A new real-time method for finding temporary and permanent road marking and its applications

R.V. Dosaev[1], K.I. Kiy[1]

[1]Keldysh Institute of Applied Mathematics of RAS, Miusskaya square 4, Moscow, Russia, 145047

Abstract. In this paper, a new real-time method for finding temporary and permanent road marking is proposed. The method is based on the geometrized histograms method for segmenting and describing color images. It is able to work with both rectilinear and curvilinear marking, as well as with color temporary and permanent road marking. The developed method is stable under illumination and is able to work even for partially disappearing road marking, typical for late winter and early spring. The proposed method has been implemented by a program written in C++, operating under Windows and Linux. The program operation has been tested on video records shot on typical Russian roads. The processing speed is about 20 fps for a standard modern computer. The results of program operation are presented and discussed. The developed program is a part of the video system of the AvtoNiva pilotless vehicle.

1. Introduction

The first efficient methods for real-time lane (road marking) detection were proposed and implemented by E.D. Dickmanns and his colleagues [1, 2]. These methods were dated back to the late 1980s and employed the so-called 4D approach proposed by E.D. Dickmanns [1]. Many methods based on Hough transform, RANSAC algorithm, and learning approaches, including deep learning of convolutional networks, have been proposed (see [3–5] and the references in these papers) since then. In spite of serious progress in investigations on this topic, serious problems connected with lane marking disappearing in early spring and late winter, wet places and occlusion still take place. In spite of impressing results reached by application of convolutional networks, there are certain problems that arise in using them, such as overfitting, a large amount of training data required, the high computational cost leading to using expensive SIMD architectures. In this connection, it seems to be reasonable to apply the geometrized histograms method for segmentation and description of color images [6–9] for finding road marking, since it makes it possible to find both white road marking of arbitrary shape and colored road making (for example, temporary one). The geometrized histograms method is based on a method for approximate description of the value distribution of a scalar (intensity) or vector function (color) giving the image. This description is called a geometrized histogram. Geometrized histograms make it possible to find even very small connected objects in the image plane (with diameter up to three pixels) and construct from them continuous objects like lane marking. Geometrized histograms can distinguish objects with very small local contrasts and automatically adapt themselves to very small local differences in intensity and color for connected objects.

In this paper, we propose an image understanding system that can find road marking using the complete image segmentation in real time using only standard computational facilities even for images taken by cheap cameras like those employed in camcorders. In section 2, we remind briefly the main points of the geometrized histograms method and explain the main ideas for applying them in finding road markings. In section 3, we present the algorithms for finding road markings. Several examples of finding road markings, demonstrating the ability of the method, are presented and discussed.

2. A brief description of the geometrized histograms method for segmentation and description of color or grayscale images

A detailed description of the geometrized histograms method can be found in [6-9]. To construct the geometrized histogram for color or grayscale image, the image is divided into strips St_i, $i = 1, \ldots, n$, of the same small width W with sides parallel to the horizontal (or vertical) axis Os of the image plane. Let us show first how to construct the geometrized histogram for a scalar function $f(x, y)$, giving a grayscale image. The geometrized histogram describes approximately the level sets L_z of $f(x, y)$, i.e., the points of the strip St_i where $f(x, y) = z$. Since we deal with the discrete representation of the image array, the projection of L_z onto Os is a union of intervals (segments) I_{kz} on this axis $Pr(L_z) = \cup_k I_{kz}$. For each segment I_{kz}, its cardinality is the number of the points of the level set L_z in the strip St_i that are projected onto this interval. It is clear that the set of cardinalities of the intervals I_{kz} for all possible z determines the classical histogram of $f(x, y)$ in the strip St_i. The collection of intervals I_{kz} approximately describes L_z, since the set of level z belongs to the preimage of $\cup_k I_{kz}$, $L_z \subset Pr^{-1}(\cup_k I_{kz})$ and the strip is narrow. The union of I_{kz} for all z determines the space of intervals on Os with the scalar function of cardinality on them. Note that intervals I_{kz} may have a nonempty intersection on Os. This occurs when one object lies over another in the strip. The space of intervals $\cup_k^z I_{kz}$ is called the local geometrized histogram (**HG**$_i$) of $f(x, y)$ in St_i. An example of the geometrized histogram of the grayscale component of a strip in a color image of a road scene can be found in Fig. 1. The ***ox*** axis in the figure presents the value of the grayscale function, which varies from 0 to 63. The ***oy*** axis presents the coordinates of pixels along the image strip. The coordinate on this axis varies from 0 to ***DimX*** – 1, where ***DimX*** is the horizontal dimension of the image (we consider a horizontal strip). The geometrized histogram in the grayscale version of this strip describes the geometry of intensity distribution in the strip. In the intersection of the strip with the lane marking, one can see a local burst of intensity. This burst is selected by a red circle. This means that the geometrized histogram of the grayscale component can be employed as a road marking detector.

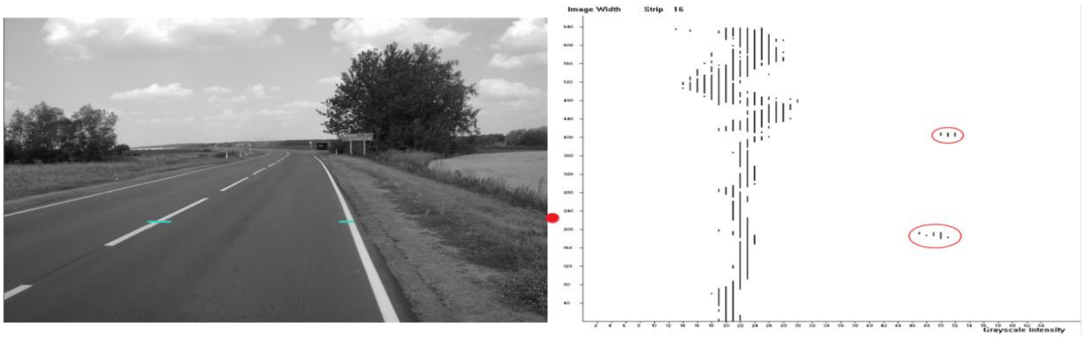

Figure 1. A road scene and the geometrized histogram of the grayscale component of the i-th horizontal strip.

However, not only white road marking is important for automatic driving vehicles. Yellow road marking is also of great interest. To detect yellow road marking, it is necessary to segment and describe color images in real-time. For this purpose, the construction of the geometrized histogram is generalized for vector functions representing color images. We deal with the representation of the color image by the function (*G/(G+B), G/(G+R), I*). Let us introduce a characteristic function *CF*. If the hue of the point belongs to the yellow part of the color triangle, then *CF* coincides with *G/(G+B)*. When passing to the next range (green, blue, red), the value of *G/(G+B)* is shifted by *M*, where *M* is the number of grades of the function *G/(G+B)*. The geometrized histogram of *CF*, supplemented for each interval I_{kz} by the classical histogram of the other color component *G/(G+R)*, is called the geometrized histogram of the color image in $\boldsymbol{St}_{\mathrm{i}}$. Using these data for each of its members I_{kz}, we determine the localization interval $\mathbf{Int}_{kz} = [beg_{kz}, end_{kz}]$ on **Os**, the range and the mean value of hue $\Delta_{\mathrm{H}}^{kz} = [H_{min}^{kz}, H_{max}^{kz}]$ and H_{mean}^{kz}, the range and the mean value of saturation $\Delta_{\mathrm{S}}^{kz} = [S_{min}^{kz}, S_{max}^{kz}]$ and S_{mean}^{kz}, and the range and the mean value of grayscale intensity $\Delta_{\mathrm{I}}^{kz} = [I_{min}^{kz}, I_{max}^{kz}]$ and I_{mean}^{kz} [6]. In addition, each interval of the geometrized histogram has the cardinality $Card^{kz}$. Figure 2 presents the geometrized histogram of a strip of a color image of a road scene. The ***ox*** axis in the figure presents the value of the characteristic function *CF*, which varies within its range of values. As in Fig. 1, the ***oy*** axis presents the coordinates of pixels along the image strip. The coordinate on this axis varies from 0 to ***DimX*** – 1, where ***DimX*** is the horizontal dimension of the image (we consider a horizontal strip). The geometrized histogram in the colored version of this strip describes the geometry of color distribution in the strip.

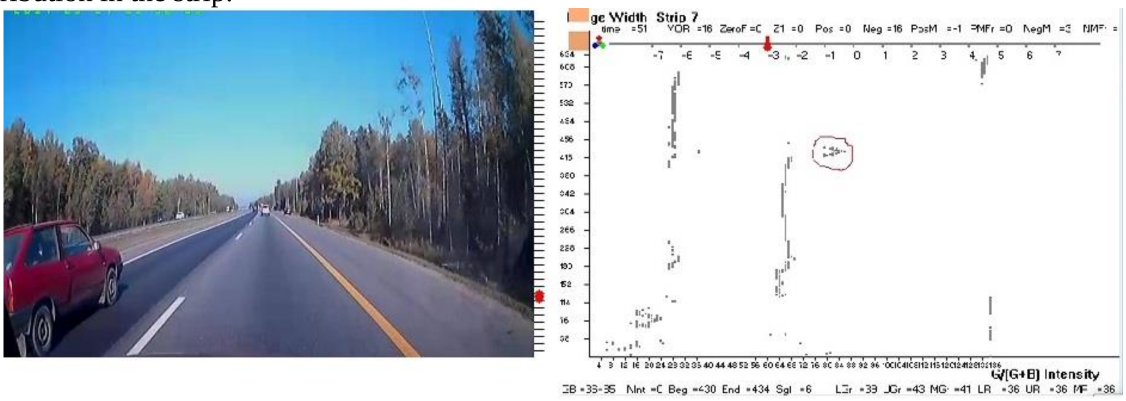

Figure 2. A road scene and the geometrized histogram of the color image of the 7-th horizontal strip.

The red closed curve in the right part of the figure demonstrates the intervals of the geometrized histogram corresponding to the part of the yellow marking in the chosen strip. Two yellow squares in the upper right-hand corner of the right part of the figure show the color of this group of intervals. Usually, there are too many intervals of the geometrized histogram of a grayscale or color image I_{kz} to solve complex real problems. To reduce the number of them, a clustering procedure is introduced [6, 8], which joins intervals $\mathbf{Int}_{kz}$ that are close as intervals on *Os* and have close intensity (for the geometrized histogram of the grayscale image) and intensity-color characteristics (for the geometrized histogram of a color image). The joined intervals are called grayscale or color bunches. Each strip St_{i} is described by the set of grayscale or color bunches B_{i}. Each grayscale bunch $b \in B_{\mathrm{i}}$ is characterized by the following parameters:

1. the localization interval $int_{\mathrm{b}} = [beg_b, end_b]$, belonging to *Os*;
2. $\Delta_{\mathrm{I}}^{b} = [I_{min}^{b}, I_{max}^{b}]$ and I_{mean}^{b} – the range and the mean value of the grayscale intensity;
3. the cardinality $Card^{b}$ (approximately, the number of points in the strip St_{i} whose coordinate x belongs to the localization interval $[beg_b, end_b]$ that have grayscale characteristics belonging to the range Δ_{I}^{b} of the grayscale bunch).

Each color bunch $b \in B_{\mathrm{i}}$ is characterized by the following parameters:

1. the localization interval $int_{\mathrm{b}} = [beg_b, end_b]$, belonging to *Os*;
2. $\Delta_{\mathrm{H}}^{b} = [H_{min}^{b}, H_{max}^{b}]$ and H_{mean}^{b} – the range and the mean value of the hue of b;
3. $\Delta_{\mathrm{S}}^{b} = [S_{min}^{b}, S_{max}^{b}]$ and S_{mean}^{b} – the range and mean value of saturation;
4. $\Delta_{\mathrm{I}}^{b} = [I_{min}^{b}, I_{max}^{b}]$ and I_{mean}^{b} – the range and the mean value of the grayscale intensity;
5. the cardinality $Card^{b}$ (approximately, the number of points in the strip St_{i} whose coordinate x belongs to the localization interval $[beg_b, end_b]$ that have the color characteristics belonging to the ranges Δ_{H}^{b}, Δ_{S}^{b}, and Δ_{I}^{b} of the color bunch).

Using the methods developed in [6—8], we can attach to each color image the graph of color bunches *STG* (STructural Graph). Suppose that we deal with horizontal strips. Each strip St_{i} is described by the set of color bunches B_{i}. $B = \cup B_{\mathrm{i}}$ is the set on nodes of *STG*. Two color (grayscale) bunches $b_1 \in B_{\mathrm{i}}$ and

$b_2 \in B_{i+1}$ in the adjacent strips are connected by an edge if their localization intervals have nonempty. A similar construction can be produced for grayscale bunches.

Color bunches b_1 and b_2 lying in the same strip are called adjacent if their localization intervals int_{b1} and int_{b2} are adjacent. Color bunches lying in the adjacent strips are called adjacent if their localization intervals have nonempty intersection. Edges of *STG* join all adjacent color bunches.

Informally, each bunch describes a certain part of a real object in the strip, its projection on *Os* and the description of numerical characteristics of this part of the object. The graph *STG* can be interpreted geometrically by overlaying localization intervals of its bunches ($[beg_b, end_b]$) on the middle lines of the corresponding strips. Figure 3 demonstrates the representation of an image by the *STG* graph. Color bunches of each strip are superimposed on or near the middle lines of the corresponding strips.

There are two types of color or grayscale bunches. Color or grayscale bunches of the first type are called dominating. Dominating bunches are the bunches that in some places of their localization interval int_b have the maximum density $dens_b = Card^b/l(int_b)$, where $l(int_b)$ is the length of the interval int_b. The second type of non-dominating bunches is generated by color or grayscale bunches that do not have maximum density at any point of their localization interval int_b. Non-dominating bunches are also important in image understanding. For example, as a rule, braking signal zones of distant vehicles in front on the road are non-dominating because of their small size (parts of the vehicle body are dominating in this case). To understand clearly the future behavior of the vehicle in front, we have to know the state of its signal zones to determine precisely the moment when it starts to brake.

Figure 3 demonstrates a set of color bunches for a road image taken by a usual cheap camcorder on one of the Russian roads. Color bunches corresponding to parts of the road almost always give dominating color bunches. Color bunches corresponding to road marking provide dominating bunches only in the nearest strips. The same is valid for grayscale bunches corresponding to road parts and parts of the road marking. Figure 3 clearly shows that the parts of road marking in the sequence of strips going to the upper part of the image generate in a sense a continuous sequence of dominating and non-dominating color (grayscale) bunches. As a rule, the bunches corresponding to parts of road marking have locally a maximum intensity and are small relative to adjacent dominating color bunches.

Figure 3. A road scene and the corresponding image of color bunches of the *STG* graph.

In the visualization, the localization intervals of dominating bunches are put on the entire middle line, while non-dominating bunches locate slightly below middle lines. The construction and numerous experiments with images have shown that color bunches represent any connected color or grayscale object in the real image with the size greater than three pixels. The description of a color or grayscale image by color bunches compresses the information on images from millions of pixels to several hundreds of bunches. However, this image description contains all important features of the image, including a description of the geometry of objects belonging to it. It is important to note that the main operations of the construction of *STG* are the same for all strips. This means that the computation of *STG* can be parallelized using particular processors in multiprocessor computers. Computational experiments on modern personal computers have shown that *STG* can be constructed in real time for HD video frames.

2.1. Problem statement and facilities for its solution

The aim of this investigation is to introduce a new technique for finding road marking, both permanent and temporary, in color images and video sequences of color images. For this purpose, the geometrized histogram of the color image and the geometrized histogram of its grayscale component will be employed. At the first step of the algorithms, in each strip, objects that can represent a part of a road marking in this strip are described and found. At the second stage, continuous sequences of the local objects found located in different adjacent strips are constructed. The construction of continuous sequences of local objects is based on a modification of the concept of left and right contrast curves (germs of left and right global objects) introduced in [6, 7]. The main specific feature of the method is that we construct the image of road markings in the graph *STG*. This makes it possible to avoid laborious operations on the image array and to provide real-time mode of the method. In the next subsection, the procedure for finding local objects in the strip that can represent different types of road marking is presented. In section 3, the algorithms for constructing the continuous sequences of local objects will be proposed.

2.2. Local constructions

The procedure for constructing local grayscale or color bunches in a narrow strip is described in detail in [6]. Let us recall its main steps. To start the construction, we select among intervals of the geometrized histogram the following ones: 1. dominating intervals that have a maximum density in some places; 2. intervals with maximum grayscale in some its places (for grayscale and color geometrized histograms); 3. intervals with maximum saturation (for color geometrized histogram). It is supposed that the density of the selected intervals is greater than some constant. We take these intervals as seeds in a clustering procedure. We add to a cluster determined by a selected seed the intervals of HG_i that have rather big intersection with the corresponding seed interval [6] and close intensity or intensity-color characteristics to those of the seed.

However, intervals with small density may be not included in this clustering procedure. This situation is typical for intervals corresponding to the road marking in distant strips. Small groups of intervals of the geometrized histogram of the grayscale component of the image, similar to those presented in Fig. 1, are called burst bunches or simply bursts. Informally, the bursts are generated by groups of rather small intervals that have close intensities and are mutually close as intervals and have an essential difference in intensity with the basic dominating bunches in a certain neighbourhood of their union on *Os*.

Let us present an algorithm for finding bursts. We label all intervals of the geometrized histogram as free or occupied. At initial stage of algorithm all intervals are free. A free interval becomes occupied (belonging to a burst), if it is chosen as a *seed* or belongs to a cluster of some seed. We choose intervals I_{kz} with a highest local intensity z in HG_i as seeds for burst bunches. For each seed, we define a neighborhood on the axis *Os* [a, b] □ I_{kz}. Let $[I_{\min}, I_{\max}]$ be the interval confining the intensities of the dominating intervals intersecting the neighborhood [a, b]. We require that distance of the intensity z from the intensity interval $[I_{\min}, I_{\max}]$ is more than a constant I_{not} making z noticeable against the basic dominating bunches whose intervals intersect [a, b]. Starting from each seed interval $\text{Int}_{kz} = [beg_{kz}, end_{kz}]$, we grow the burst b by adding intervals I_{ly}, $y < z$, [a, b] □ I_{ly}, $\text{Int}_{kz} \cap \text{Int}_{ly}$ is nonempty. For each new step, we obtain a new boundary intensity interval $[i_{\min}, i_{\max}]$ for the constructed burst b. The condition $[I_{\min}, I_{\max}] \cap [i_{\min}, i_{\max}]$ is empty is a necessary condition. Moreover, it is required that the gap between these intervals has to be more a certain constant. In addition, we employ by establishing the maximum gap between intensities of the added interval of the geometrized histogram and the boundary of the interval $[I_{\min}, I_{\max}]$ for a stop condition: if a ratio between maximum gap d_1 and a gap to the closest dominating bunch d_2 (Fig. 4) on HG_i is more than some value (e.g. 2), we stop the growing algorithm. Due to relative thresholding, the stop condition is independent of the average intensity of the marking and its vicinity. Thus, it enables to adapt the construction for different types of illumination and to efficiently find burst bunches**.**

For a color geometrized histogram, the procedure for finding local objects for constructing road marking starts from detecting rather small color bunches whose color varies in a certain range about the yellow one. This is explained by the fact that in real images, due to dust, wet places, illumination, the color of permanent and temporary yellow road markings may vary from orange to slightly yellow. The localization intervals of the bunches corresponding to yellow road marking, as well as localization intervals of bursts, have a limitation on their length. In both case (grayscale bursts and color bunches) the limitation on the length of localization intervals are not very strong. They are selected so that we can construct candidates for parts of road marking in strips without any information on the camera position and calibration. Because of this, the algorithms can operate successfully with a wide range of photos. Similar to the previous case, there are conditions on the intensity range of color bunches corresponding to road markings, discriminating them from the intensity ranges of the dominating color bunches located in a neighbourhood of the localization intervals of these bunches within the selected strips. All these conditions detect in each strip color bunches that can represent yellow road markings. All color bunches of such a type are labelled by a special array for each of the strips. It is necessary to note that close color bunches corresponding to road marking may be dominating in their strips, while distant bunches are not dominating as a rule. This can clearly be seen from Fig. 3.

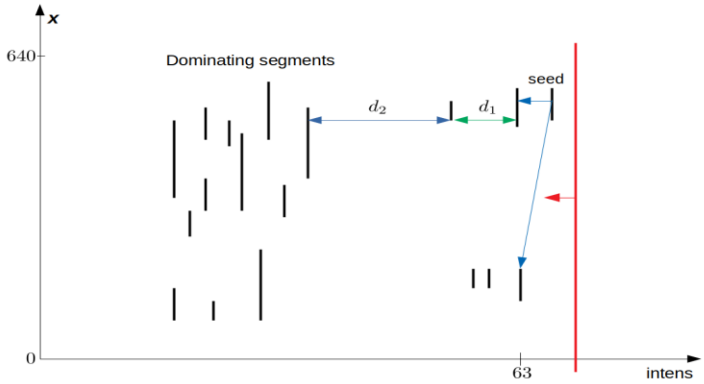

Figure 4. Burst bunches growing algorithm.

3. Construction of continuous systems of bursts or color bunches corresponding to permanent or temporary road markings

The problem of finding road markings is reduced to construction of continuous systems of local objects (bursts or color bunches) on *STG*. The method for constructing continuous objects on *STG* (the so-called left or right germs of global objects) was developed in [6, 7]. This method was implemented as a program operating in real-time. However, this method was aimed at constructing global objects, such as a road, roadsides, the sky [9], etc. and deals with rather big dominating bunches. This imposed certain limitations in the systems of rules for finding the next object in the sequence. As a rule, local objects for finding road marking are usually not dominating and rather small. The concept and the structure of the program have to be seriously modified to produce a new system for constructing road marking. The following subsections are devoted to describing the main points of these modifications.

3.1. Finding road markings based on grayscale bursts

Now we can construct chains of bursts going over strips of the image. Any chain is a sequence of adjacent bunches b_k, b_{k+1}, …, b_n, placed on adjacent strips St_k, St_{k+1}, …, St_n respectively. The chain can represent a real road marking line or its part, so it is worthwhile to find it.

We construct chain upward, beginning from its lowest strip, finding a corresponding adjacent burst on the next strip for the burst in the previous one. To find the corresponding next burst, the following two expected properties are employed: 1. A limited size and density of next bursts, and the smoothness criterion for the left and right boundary curve of the chain. According to the first rule, we filter out long horizontal bursts, which cannot be a part of road marking for obvious reason. For this purpose, approximate thresholds are established. For the closest strips, they are chosen to be suitable for a wide range of camera characteristics. For the next strips, as a rule, the lengths of candidates for extension cannot increase (in ideal case they depend only on the inclination of the line of the lane marking) and their densities should decrease. The smoothness criterion means that the absolute values of the differences of abs($end_{b(k+1)} - end_{bk}$) (the right curves) or abs($beg_{b(k+1)} - beg_{bk}$) (the left curves) for the adjacent bursts are bounded by a constant connected with the width of the strip. Introducing these constraints, we eliminate the effect of a sharp change in the shape of the boundary curve.

If there are no adjacent bursts in the strip to extend the chains, we terminate the process of chains construction. At the next stages, using the geometric characteristics of the chains, we can join some of them in order to obtain intermittent road marking. It is worth noting that the consideration of only burst bunches for chains construction makes our algorithm resistant to shadows and allows to take into account possible jumps of intensity near the border of illuminated and shadowed places. All these rules make our algorithm rather robust for changing situations in the road scene and significantly improve the quality of lane marking detection.

3.2. Finding road markings based on color bunches

Temporary road marking conveys very important information, and it is difficult to drive a car in the road without clear its recognition. It is also necessary to distinguish permanent and temporary road markings in the case when they both take place on the road. This makes it necessary to use color in detecting and recognizing road marking. To detect colored road marking, we employ the technique for constructing left and right germs of contrast global objects developed in [7, 8]. This technique is adapted to dealing with non-dominating color bunches and with color bunches satisfying additional conditions on their size, density, and color, similar to those formulated in the previous subsection. To avoid false extensions and to be able to deal with curvilinear road marking, we use conditions imposed on the relations between the directions of the segments joining the previous bunch and a possible next bunch D_1, the first bunch of the chain and the previous bunch D_2, and the first bunch of the chain and the possible next bunch D_3 of the chain. Since the curvature of the road marking is bounded, reasonable relations between D_1, D_2, and D_3 allow us to avoid false extensions. This means that we do not suppose that the line of road marking investigated is just a straight line. In this way, we can find the road marking of a curvilinear shape as well.

3.3. Using global image analysis for system improvement

The global image analysis developed in [9] makes it possible to solve in real-time such tasks as detection of the boundary of the vegetational and ground roadsides, finding the road and its parts in the case of occlusion caused by other participants of the traffic, and sky region detection. These tasks are solved based on the analysis of *STG* without operations at the pixel level. All such operations are performed in constructing *STG*. It was stressed in the previous papers on the subject of road marking detection that finding road boundaries and the vanishing point is highly desirable ([10, 11]). Since in our method the problem of finding road marking is solved together with the problems listed above and does not practically affect the total computation time, we use the opportunity to reduce the region of interest, decreasing the computational effort and eliminating a lot of noisy objects (like lights on a wet road). For example, we employ the information about the lower bound of the sky region.

4. Programming implementation, demonstration of the results, and the future work

The algorithms for finding road marking have been implemented by a program written in C++ and operating under Windows and Linux environments. This program processes video sequences in real time on standard computers with processors I3-I7 and records the results for each frame of the video sequence tested. For frames of resolution 640x480, the program solves the complex of tasks such as finding the sky, the road, the vegetation surrounding, and road marking with operation speed about 20 fps when all operations of the code are performed sequentially. The first experiments with the use of parallel computations of the mentioned processors have shown that the operation speed in computing *STG* can be reduced a factor of eight.

The program has been tested on many real images taken from cars on different Russian roads under different seasons, times of the day, under different illumination conditions. Figures 5, 6 present two examples from records of the results. The road marking bunches is painted in red color. These results were obtained for a division of the image in horizontal strips. However, to find stop lines and pedestrian crossings, it is natural to divide images into vertical strips. This will be subject of our next publication.

Figure 5. An example of finding a road marking.

Note that the developed technique is also applicable to finding such interesting objects as construction cones, emergency triangles, and road signs. In this case it is also necessary to construct chains of non-dominating local colored objects generating a specific shape. One of the next publications will be developed to this subject.

Using the methods for finding big objects in the road scene such as road, roadsides, sky, a system for the complex analysis of road scenes is also being prepared. Several examples of processing video sequences of road scenes can be found in [12].

Figure 6. An example of finding a road marking in an image from a video sequence.

5. References

[1] Dickmanns, E.D. Dynamic vision for perception and control motion. – Berlin: Springer-Verlag Ltd., 2007. – 474 p.

[2] Dickmanns, E.D. Vision for ground vehicles: history and prospects // Int. J. of Vehicle Autonomous Systems. – 2002. – Vol. 6(3). – P. 1-44.

[3] Wu, P.-Ch. Lane-mark extraction for automobiles under complex conditions / P.-Ch. Wu, Ch.-U. Chang, Ch.H. Lin // Pattern Recognition. – 2014. – Vol. 47(8). – P. 2756-2767.

[4] Tian, Y. Lane marking detection via deep convolutional neural network / Y. Tian, J. Gelernter, X. Wang // Neurocomputing. – 2018. – Vol. 280. – P. 46-55.

[5] Mathibela, B. Reading the road: road marking classification and interpretation / B. Mathibela, P. Newman, I. Posner // IEEE Transactions on Intelligent Transportation Systems. – 2015. – Vol. 16(4). – P. 2072-2081.

[6] Kiy, K.I. A new real-time method for description and generalized segmentation of color images / K.I. Kiy // Patt. Rec. Image Analysis: Advances in Mathematical Theory and Applications. – 2010. – Vol. 20(2). – P. 169-178.

[7] Kiy, K.I. Segmentation and detection of contrast objects and their application in robot navigation / K.I. Kiy // Patt. Rec. Image Analysis: Advances in Mathematical Theory and Applications. – 2015. – Vol. 25(2). – P. 338-346.

[8] Kiy, K.I. A new real-time method of contextual image description and its application in robot navigation and intelligent control / K.I. Kiy // Computer Vision in Control Systems-2, Innovations in Practice, Intelligent Systems Reference Library 75. – Springer, 2015. – P. 109-133.

[9] Kiy, K.I. A new method of global image analysis and its application in understanding road scenes / K.I. Kiy // Patt. Rec. Image Analysis: Advances in Mathematical Theory and Applications. – 2018. – Vol. 28(3). – P. 483-494.
[10] Liu, W. Extraction of lane markings using orientation and vanishing point constraints in structured road scenes / Sh. Li, X. Huang // International Journal of Computer Mathematics. – 2014. – Vol. 91(11). – P. 2359-2373.
[11] Hou, Ch. An efficient lane markings detection and tracking method based on vanishing point constraints / J. Hou, Ch. Yu // 35th IEEE Int. Control Conf. (CCC), 2016. – P. 6999-7004.
[12] Demonstration site [Electronic resource]. — Access mode: https://www.facebook.com/100004887018729/videos.

Acknowledgments
This work was partially supported by the Russian Foundation for Basic Research, projects no. 16-08-00880, 18-07-00127, and 19-08-01159.