An image understanding system based on the geometrized histograms method: finding the sky in road scenes

K I Kiy[1]

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

Abstract. In this paper, the technique provided by the geometrized histogram method for segmentation and description of color images is developed and improved in order to analyze the adjacency relation of left and right germs of contrast objects (left and right contrast curves) on the *STG*. This adjacency relation involves and generalizes the adjacency relation for regions in classical segmentation methods (the so-called RAG). Using this order relation, the adjacency relation for left and right germs of contrast objects is established. This order relation is also employed for finding relations between left and right germs with prescribed geometric and color-intensity characteristic that are not adjacent and lay apart at a distance. In addition, the concept of contours that are close to vertical on *STG* is introduced. Based on the adjacency relation proposed, a technique for constructing complex contrast objects with a prescribed geometric shape and color-intensity description is proposed. The developed technique is applied to analyzing road scenes in order to find the sky in video sequences. The results of finding the object in video sequences by a program complex, implementing these ideas, are presented and discussed.

1. Introduction

In spite of serious progress in image segmentation and analysis [1-4] and many new ideas arising in machine learning and deep learning of convolutional networks, there are still serious difficulties in implementing global image analysis in real time. These difficulties are mainly connected with many different objects occurred in the scene, occlusion, and difficult and diverse illumination conditions in the real world. This makes it difficult to analyze the joint behavior of several real objects and even to assemble parts of objects separated by occlusion using one or another method of image analysis (classical segmentation, sliding windows with machine learning or deep learning of convolutional networks). Moreover, as the mortal accident with the Tesla pilotless vehicle has shown, it is crucially important not only to classify any frame, but also to have clear understanding of the state of important objects in the image and to produce their conceptual description in order to recognize the case of their complete change due to possible occlusion occurred. It is also necessary to analyze the dynamics of the motion and changes in the shape of objects in video sequences. In this case, it can prevent us from recognizing the body of a blue van as a part of the sky. It is necessary for the image understanding system of a robot to be able to select the sky region, to determine its shape and the location of its boundary. It is also desirable to describe the semantic type of the sky region. The information about the sky region has to contain its color and intensity characteristics and a semantic interpretation of the regions over which it lies. The complete change in this information within a small number of frames

can inform the system about the dangerous occlusion occurred. This information has to be used in order to prevent a possible accident.

In this paper, we propose an image understanding system that can solve such problems in real time using only standard computational facilities. The approach to designing image understanding systems of such a type is based on the geometrized histograms method proposed by the author [5-7]. This method not only segments color images in real time, but also makes it possible to construct adjacency relations between detected objects and to introduce higher-order adjacency relations for objects that are rather distant in the image. This technique is applied to designing an image understanding system for finding and analyzing sky regions in images and video sequences of images of road scenes. The designed image understanding system finds the sky in video sequences of suburban and country roads very efficiently. The results can be found on the sites [9, 10].

2. A brief description of the geometrized histograms method

This method combines the advantages of statistical methods connected with studying conventional histograms of color or multichannel images [11] (real-time results) and conventional segmentation methods based on regions and contours [1, 2] (detailed shape-description). It was designed keeping in mind the application to constructing real-time image understanding systems. The origin of the method is dated back to the last 1980s, when the very early version of the method was applied to designing a vision system of a pilotless vehicle [12]. In addition, many papers are devoted to the problem of separating contours that belong to the boundaries of real regions. For example it is the main point in [13] and other papers can be found in the references of this paper. This problem can also be solved within the scope of the geometrized histograms method. Moreover each such contour can be furnished with the data characterizing the part of the region which it bounds (the so-called left and right contrast boundary curves). This data is very convenient for constructing real objects from different parts in the case of complex illumination, using both intensity-color characteristics and shape description of the considered parts.

A detailed description of the geometrized histograms method can be found in [5-7]. Let us explain briefly the concept of the geometrized histogram of a color image. To construct the geometrized histogram, the image is divided into strips St_i, $i = 1,…n$, of the same width W with boundaries parallel to the horizontal or vertical axis of the image plane Os. Suppose that we deal with horizontal strips. The case of vertical strips is considered in a similar way. To describe approximately the image in a chosen narrow image strip, it is necessary to describe approximately the distribution of values of the vector function specifying it. The vector functions (R, G, B), (H, S, I), or $(G/(G+B), G/(G+R), I)$, introduced by the author, can be examples of this vector function. This approximate description will be called the geometrized histogram of the image in the strip. Let us explain 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 set of points (x, y) of the strip St_n, where $f(x, y) = z$. Since we deal with the discrete representation of the image, 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_n 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_n. The collection of intervals I_{kz} approximately describes L_z, since the set of level z belongs to the preimage of $\cup_k I_{kz}$, $L_z \subset Pr^{-1}(\cup_k \mathbf{I}_{kz})$, 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} for different z may have a nonempty intersection on Os. This occurs when the intervals correspond to different objects in the strip and one object lies over another in it. The space of intervals $\cup_k^z I_{kz}$ is called the local geometrized histogram ($\mathbf{HG}_n$) of $f(x, y)$ in St_n.

Let us show how to generalize this construction to the case of a vector function giving a color image. We deal with the function $(G/(G+B), G/(G+R), I)$ [5, 7], representing the color image. 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, added 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 St_n. Each interval I_{kz} of the geometrized histogram of CF is called the localization interval $\mathbf{Int}_{kz} = [beg_{kz}, end_{kz}]$ of the geometrized histogram of the color image in St_n. Since each interval I_{kz} is furnished with the classical histogram of the other color component $G/(G+R)$, we can attach to $\mathbf{Int}_{kz}$ definite ranges of color characteristics and the mean values of these color features. Therefore, for each localization interval $\mathbf{Int}_{kz}$, it is possible to find the range and the mean value of its hue $\Delta_H^{kz} = [H_{min}^{kz}, H_{max}^{kz}]$ and H_{mean}^{kz}, the range and the mean value of its saturation $\Delta_S^{kz} = [S_{min}^{kz}, S_{max}^{kz}]$ and S_{mean}^{kz}, and the range and the mean value of its grayscale intensity $\Delta_I^{kz} = [I_{min}^{kz}, I_{max}^{kz}]$ and I_{mean}^{kz} [5, 7]. In addition, each interval of the geometrized histogram has the cardinality $Card^{kz}$.

Usually, there are too many intervals of the geometrized histogram I_{kz} to solve real problems. To reduce the number of them, a clustering procedure is introduced [5, 7], which joins intervals $\mathbf{Int}_{kz}$ that are close as intervals on Os and have close intensity-color characteristics. The joined intervals are called color bunches. Each strip St_i is described by the set of color bunches B_i. Each color bunch $b \in B_i$ is characterized by the following parameters:

1. the localization interval $int_b = [beg_b, end_b]$, belonging to Os;
2. $\Delta_H^b = [H_{min}^b, H_{max}^b]$ and H_{mean}^b – the range and the mean value of the hue of b;
3. $\Delta_S^b = [S_{min}^b, S_{max}^b]$ and S_{mean}^b – the range and mean value of saturation;
4. $\Delta_I^b = [I_{min}^b, I_{max}^b]$ 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).

In this way, we can attach to each color image the graph of color bunches STG (STructural Graph). $B = \cup B_i$ is the set on nodes of STG. 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 superimposing localization intervals of bunches ($[beg_b, end_b]$), belonging to it, on the middle lines of the corresponding strips. Figure 1 demonstrates the representation of an image by the STG graph. Color bunches of each strip are superimposed near middle lines of the corresponding strips of the grayscale, image corresponding to the considered color image.
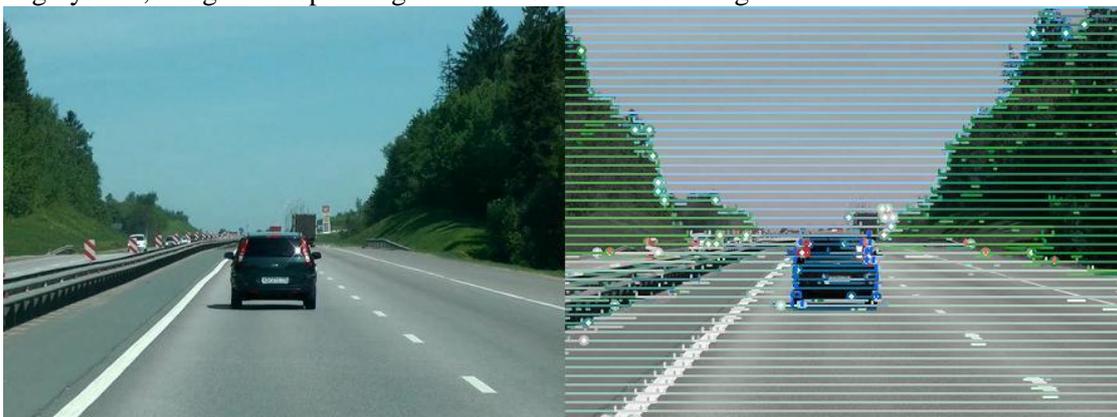

Figure 1. A road scene and the corresponding image of color bunches of the STG graph.

There are two types of color bunches. Color bunches of the first type are called dominating bunches. A dominating bunch is a bunch that at some points of its localization interval int_b has a maximum density $dens_b = Card^b/l(int_b)$, where $l(int_b)$ is the length of the interval int_b. It is clear that the localization intervals of dominating bunches generate a covering of the middle line of the corresponding strip. In this visualization, the localization intervals of dominating bunches are

superimposed on the entire middle lines of strips. In addition, we have some kinds of color bunches that are not dominating. These bunches may also be very important. For example as a rule, the signal zones of a distant vehicle (side-lights, brake lights) may have densities less than the densities of bunches, corresponding to the body of the vehicle. However, these bunches are very important in order to recognize the next actions of the driver of a vehicle going in front of our car. In the visualization, the color bunches of the second type are put slightly below the middle line. The procedure of construction of color bunches was prepared keeping in mind the possibility of detecting any connected colored set having a contrast with surrounding objects in the image. Numerous experiments with images have shown that color bunches represent any connected color object that have a contrast with in the real image with the size greater than three pixels. The description of a color 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.

2.1. Continuous object on STG

In [6, 7] the concepts of left and right contrast curves (left and right germs of global contrast objects) in *STG* were introduced, and a bipartite graph of left and right contrast curves *LRG* was constructed. Let the image be divided into horizontal strips. A left (right) contrast curve is a chain of color bunches b_i with contrast right (left) neighbors located in adjacent strips St_i $i = k, k + 1, \ldots k + d$, such that the intensity-color characteristics of these bunches vary continuously from strip to strip, as well as the coordinate x of their left (right) ends [6, 7]. Such a chain is constructed upward, beginning from its lowest strip, finding the continuous extension of the previous bunch to the next strip [6, 7]. Figure 2 presents two examples of left (right) contrast curves (germs of contrast objects) in images.

Figure 2. Two parts of the sky represented by germs of global contrast objects in *STG*.

In this way, up to 256 different left and right contrast curves are found. By the construction, no more than one left and one right germ of global contrast objects can pass through any color bunch. On the set of all color bunches, functions Germ$_{left}$(*STG*) and Germ$_{right}$(*STG*) are determined. At each color bunch, these functions take as the value the number of the left (right) germ passing through this bunch or −1, if there is no such germ. Each color bunch of a left (right) contrast curve has a contrast contact with its right (left) neighbor. It is supposed that each left (right) contrast curve is a left (right) part of a certain hypothetical global object. Any left (right) contrast curve has its own linear geometric pattern determined by the discrete set of left (right) ends of the localization intervals of its color bunches. In the right image of Fig. 2, the presented left contrast curve (painted for visibility by dark intervals) is simultaneously a right contrast curve, since each of bunches belonging to it has both left and right contrast neighbors (parts of the forest or the boundary of the frame). Left and right ends of color bunches of the contrast curve of the right side of Fig. 2 specify standard boundaries of a sky region in a road scene. The right contrast curve in the left image of Fig. 2 has to be completed to generate the whole sky region. Of course, it is the most typical situation. In what follows, we present a reasoning system that performs the operation of extension sky regions. Together with a linear pattern, each germ of a global object *G* has an area pattern determined by the figure in the image plane generated by the localization intervals of its color bunches. For each germ *G* of a contrast object, we define its weight $W_G = \Sigma_j l(int_{bj})$ as the sum of lengths of localization intervals of its color bunches b_j. The substantial characteristic of an object in a perspective image is its behavior at infinity (in the motion to the upper

boundary of the image). To determine this behavior, the image is divided into zones by straight lines parallel to its lower boundary. Several strips of the image can be involved in each zone. Denote by W_G^i the part of the weight of G belonging to the zone with number i. The sequence $\{W_G^i\}$ determines the behavior of G at infinity. Both the linear and area patterns determine the full geometric pattern of G. In addition to the geometric pattern, each contrast curve has intensity-color characteristics determined by its color bunches. For the image of the right side of Fig. 2, together with the geometry of boundaries, we can produce the label "bright blue sky without clouds". To be able to assemble the sky region from available left and right germs of global contrast objects, we have to explain a new technique developed for this purpose in [6–8].

2.2 Adjacency relations graphs

In each strip, we are able to select among all color bunches a basic set of bunches dominating in some part of the strip (having the greatest density in it). It is obvious that localization intervals of dominating bunches give a covering of the middle line of the corresponding strip. For each dominating bunch, it is possible to find its closest left and right dominating neighbors. Using this construction, we can select a completely ordered basic subset of dominating color bunches that provide a covering of the middle line. It is possible to introduce a complete ordering in this basic subset and to number dominating color bunches of this subset from 0 to a certain k. Figure 3 demonstrates basic subsets for two strips of the image of Fig. 1. In addition, Figure 3 shows that all important parts of objects in these strips are taken into account in the descriptions of strips by color bunches.
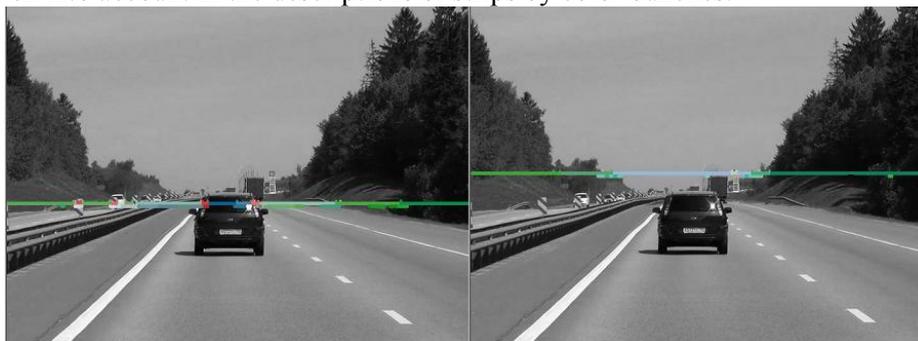

Figure 3. Two basic sets of color bunches in two different strips.

All linear ordered basic subsets of bunches, joined for all strips, generate on the image a "search lattice" SeachLat (*STG*) [8]. The constructed SeachLat (*STG*) (bunches are numbered with preservation of the adjacency relation) allows one to construct the adjacency graph *ADG*, which determines adjacency relation for left (right) germs of contrast objects in *STG*.

Each left (right) contrast curve (germ of global contrast object) is a continuous sequence of color bunches in a chain of adjacent strips (see Fig. 2). The values coordinates of left (right) ends of the localization intervals vary continuously, as well as their intensity-color characteristics [6]. By construction, only one left or right contrast curve can pass through any color bunch. If this curve exists, then it is uniquely determined by the functions Germ$_{left}$(*STG*) and Germ$_{right}$(*STG*). Suppose that we have a germ of a contrast global object G. Starting from the first bunch b_1 of this germ in its first strip and moving to the left and right of it, we find all adjacent germs of G in the considered strip. In this way, considering the germs passing through the direct neighbors of b_1, we can construct the direct adjacent germs of G in the strip. Moving from strip to strip, we are able to construct the part of the adjacency graph *ADG* connected with the left and right adjacent germs.

Consider the extension of adjacency relations in the downward and upward directions. In the construction of the set of color bunches in any strip, we generated a structure that informs us about all color bunches that pass through a definite point belonging the axis *Os* (the middle line of this strip). For each color bunch, we are able to find its first and last adjacent bunches in SeachLat (*STG*) in the upper and lower adjacent strips, based on this structure. Using this information, we are able to extend the adjacency relations downward and upward. Considering in each strip adjacent germs, passing through the next bunches of the search lattice, we are able to introduce a multiple adjacency graph

MADG or adjacency graphs of higher orders Adg$_i$(*STG*). *MADG* makes it possible to perform global image analysis, e.g., to analyze components of the same global object even in the case of occlusion [8]. For example, we are able to investigate two roadsides (left and right) simultaneously or two parts of any object separated by occlusion. The graph *MADG* makes it possible to assemble complex real objects which contain heterogeneous parts. In this graph, not only relations between objects that have common boundaries are established, but between objects separated by occlusion as well. New results connected with a detailed construction and application of *ADG* and *MADG* can be found in [8].

3. Construction of a reasoning system for finding the sky in road scenes
The problem of finding the sky is one of the problems solved in the course of developing the control system of the autonomous robot AvtoNiva, produced by a research group in Keldysh Institute of Applied Mathematics of the Russian Academy of Sciences. For this purpose, it is not necessary to obtain a detailed description of the sky region at the pixel level. We need only approximate, qualitative and semantic description of the sky region that can be used in the control system for qualitative estimation of the road neighborhood and for detecting possible occlusion caused by unpredictable actions of other participants of the traffic. The statement of the problem under these conditions is described in the next subsection.

3.1 Problem statement and quality estimation
It is supposed that the image of a road scene is divided into a number of strips of the same width with the boundaries parallel to the horizontal axis of the image plane. For example for an image of resolution 640x480, we used 48 strips. We have to determine an array *Boun*(*n*), which specifies the pixel boundary of the sky for each column *n* of the image array. It is not supposed that the sky region is simply-connected. Due to occlusion it may contain several components. We have to find approximate color and intensity characteristics of each connected component and to describe its possible semantic type, e.g., "bright blue sky without clouds". If the detected region of the sky in the form specified above takes into account about 90% of real pixel sky region (with minimum possible false positives) and the lower boundary of the sky region is found with the accuracy up to one strip, then the solution found is considered as quite successful.

Using this data, describing the character of the sky boundary, we can obtain certain useful information about the road behavior (a straight road, a forthcoming turn, descent, ascent, etc.) even in the case of heavy occlusion of the road caused by other vehicles. We are also able to recognize the dangerous occlusion caused by the car in front, taking into account among other features the complete change in the pattern of the sky region. It is also supposed that the problem has to be solved on a standard PC in real time. Since the accuracy of determining the sky boundary is up to one strip, it is proposed to find the set of color bunches B_S belonging to the sky region. Then to find the pixel boundary in each column *n*, the lowest bunch $b_1 \in B_S$ passing through this column is found. We take the lower boundary of the strip within the localization interval of b_1 as the pixel sky boundary *Boun*(*n*) of the sky region in this place. These assumptions make it possible to solve the problem in a real-time mode. It is clear that the boundary of the sky is given by a piecewise constant function of *n*.

3.2 Specific features of the problem and its solution
The careful study of a large number of images of scenes containing sky regions, taken under different illumination conditions, at different times of the day, and during different seasons, has shown that the sky region may be a very complex object. It may contain many different more or less homogeneous parts that are quite different in color and intensity. The sky region is especially diverse during sunset or sunrise. The appearance of clouds may change the pattern dramatically. The presence of sky-similar objects such as walls of buildings (especially without windows) makes the problem even more complex. Under these conditions, it may be impossible to solve the problem using only one frame. Sometimes even a human may fail to determine the boundary of the sky quite correctly using only one image. Only external knowledge and views from other positions may help. For instance in Fig. 4, a part of a circle of the white antenna left of the low white clouds over the roof of a Sberbank building gives an example of such a situation.

Therefore, to obtain an adequate solution, at least two stages of solving the problem are necessary. At the first stage, a single frame is analyzed and a preliminary solution is described. At the second stage, we compare and study a set of adjacent frames in order to provide the final solution.

Let us describe the first stage of solution. At the first step, we generate a preliminary conceptual and semantic description of all left and right contrast curves constructed by algorithms described in [6, 7]. To describe the geometry of any contrast curve, we use methods proposed in [8].

Let us briefly describe them. For this purpose, we divide the boundary points of any contrast of curves (a set of left (right) ends of localization intervals of the color bunches involved in this contrast curve) into branches on which the coordinate x (the horizontal coordinate in the image plane) of its nodes increases or decreases. This is aimed at finding the perspective in the image. As the additional constraint, we suppose that the absolute values of the differences abs($end_{b(k+1)} - end_{bk}$) (right curves) or abs($beg_{b(k+1)} - beg_{bk}$) (left curves) for the adjacent nodes of the curve on these branches are bounded by a constant connected with the width of the strip. Introducing these constraints, we eliminate the effect of sharp change of the shape of the boundary curve.

Then we test whether these branches belong to certain straight line segments or they are convex or concave. To test the linear hypothesis, we use histograms of inclines of the segments connecting adjacent nodes of the contrast curves. Details can be found in [8]. As was mentioned above, the distribution of lengths of localization intervals along the curve or the sums of lengths W_G^i within selected zones determines the behavior of the corresponding germ of a global object at infinity. The parameters of this distribution distinguish contrast curves that are long and narrow with decreasing lengths (like parts of the road) and long and wide with increasing and non-decreasing lengths (like forests, fields, parts of the sky, bodies of cars). Then we select both the left and right contrast curves with the maximum weight $W(b)$ that have color-intensity characteristics possible for parts of the sky and locate in the top part of the frame and have the corresponding behavior at infinity. Based on the left and right curves of the maximum weight found, using the search lattice on the image, we construct the whole region of the sky. Moving to the left or right on the search lattice, we add sky-similar germs and stop the extension of the sky region when the regions classified as forests, fields, roads, etc., occur. In a similar way, moving on the search lattice to the bottom of the image, we add sky-similar germs again and stop the extension when the regions mentioned above are met. It is especially difficult to eliminate regions generated by buildings, having intensity-color characteristics similar to those of sky regions. For this purpose, the reasoning system finds sky-similar regions with straight boundaries and tests whether these regions have subobjects inside with vertical boundaries (windows, doors).

To study suspicious regions, we need some new definitions. For this purpose, we introduce concepts of contours in *STG* having a rather big angle with the axis *Os* (the boundary lines of strips into which the image is divided). For horizontal (vertical) strips, we obtain contours in *STG* that are close to vertical (horizontal) ones. In turn, these contours give the corresponding vertical (horizontal) contours in the image if we consider pixel coordinates of the ends of the corresponding color bunches constituting the contours in *STG*. Let us give the definition of contours close to perpendicular to the axis *Os* (simply, contours in what follows) in *STG*. These contours are generated by left (right) ends of basic color bunches, belonging to the search lattice SeachLat. For each strip of the image, an array $Loc[i]$ of length DimX/k is generated, where DimX is the horizontal dimension of the image array, while k is a compression coefficient (e.g., $k = 4$). At each point i, $Loc[i] = d$, where d is the number of the basic bunch with localization interval passing through the point ki of the middle line of the corresponding strip. Consider a strip St_k. Let $b_k \in$ SeachLat be a basic color bunch with the localization interval [beg_{bk}, end_{bk}]. Remind that image strips are numbered bottom to top. Consider the next strip St_{k+1}. Using the array $Loc[i]$ of the next strip, we find the basic color bunch that passes through beg_{bk} (end_{bk}) in the next strip. Then moving along SeachLat, we find the basic color bunch b_{k+1} such that the distance between beg_{bk} (end_{bk}) and $beg_{b(k+1)}$ ($end_{bk(k+1)}$) is minimal. If this distance is less than a certain constant that bounds the angle of the shift, the contour is extended to b_{k+1}. Using left (right) ends of bunches, we obtain left (right) contours. For a contour of length n, the following characteristics are introduced: 1. the maximum deviation from the vertical max_i abs($end_{bk} - end_{bk+i}$); 2. the total deviation from the vertical direction $dtot = $ abs($end_{bk} - end_{bk+n}$)/n.

Definition. A contour is close to the perpendicular to the axis *Os* if *dmax* and *dtot* are bounded by certain constants.

If we consider two graphs of color bunches STG_V and STG_H, constructed for vertical and horizontal strips, and find in both graphs contours close to perpendicular to the axes *Os*, we can generate in the image contours close to vertical and horizontal simultaneously. This technique is employed to select buildings in the image and to eliminate their regions from the sky region. Figure 4 demonstrates two complex examples of images of a city landscape.

In spite of several small mistakes, the level of the sky in both images is found quite correctly. The results of processing video sequences by the presented system can be found in [9, 10]. It is important to note that the results completely support the conclusion that the problem of finding the sky in images is not local and requires careful global analysis of the frame. At the end of the first stage, we find the boundary of the sky region in the form of an array specifying the number of the first pixel of the sky region in each column of the image *Boun*(*n*), where *n* is the number of the corresponding column. We also have the set *sk_germs* that contains of all germs (contrast curves) included in the sky region. Analyzing the parameters of the germs of the set, we produce the semantic description of the sky image.

Figure 4. Sky regions in images of a city landscape.

At the second stage of the solution, we compare arrays *Boun*(*n*) for the current and previous frames. We also compare the semantic descriptions of the sky regions found of adjacent frames. In the case of their correspondence, we adopt a new solution. Otherwise, we analyze the differences and decide whether a dangerous occlusion occurs, taking into account other features such as possible signal zones (side-lights, brake signals) and vertical and horizontal contours of a hypothetical vehicle in front.

The technique of contours close to vertical or horizontal ones can also be applied to the analysis of scenes in villages and towns to detect fences, cottages in order find roads in complex conditions of shadows and absence of road marking. This will be a subject of the next publications.

4. Software implementation, demonstration of the results and discussion

The image understanding system has been implemented by a program written in C++ and operating under Windows and Linux. 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 operation speed is about 20 fps. The program has been tested on dozens of video sequences taken from cars on different Russian roads under different seasons, times of the day, under different illumination conditions. Figure 5 presents several examples from records of the results for three video sequences. On country roads from the considered series the percentage of positive results varied from 98 to 100 %. Even on new video sequences of this type processed for the first time without modifying the program, a very high percentage of positive results were obtained. Some problems may appear when processing video sequences taken in villages and towns. This is connected with buildings with walls that cannot be distinguished from the adjacent parts of the sky using only one frame. The data base of rules and features are being modified, as well as the work with adjacent frames, in order to eliminate these problems. The results of processing several video sequences can be found in [9, 10]. In addition to the sky region, the video system of the control system

of AvtoNiva finds other regions interesting for controlling the vehicle such as the boundaries of the vegetation regions, road regions, and the other vehicles on the road. The solution to a part of these tasks was described in [8] and further results were presented in a brief publication [14]. The detailed publication on this topic is being prepared.

Figure 5. Examples of records of experiments with video sequences.

5. References

[1]   Forsyth D A and Ponce J 2003 *Computer Vision, a Modern Approach* (London: Prentice Hall)
[2]   Mishra A K and Aloimonos Y 2009 Active segmentation *Int. J. Humanoid Rob.* **6** 361-366
[3]   Chen Ch, Papandreou G, Kokkinos I, Murphy K, and Yuille A L 2016 Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected CRFs *Preprint arXiv 1606.00915*
[4]   Divvala S K 2012 *Context and subcategories for sliding window object recognition* PhD Thesis (Pittsburgh: Carnegie Mellon University)
[5]   Kiy K I 2010 A new real-time method for description and generalized segmentation of color images *Pattern Recognit. Image Anal.* **20** 169-178
[6]   Kiy K I 2015 Segmentation and detection of contrast objects and their application in robot navigation *Pattern Recognit. Image Anal.* **22** 338-346
[7]   Kiy K I 2015 A new real-time method of contextual image description and its application in robot navigation and intelligent control *Computer Vision in Control Systems-2 Innovations in Practice Intelligent Systems Reference Library* **75** 109-133
[8]   Kiy K I 2018 A new method of global image analysis and its application in understanding road scenes *Pattern Recognit. Image Anal.* **25**
[9]   *Electronic Materials* (Access mode: http://video.mail.ru/kikip_46/_myvideo)
[10] *Electronic Materials* (Access mode: https://www.facebook.com/100004887018729/videos)
[11] Denisova A Y and Sergeev V V 2016 Algorithms for calculating multichannel image histograms using hierarchical data structures *Computer Optics* **40(4)** 535-542 DOI: 10.18287/2412-6179-2016-40-4-535-542
[12] Kiy K I, Klimantovich A V and Buivolov G A 1995 Vision-based system for road following in real time *Proc. 7th Int. Conf. on Advanced Robotics (San Feliu de Goixols, Catalonia, Spain)* **1** 517
[13] Belim S V and Kutlunin P E 2015 Boundary extraction in images using a clustering algorithm *Computer Optics* **39(1)** 119-124 DOI: 10.18287/0134-2452-2015-39-1-119-124
[14] Kiy K I 2018 Image understanding systems based on the geometrized histograms method *Proc. 7th Int. Conf. on Extreme Robotics and Conversion Tendencies (Saint-Petersburg)* 140