SCIENTIFIC PAPERS
OF THE UNIVERSITY OF PARDUBICE
Series B
The Jan Perner Transport Faculty
14 (2008)

ORIENTATION OF VEHICLES USING A CAMERA

Josef ŠROLL

Department of Informatics in Transport

1. CURRENT SITUATION IN THE STUDIED PROBLEMS

Locating position is a basic requirement when driving a vehicle. A typical way is the driving person´s perception. In an attempt to replace the driver, one of the decisive factors is the substitution of human perception by other methods. Persons prevailingly perceive light-based information using their vision, less frequently other senses – noise in tunnels, vehicle vibrations caused by surface irregularity, etc. Different technical means based on various principles have been used to locate a vehicle position. Taking in view the ways of such location, there exist different accuracy in measuring and different suitability. Radio-signal based systems (LORAN, GPS) can be used only in suitable environment, free from reflexion. Inertial navigation systems do not depend on external environment, however, they generate an error that gradually increases (cumulates) depending on time.

The use of a front view camera is assumed as a research tool. A part of the carriageway is found in the field of vision, an area in which the horizon and different size objects appear along the road.

The problems to solve are rather extensive. The system will probably not be a dominant navigation system, however, it may well become irreplaceable in some specific cases, such as the orientation of robots in confined areas using a mini camera in places where other systems cannot work (pipelines, sampler navigation e.g. in laparoscopy, etc.) We have different mini-cameras at disposal today. Unfortunately, they are always coded in a certain format suitable for transmission, but not for further image processing. The first step will always be its de-coding from the used format (JPEG, AVI, TV standard) into a simple bitmap. However, we have sufficient support for this process, that is, besides the requirements for certain equipment, there is no further difficulty. To process images in a required way, the support is lacking. Certain solution was hoped to be an openCV library (CV - computer vision) by Intel. This contains numerous algorithms for objects identification, images conversion, etc. However, the problem solved here is ignored, being considered unimportant

2. METHODS OF ANALYSIS AND SOLUTION OF THE PROBLEM

Several specialized programmes have been developed to unambiguously identify several aspects of the problem. The first was a programme that made an auto-correlation image function at different magnifying degree, which stimulated the shift.

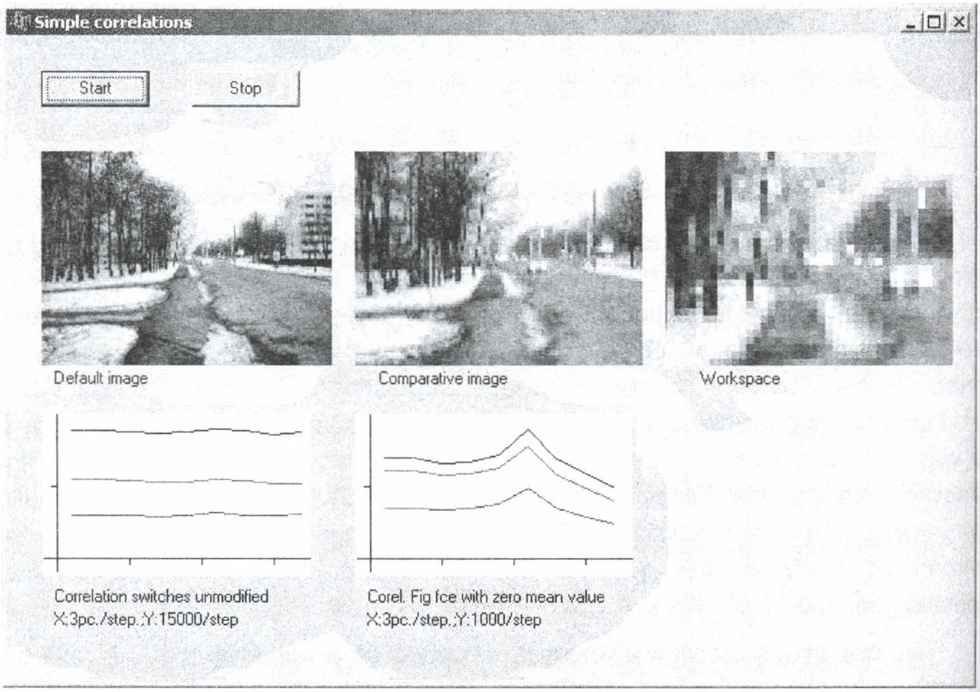


Fig. 1 *Correlation function of original image and image without the middle value*

The first step is the necessity to remove the middle colour value from the images. Unless you do so, the correlation coefficient will be high, but lacking a marked culmination.

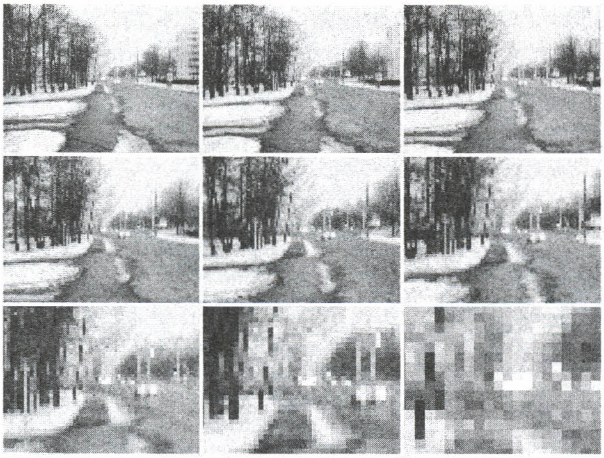


Fig. 2 *Individual parts of a digitally shifted image*

Then it was needed to divide the image into sections that can contribute to the determination of the vehicle position, and into sections in which too many interfering objects appear.

3a	3b	3c
2a	2b	2c
1a	1b	1c

Fig. 3 *Image division*

To determine the importance of individual sections, suitable weighing coefficients were sought. It is important for practical use that several subsequent images provided for a correlation in excess of an average. This depends on several factors, especially on the size of the objects alongside the route. For theoretical calculations, we can use different sizes, the most suitable being the scanning of real scenes, and their evaluation and calculation of the average. Since theoretical considerations cannot integrate all the feasible options, numerous field measurings have been carried out based on gaining an image and evaluation of correlation functions of its individual parts. The outcomes resulted in making summarized tables for road and for rail vehicles.

Tab. 1 *Road vehicle coefficient*

count		
0,62	0,17	0,68
0,76	0,37	0,81
0,67	0,28	0,79

coefficients		
0,237	0,033	0,132
0,148	0,072	0,157
0,130	0,054	0,153

Tab. 2 *Rail vehicle coefficients*

count		
0,63	*0,28*	*0,64*
0,66	*0,59*	*0,67*
0,58	*0,16*	*0,59*

coefficients		
0,095	*0,042*	*0,096*
0,099	*0,089*	*0,081*
0,087	*0,024*	*0,089*

The coefficients for driving in free terrain, generally the space with no other traffic, do not have to be taken in consideration for the images parts loaded by traffic situation. Therefore the credibility coefficients set up is not required for mutually different values; the set up to 1/9 is sufficient, so that

$$\sum_{i=1}^{9} k_i = 1 \qquad (1)$$

where k_i are weights (or credibility coefficients) of individual image sections.

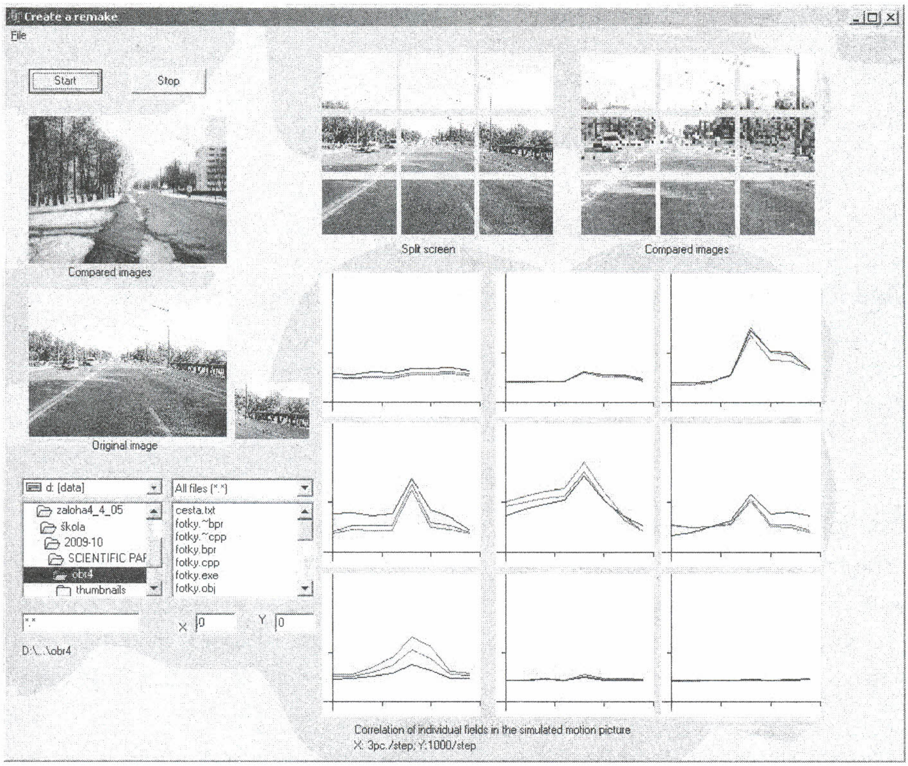
The camera image that underwent a correlation with several previously saved images, or with all the images, makes a correlation coefficient with each of these images. Correlation is carried out for each color file individually. The resultant correlation coefficient of two images presents their eucleidal distance:

$$k_V = \sqrt{k_R^2 + k_G^2 + k_B^2} \qquad (2)$$

where:

k_V .... resultant correlation coefficient,

k_R, k_G and k_B .... correlation coefficients of one colour file.



Correlation of individual fields in the simulated motion picture
X: 3pc./step; Y:1000/step

Fig. 4 *Determination of divided image correlations*

The highest coefficient found and the coefficients of the previous and following image are taken in consideration for further processing. An approximation parabola $y = ax^2 + bx + c$ is interleaved among the three coefficients and its culmination is found. The culmination coordinates are considered the point searched for.

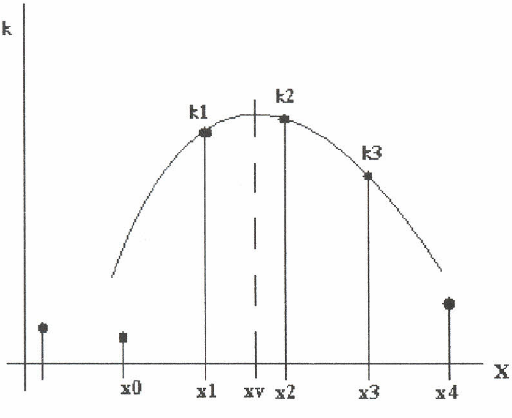


Fig. 5 *Aproximation parabola*

In the coordinate system where $x_1=0$, the coefficients of the parabola are the following:

$$c = k_1$$
$$b = \frac{4k_2 - 3k_1 - k_3}{2h}$$
$$a = \frac{k_1 - 2k_2 + k_3}{2h^2} \qquad (3)$$

And the coordinates of its culmination are the following:

$$x = -\frac{b}{2a} \qquad (4)$$

For non-zero x_1 it is:

$$x_v = x_1 + x \qquad (5)$$

The correlation coefficients width has been defined as a value in which the excess value is higher then 50% of the maximum.

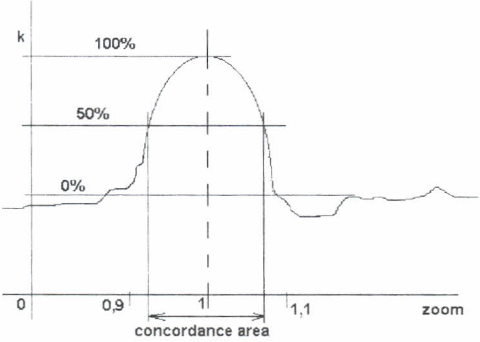


Fig. 6 *Definition of correlation coefficient width*

Theoretical solution of the scanning frequency is based on the optical camera resolution.

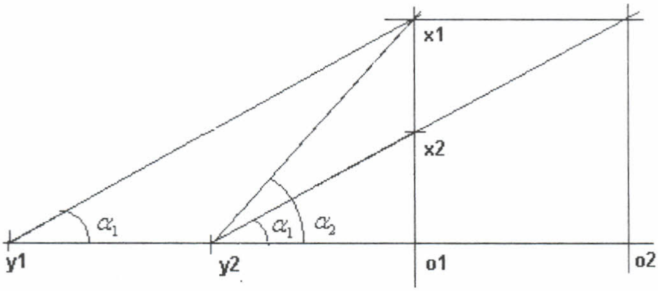


Fig. 7 *Shift of a camera equipped vehicle*

To determine the optical camera scanning frequency, the following relations have been derived:

$$y_1 = \frac{z}{tg\beta_1} \qquad (6)$$

$$y_2 = \frac{z}{tg\left(\beta \frac{n-1}{n}\right)} \qquad (7)$$

where:

y_1, y_2	camera coordinates
z..	minimum distance between objects alongside the road
β	camera vision angle
n..	camera resolution (number of pixels per a line)

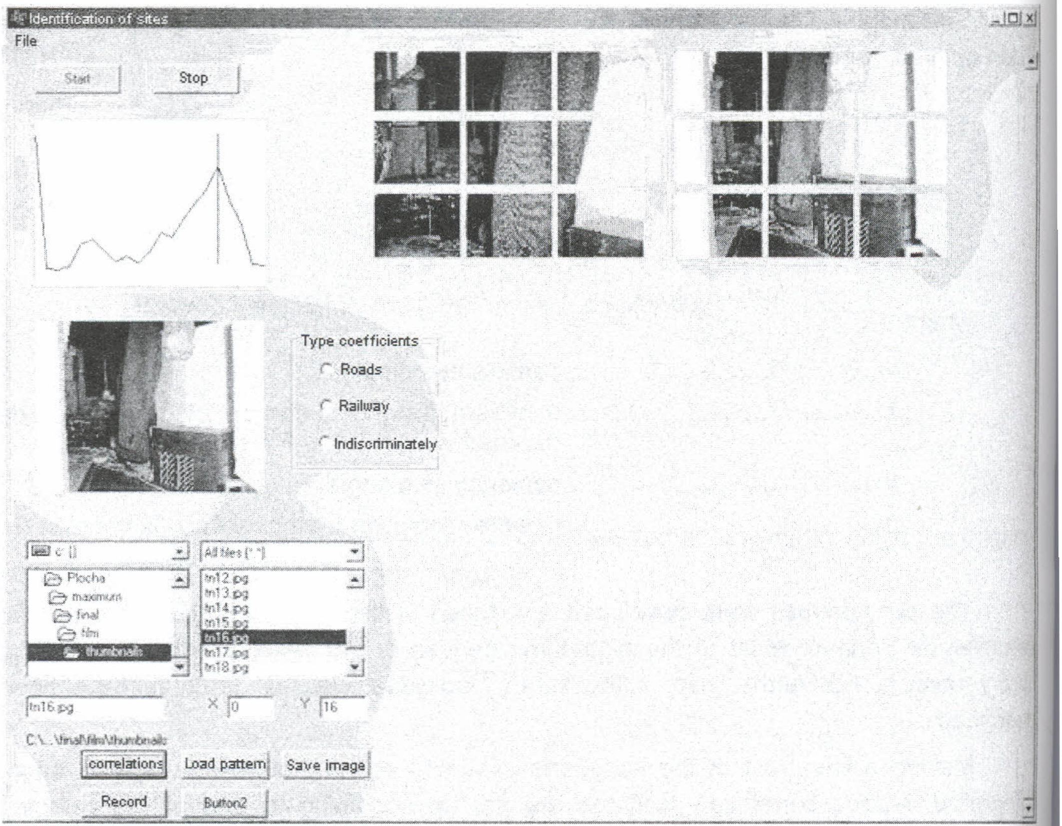
The programmes were developed and tuned in the environment of C++ Builder compiler by Borland. Most of the algorithms derived in this paper are included in the library mycv.h. For further use of the library, individual classes´ features have been described.

Making a linear list of the images is assumed at the first vehicle drive-through. Integrated images correlation methods are set up according to one of the selected modes: at the switch off mode, 1/9 is automatically set up (they are all equal), a road vehicle sets the middle column coefficients to zero and other to 1/6, a rail vehicle sets up the field coefficients to 1b and 3b and other to 0.9/7.

Primary coordinates resources have not been sought for, because the character of the whole tasks demonstrates that e.g. GPS signal may not be at disposal providing that the vehicle is operated in a confined space, such as a warehouse, underground tunnel, cave or, in the future, on other planet surface. Therefore the image numbers are saved as coordinates and in a repeated drive it is the position in the line to be considered the coordinate. The use of a different coordinate system for input data is an elementary task.

A linear list is made at the first drive that may be saved on the disc for later use. All the objects are saved including the coordinates, medium values and credibility coefficients.

An application has been made for the purpose of demonstration by means of this library that shows the required tasks. The application downloads the images sequence by a regular web camera, and the images serial numbers are saved as the coordinates. In the next stage, the coordinate of the current location is found according to an actually scanned image and based on the previously saved sequence.



tn16.jpg

X 0

Y 16

C:\...\final\film\thumbnails

Fig. 8 *Position identification*

The coordinates are displayed on a panel. The panel describes other information, such as the course of the correlation coefficients with the environment of the found image and the controlling elements for the scanning start and for saving of the sample sequence of images.

3. Results

The camera deviation is shown in the left or right image shift. The discordance between the image and its computer sample reduces the correlation coefficient. Detection is easy: we simply move the image to the left or to the right and perform another correlation. Performing of the correlation coefficients comparison helps us to determine whether there has been any lateral shift. When using this method, we have to perform at least two more correlations. The requirement of a parallel images processing is evident.

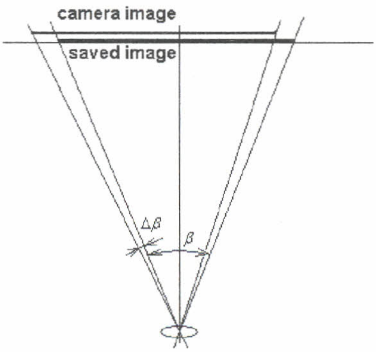


Fig. 9 *Lateral camera deviation*

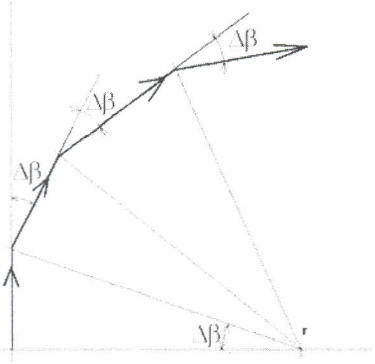


Fig. 10 *Vehicle with a camera approaching a bend*

For the drive of a camera equipped vehicle at a constant speed into a bend with a constant radius, the following applies:

$$\Delta y = r.\sin \Delta\beta \tag{8}$$

$$r = \frac{\Delta y}{\sin \Delta\beta} \tag{9}$$

This shows that the lateral deviation determination is of a key importance for a camera based navigation. The finding that the information about lateral deviation is insufficient, it is the determination of the turning radius that enables to control the driving direction. Another important finding presents the fact that the vehicle reacts with a certain delay. Firstly, a correlation difference must happen to determine the lateral deviation, only then the vehicle is able to react and to continue driving straight on up to the next image position, at least.

The situation is similar in finishing the bend: the vehicle turns at the time when the navigation is set up to the straight direction. Providing that optional filtration by multiple images is used, the delay may be even longer. Such deviation from the original route may not often be acceptable and must be corrected. Possible corrections are carried out by making image samples. First option is to deviate the camera already in making the primary record depending on driving, that is, before the vehicle starts turning. Second option is to adjust the primary record prior to its use.

The following chart presenting a digital image shift is carried out to detect lateral deviation:

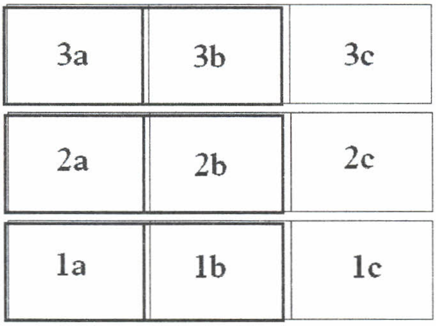


Fig. 11 *Image shift for lateral deviation detection*

4. Contribution to the science and practice

It was required that the system was able, based on current image and the images saved in the past drive, to determine the coordinates of current position and derive the appropriate algorithms enabling the user to gain the vehicle coordinates by comparing optical camera images with previously saved coordinates.

The objective has been achieved. The algorithms derived in this paper have been employed on a small vehicle model (made of a children brick box) equipped with a web camera (Herkules de Luxe). Regarding the requirements for bitmaps processing, the vehicle can only move slowly to enable the computer to process the camera images. The speed increase is a matter of the processor equipment: it may be reached by increasing its speed or by a parallel operation of several processors. Both options are feasible, which is referred to in the paper.

The algorithms have been programmed in language C++ development C++ Builder. It is important to notify that the compiler works very effectively and manual compilation of the programmes e.g. into the language Assembler will not result in any significant acceleration of the resulting codes.

The above mentioned facts prove that the set objective has been achieved. Previously non-published theories and algorithms were developed as by-products of the research.

Submitted: 26.04.2009

References

1. SONKA M., HLAVAC V. *Image Processing, Analysis, and Machine Vision Roger Boyle*, PWS Boston 1999, 2nd edition; ISBN 0-534-95393-X.
2. PRATT W. *Digital image processing* (3ed., Wiley, 2001)(T)(ISBN 0471374075)(738s).

Resumé

POUŽITÍ OPTICKÉ KAMERY PRO URČENÍ SOUŘADNIC VOZIDLA

Josef ŠROLL

Cílem bylo najít vhodné algoritmy, které umožní nalézt souřadnice vozidla na základě obrázků z optické kamery. Předpokládá se, že vozidlo projede nejprve stanovenou trasu a vytvoří posloupnost obrázků z optické kamery, které budou doplněny příslušnými souřadnicemi. Při opakovaných průjezdech již systém porovnává obrázky z optické kamery s obrázky uloženými při prvním průjezdu. Na základě tohoto porovnání určí souřadnice vozidla, kde se právě nachází. Zadaná úloha se ukázala mnohem náročnější, než se původně předpokládalo. Této problematice se systematicky patrně nikdo nevěnuje a proto se nepodařilo nalézt potřebnou podporu. Naděje, vkládané do knihovny programů openCV se nenaplnily. Z toho důvodu bylo nutno pro splnění původního cíle řešit řadu dalších problémů. To však dalo vzniknout novým algoritmům, které dosud nebyly řešeny.

Zadaného cíle bylo dosaženo. Algoritmy, odvozené v této práci byly zprovozněny na malém modelu vozidla (realizovaného z dětské stavebnice) s webovou kamerou (Herkules de Luxe). Vzhledem k velkým nárokům na procesor zpracovávající bitové mapy se vozidlo může pohybovat jen pomalu, aby počítač dokázal včas zpracovat obrazy z kamery. Zvýšení rychlosti je však záležitostí už jen síly procesorového vybavení, buď řádovým zvýšením jeho rychlosti, nebo paralelním chodem více procesorů, což je možné, na což je v této práci na příslušných místech poukazováno.

Aplikace byla naprogramována v jazyce C++. Za tímto účelem byl použit překladač C++ Builder firmy Borland. Standardní knihovny tohoto překladače již obsahují konverzní programy pro převod obrazových formátu JPEG na bitové mapy.

Summary

ORIENTATION OF VEHICLES USING A CAMERA

Josef ŠROLL

The aim was to find suitable algorithms that help to find a vehicle coordinates based on the optical camera images. It is assumed that the vehicle firstly drives through a determined route creating a sequence of optical camera images that will be complemented by relevant coordinates. At repeated drives-through, the system already compares the optical camera images with those saved at the first drive-through. Based on this comparison, current vehicle coordinates can be determined. The task appeared to be much more demanding then it looked in the beginning. This problem has probably not been dealt with, therefore it was impossible to find the required support. The originally expected openCV programme libraries assistance has not been helpful. For this reason, it was necessary to solve numerous other problems to be able to complete the original task. However, new algorithms were found, that have not been solved so far.

The set goal has been reached. The algorithms derived in this paper were launched into operation at a small vehicle model (using a children building box) with a web camera (Herkules de Luxe). Regarding extensive cost of a bit-maps processor, the vehicle can only move quite slowly to enable the computer to process the camera images timely. Speed increase is only a matter of the processor equipment, either by its speed increase or by a parallel run of more processors. The later solution is feasible, which is referred to in the paper.

The application was programmed in C++ language. The Borland C++ Builder compiler was used for this purpose. Standard libraries of the compiler are already equipped with programmes for JPEG images conversion into bit-maps.

Zusammenfassung

DER BESTIMMUNG DER KOORDINATEN DES FAHRZEUGS MIT EINGEBAUTEN OPTISCHEN KAMERA

Josef ŠROLL

Dieser Artikel beschäftigt sich mit dem Einsatz der optischen Kamera zur Festsetzung der Koordinaten von dem bewegten Fahrzeug. Man nimmt die Gewinnung der Serie von Aufnahmen bei der ersten Fahrzeugsdurchfahrt durch unbekannte Gelände an. Bei der Wiederholungsdurchfahrt orientiert sich das Fahrzeug schon nach den Bildern, die von der im Fahrzeug eingebauten Kamera gewonnen wurden. Die Bilder aus Kamera werden auf 9 Teilpartien geteilt und mit ihren Bildermustern aus der ersten Fahrt korreliert. Die Gewichtskoeffiziente der einzelnen Korrelationen für die Teile des Verkehrswegbildes wurden durch die Bearbeitung der grossen Kollektion von Strassen-und Eisenbahnaufnahmen mit der im Fahrzeug untergebrachten Kamera festgelegt.