., Monfort M., Muller U., Zhang J., Zhang X., Zhao J., Zieba K., "End to end learning for self-driving cars", Retrieved from https://arxiv.org/abs/1604.07316, 2016, Accessed August 4, 2022.
13. Naiemi F., Ghods V., Khalesi H., "Scene text detection and recognition: a survey", Multimedia Tools and Applications, 2022, vol. 81, 10.1007/s11042-022-12693-7.

14. Reddy S., Mathew M., Gomez L., Rusinol M., Karatzas D., Jawahar C.V., "RoadText-1K: Text Detection amp; Recognition Dataset for Driving Videos", 2020 IEEE International Conference on Robotics and Automation (ICRA), pp. 11074–11080, 10.1109/ICRA40945.2020.9196577.
15. Bulatov K., Fedotova N., Arlazarov V. V., "An approach to road scene text recognition with per-frame accumulation and dynamic stopping decision", Thirteenth International Conference on Machine Vision, 2021, 10.1117/12.2586912.
16. Bulatov K., Razumnyi N., Arlazarov V.V., "On optimal stopping strategies for text recognition in a video stream as an application of a monotone sequential decision model", Int. J. on Document Analysis and Recognit, 2019, vol. 22, number 3, pp. 303–314, 10.1007/s10032-019-00333-0.
17. Bulatov K., "A method to reduce errors of string recognition based on combination of several recognition results with per-character alternatives", Bulletin of the South Ural State University. Ser. Mathematical Modelling, Programming & Computer Software, 2019, vol. 12, number 3, pp. 74–88, 10.14529/mmp190307.
18. Bulatov K., Arlazarov V. V., "Determining optimal frame processing strategies for real-time document recognition systems", Document Analysis and Recognition – ICDAR 2021, Lecture Notes in Computer Science, vol. 12822, 2021, 10.1007/978-3-030-86331-9_18.
19. Fiscus J. G., "A post-processing system to yield reduced word error rates: Recognizer Output Voting Error Reduction (ROVER)", 1997 IEEE Workshop on Automatic Speech Recognition and Understanding Proceedings, 1997, pp. 347–354, 10.1109/ASRU.1997.659110.
20. Petrova O., Bulatov K., Arlazarov V. L., "Methods of weighted combination for text field recognition in a video stream", Proc. SPIE (ICMV 2019), 2020, vol. 11433, pp. 704–709, 10.1117/12.2559378.
21. Tolstov I., Martynov S., Farsobina V., Bulatov K., "A modification of a stopping method for text recognition in a video stream with best frame selection", Proc. SPIE (ICMV 2020), 2021, vol. 11605, pp. 464–471, 10.1117/12.2586928.
22. Mita T., Hori O., "Improvement of Video Text Recognition by Character Selection", Proceedings of Sixth International Conference on Document Analysis and Recognition, 2001, pp. 1089–1093, 10.1109/ICDAR.2001.953954.
23. Czúni L., Nagy A. M., "Improving object recognition of CNNs with multiple queries and HMMs", Twelfth International Conference on Machine Vision (ICMV 2019), 2020, vol. 11433, pp. 266–272, 10.1117/12.2559393.
24. Bulatov K. B., Polevoy D. V., "Reducing overconfidence in neural networks by dynamic variation of recognizer relevance", ECMS, 2015, pp. 488–491.
25. Bulatov K., Fedotova N., Arlazarov V. V., "Fast Approximate Modelling of the Next Combination Result for Stopping the Text Recognition in a Video", 2020 25th International Conference on Pattern Recognition (ICPR), 2021, pp. 239–246, 10.1109/ICPR48806.2021.9412574.
26. Zilberstein S., "Using Anytime Algorithms in Intelligent Systems", AI Magazine, 1996, vol. 17, number 3, pp. 73–83, 0.1609/aimag.v17i3.1232.
27. Yujian L., Bo L., "A Normalized Levenshtein Distance Metric", IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007, vol. 29, number 6, pp. 1091–1095, 10.1109/TPAMI.2007.1078.

**Maximova Taisiya R.** Smart Engines, Moscow, Russia. Programmer. Topics of interest: real-time recognition, text recognition, video stream recognition and Gaussian measures. E-mail: t.maksimova@smartengines.com

**Bulatov Konstantin B.** PhD, Federal Research Center "Computer Science and Control" of Russian Academy of Sciences, 44/2 Vavilova str. Moscow, 119333, Russia. E-mail: kbulatov@smartengines.com.

## Снижение ошибки и вычислительной нагрузки в распознавании текста дорожной сцены

Т. Р. Максимова[I], К. Б. Булатов[I, II]

[I] Smart Engines, Москва, Россия
[II] Федеральный исследовательский центр "Информатика и управление" РАН, Москва, Россия

**Аннотация.** Статья посвящена проблеме снижения вычислительной нагрузки для распознавания текста дорожной сцены принятием решения об остановке, прекращающем дальнейшее распознавание. Описывается построение правил остановки для систем распознавания текста в реальном времени с комбинацией результатов и экспериментальной оценкой на открытом наборе данных RoadText-1k. Обнаружено, что для быстродействующих систем метод комбинации ROVER (Recognizer Output Voting Error Reduction) и голосование являются наилучшими для метрики Левенштейна и дискретной метрики соответственно, однако с увеличением времени обработки

каждого кадра ROVER становится стабильно лучше. Хотя выбор наиболее сфокусированного кадра является худшей стратегией для быстродействующих систем, ее сравнительный рейтинг повышается с увеличением времени обработки. Важно отметить, что выбор наиболее сфокусированного кадра и объединение трех наиболее сфокусированных кадров предпочтительнее для быстродействующих систем, когда требуется снизить нагрузку.

**Ключевые слова:** метод комбинации, уменьшение вычислительной нагрузки, распознавание в реальном времени, анализ дорожной сцены, распознавание текста, распознавание видеопотока.

**DOI** 10.14357/20718632240301 **EDN** MMVTBM

## Литература

1. Yuan-Ying Wang, Hung-Yu Wei, "Road Capacity and Throughput for Safe Driving Autonomous Vehicles", IEEE Access, 2020, vol. 8, pp. 95779–95792, 10.1109/ACCESS.2020.2995312.
2. Paden B., Čáp M., Zheng Yong S., Yershov D., Frazzoli E., "A survey of motion planning and control techniques for self-driving urban vehicles", IEEE Transactions on Intelligent Vehicles, vol. 1, 1998, pp. 33–55, 10.1109/TIV.2016.2578706.
3. Chen Z., Huang X., "End-to-end learning for lane keeping of self-driving cars", 2017 IEEE Intelligent Vehicles Symposium (IV), 2017, pp. 1856–1860, 10.1109/IVS.2017.7995975.
4. Gündüz G., Acarman A. T., "A Lightweight Online Multiple Object Vehicle Tracking Method", 2018 IEEE Intelligent Vehicles Symposium (IV), 2018, pp. 427–432, 10.1109/IVS.2018.8500386.
5. Matsuda A., Matsui T., Matsuda Y., Suwa H., Yasumoto K., "A System for Real-time On-street Parking Detection and Visualization on an Edge Device'", 2021 IEEE International Conference on Pervasive Computing and Communications Workshops and other Affiliated Events (PerCom Workshops), 2021, pp. 227–232, 10.1109/PerComWorkshops51409.2021.9431076.
6. Balamuralidhar N., Tilon S., Nex F., , 2021. "MultEYE: Monitoring system for real-time vehicle detection, tracking and speed estimation from UAV imagery on edge-computing platforms", Remote sensing, 2021, 13(4), p. 573, 10.3390/rs13040573.
7. Zhu Z., Liang D., Zhang S., Huang X., Li B., Shimin Hu, "Traffic-Sign Detection and Classification in the Wild", IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016, 10.1109/CVPR.2016.232.
8. Konushin A.S., Faizov B.V., Shakhuro V.I., "Road images augmentation with synthetic traffic signs using neural networks", Computer Optics, vol. 45, 2021, pp. 736–748, 10.18287/2412-6179-CO-859.
9. Rajesh R., Rajeev K., Suchithra K., Lekhesh V.P., Gopakumar V., Ragesh N.K., "Coherence vector of Oriented Gradients for traffic sign recognition using Neural Networks", The 2011 International Joint Conference on Neural Networks, 2011, 10.1109/IJCNN.2011.6033318.
10. Лобанов М. Г., Шоломов Д. Л, "Об ускорении архитектуры сверточной нейронной сети на базе ResNet в задаче распознавания объектов дорожной сцены", Информационные Технологии и Вычислительные Системы, 2019, выпуск 3, с 57-65, 10.14357/20718632190305.
11. Limonova E. E., Alfonso D. M., Nikolaev D. P., Arlazarov V. V., "Bipolar Morphological Neural Networks: Gate-Efficient Architecture for Computer Vision", IEEE Access, vol. 9, pp. 97569–97581, 2021, doi: 10.1109/ACCESS.2021.3094484.
12. Bojarski M., Testa D., Dworakowski D., Firner B., Flepp B., Goyal P., Jackel L. D., Monfort M., Muller U., Zhang J., Zhang X., Zhao J., Zieba K., "End to end learning for self-driving cars", Retrieved from https://arxiv.org/abs/1604.07316, 2016, Accessed August 4, 2022.
13. Naiemi F., Ghods V., Khalesi H., "Scene text detection and recognition: a survey", Multimedia Tools and Applications, 2022, vol. 81, 10.1007/s11042-022-12693-7.
14. Reddy S., Mathew M., Gomez L., Rusinol M., Karatzas D., Jawahar C.V., "RoadText-1K: Text Detection amp; Recognition Dataset for Driving Videos", 2020 IEEE International Conference on Robotics and Automation (ICRA), pp. 11074–11080, 10.1109/ICRA40945.2020.9196577.
15. Bulatov K., Fedotova N., Arlazarov V. V., "An approach to road scene text recognition with per-frame accumulation and dynamic stopping decision", Thirteenth International Conference on Machine Vision, 2021, 10.1117/12.2586912.
16. Bulatov K., Razumnyi N., Arlazarov V.V., "On optimal stopping strategies for text recognition in a video stream as an application of a monotone sequential decision model", Int. J. on Document Analysis and Recognit, 2019, vol. 22, number 3, pp. 303–314, 10.1007/s10032-019-00333-0.
17. Bulatov K., "A method to reduce errors of string recognition based on combination of several recognition results with per-character alternatives", Вестн. ЮУрГУ. Сер. Матем. моделирование и программирование, **12**:3 (2019), 74–88, 10.14529/mmp190307.
18. Bulatov K., Arlazarov V. V., "Determining optimal frame processing strategies for real-time document recognition systems", Document Analysis and Recognition – ICDAR 2021, Lecture Notes in Computer Science, vol. 12822, 2021, 10.1007/978-3-030-86331-9_18.
19. Fiscus J. G., "A post-processing system to yield reduced word error rates: Recognizer Output Voting Error Reduction (ROVER)", 1997 IEEE Workshop on Automatic Speech Recognition and Understanding Proceedings, 1997, pp. 347–354, 10.1109/ASRU.1997.659110.
20. Petrova O., Bulatov K., Arlazarov V. L., "Methods of weighted combination for text field recognition in a video

stream", Proc. SPIE (ICMV 2019), 2020, vol. 11433, pp. 704–709, 10.1117/12.2559378.
21. Tolstov I., Martynov S., Farsobina V., Bulatov K., "A modification of a stopping method for text recognition in a video stream with best frame selection", Proc. SPIE (ICMV 2020), 2021, vol. 11605, pp. 464–471, 10.1117/12.2586928.
22. Mita T., Hori O., "Improvement of Video Text Recognition by Character Selection", Proceedings of Sixth International Conference on Document Analysis and Recognition, 2001, pp. 1089–1093, 10.1109/ICDAR.2001.953954.
23. Czúni L., Nagy A. M., "Improving object recognition of CNNs with multiple queries and HMMs", Twelfth International Conference on Machine Vision (ICMV 2019), 2020, vol. 11433, pp. 266–272, 10.1117/12.2559393.
24. Bulatov K. B., Polevoy D. V., "Reducing overconfidence in neural networks by dynamic variation of recognizer relevance", ECMS, 2015, pp. 488–491.
25. Bulatov K., Fedotova N., Arlazarov V. V., "Fast Approximate Modelling of the Next Combination Result for Stopping the Text Recognition in a Video", 2020 25th International Conference on Pattern Recognition (ICPR), 2021, pp. 239–246, 10.1109/ICPR48806.2021.9412574.
26. Zilberstein S., "Using Anytime Algorithms in Intelligent Systems", AI Magazine, 1996, vol. 17, number 3, pp. 73–83, 0.1609/aimag.v17i3.1232.
27. Yujian L., Bo L., "A Normalized Levenshtein Distance Metric", IEEE Transactions on Pattern Analysis and Machine Intelligence, 2007, vol. 29, number 6, pp. 1091–1095, 10.1109/TPAMI.2007.1078.

**Максимова Таисия Романовна.** Smart Engines, Москва, Россия. Программист. Научные интересы: распознавание в реальном времени, распознавание текста, распознавание в видеопотоке и гауссовские меры. E-mail: t.maksimova@smartengines.com

**Булатов Константин Булатович.** Федеральный исследовательский центр "Информатика и управление" Российской академии наук, Москва, Россия. Старший научный сотрудник, кандидат технических наук. Область научных интересов: комбинаторные алгоритмы, компьютерное зрение, анализ и распознавание документов. E-mail: kbulatov@smartengines.com