17

Statistical Video Based Control of Mobile Robots

Krzysztof Okarma and Piotr Lech
West Pomeranian University of Technology, Szczecin
Poland

1. Introduction

A typical approach to the control of mobile robots is based on the analysis of signals from various kinds of sensors (e.g. infra-red or ultrasound). Another type of input data used for motion control of robots can be video signals acquired by the cameras. In such case many algorithms can be utilised e.g. SLAM (Simultaneous Localization and Mapping) and its modifications, being still an active field of research (Kalyan et al., 2010), which require usually the full analysis of the images (or video frames) being the input data (Se et al., 2001; 2002), in some applications using also additional operations such as Principal Component Analysis (Tamimi & Zell, 2004). Similar problems may also occur in the robot localisation and map building processes (Hähnel et al., 2003).
An interesting alternative can be the application of the fast image analysis techniques based on the statistical experiments. Presented statistical approach is related to the first step of such control systems, which can be considered as the pre-processing technique, reducing the amount of data for further analysis with similar control accuracy. Discussed methods can be utilised for many control algorithms based on the image and video data e.g. the proportional steering of line following robots (Cowan et al., 2003) considered as a representative example in further discussions. Nevertheless, some other applications of the presented approach are also possible e.g. fast object classification based on the shape analysis (Okarma & Lech, 2009) or motion detection, as well as fast reduced-reference image quality estimation (Okarma & Lech, 2008b).

2. Reduction of power consumption by mobile robots

Considering the autonomous mobile robots, one of their most important parameters, corresponding directly to their working properties, is the power consumption related to the maximum possible working time. Limited energy resources are often the main element reducing the practical applicability of many such constructions regardless of many modern energy sources which can be used such as e.g. solar based solutions. The limited capacity of the installed batteries reduces the range and capabilities of these devices and each of the possible attempts to optimise energy consumption usually affects the mobility and efficiency of the robots.
It is possible to reduce the consumed energy by finding the shortest path to the target (robot's drive) or conducting the optimisation of the amount of processed data by the control algorithm. Since the computational cost of mapping algorithms, searching for the optimal path, finding robot's own location, visual SLAM algorithms etc., based on the image analysis is strongly dependent on the resolution and representation of the analysed image, a significant "smart" decrease of the amount of processed data seems to be an interesting direction of

research in this area. Reducing the computational load of the processor the overall demand for the electrical energy decreases, so it can be assumed that limiting the amount of processed data in the system, energy savings can be obtained, despite using a non-optimal (longer) motion path. For the video based control algorithms such restriction can be achieved simply by reducing the image resolution or reducing the frame rate of the acquired image sequence used for he robot's motion control. Nevertheless, such decrease of the amount of analysed data may lead to some steering errors, as illustrated in Fig. 1, where the optimal shortest path cannot be used due to the presence of some obstacles. Each of the indicated alternative paths has a different length related to the different energy necessary for the robot to reach the target point.

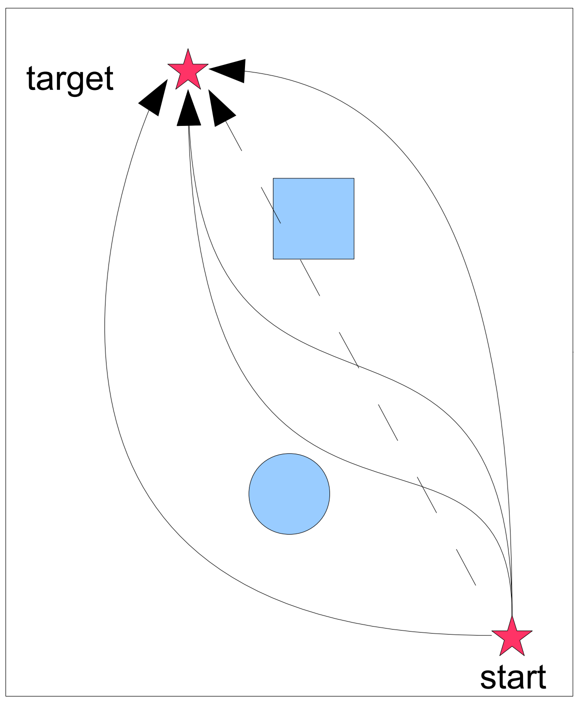


Fig. 1. Illustration of the problem of the optimal path searching in the presence of obstacles

The optimal robot's path in the known environment can be chosen using the machine vision approach with motion detection and robot's motion parameters estimation, using the general algorithm presented in Fig. 2 such that the total energy consumption, related both to the robot's drive and the control part dependent mainly on the processing of the vision data, is minimum. Conducting the experimental verification of the idea for the mobile robot based on a simple netbook platform using the joystick port for controlling the drives, for the testing area of 6 square meters illustrated in Fig. 1, the maximum power consumption is about 5.2 A (42.5% by the drive part and 57.5% by the control part of the system). The length of the optimum straight path, assuming no obstacles on the scene, is equal to 2 meters so all the alternative trajectories are longer.
The experiments conducted in the environment described above has been related to the decrease of the resolution of the image (obtained results are presented in Table 1) and the frame rate for the fixed resolution of 320×240 pixels (results shown in Table 2). For the reduction to 8 frames per second the vision based control algorithm could not find any appropriate path.

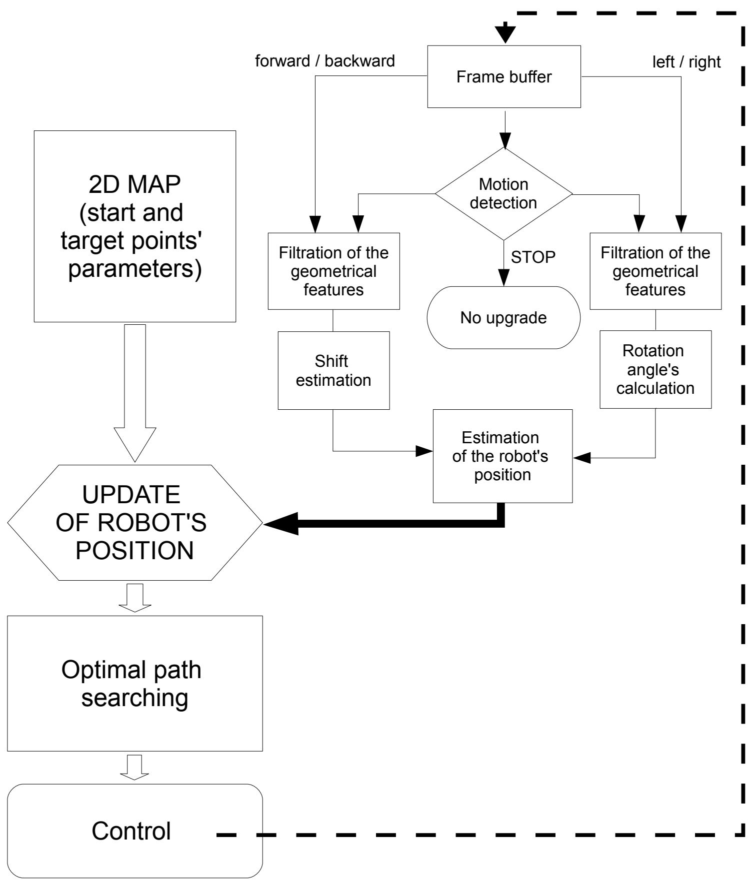


Fig. 2. Illustration of the general vision based control algorithm of the mobile robot

Resolution in pixels	Length of the path [m]	Total energy consumption [kJ]
640 × 480	2.15	5.022
352 × 288	2.39	4.519
320 × 240	2.69	4.268
176 × 144	2.80	4.168
160 × 120	2.95	4.369

Table 1. Path length and power consumption of the mobile robot used in the experiments dependent on the image resolution

Frame rate [fps]	Length of the path [m]	Total energy consumption [kJ]
25	2.39	4.519
21	2.44	4.432
16	2.66	4.332
12	3.23	4.668
10	3.33	4.798

Table 2. Path length and power consumption of the mobile robot used in the experiments dependent on the frame rate of the analysed video sequence

Analysing the results presented in Tables 1 and 2 a significant influence of the amount of the analysed video data on the obtained results and the power consumption can be noticed. The decrease of the resolution as well as the reduction of the video frame rate causes the increase of the path's length but due to the reduction of the processor's computational load the total energy consumption may decrease. Nevertheless, the decrease of the amount of the processed data should be balanced with the control correctness in order to prevent an excessive extension of the robot's path.

Since the simple decrease of the resolution may cause significant control errors as discussed above, a "smart" decrease of the number of analysed pixels, which does not cause large control errors should be used for such purpose much more efficiently. In the proposed statistical approach the main decrease of the amount of data is related to the random draw of a specified number of pixels for further analysis using the modified Monte Carlo method. In a typical version of this method (as often used for the fast object's area estimation) the pixels are chosen randomly by an independent drawing of two coordinates (horizontal and vertical), what complicates the error estimation. For the simplification, the additional mapping of pixels to the one-dimensional vector can be used, what is equivalent to the data transmission e.g. as an uncompressed stream. The additional reduction takes place during the binarization performed for the drawn pixels.

3. The Monte Carlo method for the fast estimation of the objects' geometrical properties

3.1 The area estimation

Conducting a statistical experiment based on the Monte Carlo method, an efficient method of estimation of the number of pixels in the image fulfilling a specified condition can be proposed, which can be used for the motion detection purposes as well as for the area estimation of objects together with their basic geometrical features. Instead of counting all the pixels from a high resolution image, the reduction of the number of pixels used for calculations can be achieved by using a pseudo-random generator of uniform distribution. Due to that the performance of the algorithm can be significantly increased. The logical condition which has to be fulfilled can be defined for a greyscale image as well as for the colour one, similarly to the chroma keying commonly used in television. leading to the binary image, which can be further processed. In order to prevent using two independent pseudo-random generators it is assumed that the numbered samples taken from the binary image are stored in the one-dimensional vector, where "1" denotes the black pixels and "0" stands for the white ones representing the background (or reversely). Then a single element from the vector is drawn, which is returned, so each random choice is conducted independently. For simplifying the further theoretical analysis the presence of a single moving dark object (Ob) in the scene (Sc) with a light background can be assumed.

In order to determine the object's geometrical features as well as some motion parameters, such as direction and velocity, the random choice of pixels cannot be performed using the whole image due to the integrating properties of the Monte Carlo method. For preventing possible errors caused by such integration the scene can be divided into smaller and smaller squares, so the K_N squares from N elements of the scene would represent the object. In such case the probability of choosing the point on the object's surface (assuming the generator's uniform distribution) is equal to:

$$p = \frac{K_N}{N} \tag{1}$$

For the infinite number of samples the reduction of the sampling distance takes place and the probability of choosing the point representing the object on the image can be expressed as:

$$p = \lim_{N \to \infty} \frac{K_N}{N} = \frac{A_{Ob}}{A_{Sc}} \tag{2}$$

so the estimated area of the object is:

$$A_{Ob} \approx K_N \cdot \frac{A_{Sc}}{N} \tag{3}$$

Since the total area of the scene is equal to $A_{Sc} = N \cdot k_x \cdot k_y$, the estimated object's area is equal to $A_{Ob} \approx K_N \cdot k_x \cdot k_y$, where k_x and k_y are the scale factors for horizontal and vertical coordinates respectively, equivalent to the number of samples per unit.

Considering the above analysis, the probability of choosing the point belonging to the object can be used in the proposed algorithm with a reduced number of analysed samples instead of the full image analysis using a statistical experiment based on the Monte Carlo method. This method originates directly from the law of large numbers, because the sequence of successive approximations of the estimated value is convergent to the sought solution and the distance of the actual value after performing the specified number of statistical tests to the solution can be determined using the central limit theorem.

After the binarization the luminance (or colour) samples represented as "ones" or "zeros" corresponding to the allowed values of the specified logical condition, which are stored in one-dimensional vector are chosen randomly. For a single draw a random variable X_i of the two-way distribution is obtained:

$$X_i = \begin{cases} 1 & \text{for black samples} \\ 0 & \text{for white samples,} \end{cases} \tag{4}$$

leading to the following probability expressions:

$$P(X_i = 1) = p \qquad P(X_i = 0) = q \tag{5}$$

where $p + q = 1$, $E(X_i) = p$, $V(X_i) = p \cdot q$.

An important element of the method is the proper choice of the logical condition for the binarization allowing a proper separation of the object from the background. Such choice depends on the specific application, for example for the light based vehicles' tracking purposes the chroma keying based on the CIELAB colour model can be appropriate (Mazurek & Okarma, 2006).

For n independent draws the variable Y_n is obtained:

$$Y_n = \frac{1}{n} \cdot \sum_{i=1}^{n} X_i \tag{6}$$

According to the Lindberg-Levy's theorem, the distribution of Y_n tends to the normal distribution $N(m_y, \sigma_y)$ if $n \to \infty$. Since the expected value and variance of Y_n are equal to $E(Y_n) = p$ and $V(Y_n) = \frac{p \cdot q}{n}$ respectively, the distrubution of the random value Y_n is normal with the following parameters:

$$m_y = p \tag{7}$$

$$\sigma_y = \sqrt{\frac{p \cdot q}{n}} \tag{8}$$

Considering the asymptotic normal distribution $N(p, \sqrt{p \cdot q/n})$ it can be stated that the central limit theorem is fulfilled for the variable Y_n.
Substituting:

$$U_n = \frac{Y_n - m_y}{\sigma_y} \tag{9}$$

obtained normal distribution can be standardized towards the standard normal distribution $N(0, 1)$.
In the interval estimation method the following formula is used:

$$p(|U_n| \leq \alpha) = 1 - \alpha \tag{10}$$

Assuming the interval:

$$|U_n| \leq u_\alpha \tag{11}$$

considering also the formulas (3), (7), (8) and (9), the following expression can be achieved:

$$\left| Y_n - \frac{K_N}{N} \right| \leq \varepsilon_\alpha \tag{12}$$

where

$$\varepsilon_\alpha = \frac{u_\alpha}{\sqrt{n}} \cdot \sqrt{\frac{K_N}{N} \cdot \left(1 - \frac{K_N}{N}\right)} \tag{13}$$

The probability estimator p (eq. 1), for k elements from n draws, fulfilling the specified logical condition used for the definition of X_i (eq. 4) and representing the number of the drawn pixels representing the object, can be expressed as:

$$\hat{p} = \frac{k}{n} = \frac{1}{n} \cdot \sum_{i=1}^{n} X_i = Y_n \tag{14}$$

and the object's area estimator as:

$$\hat{A}_{Ob} = \hat{p} \cdot A_{Sc} = \frac{k}{n} \cdot A_{Sc} \tag{15}$$

Using the equations (12) and (15) the obtained formula describing the interval estimation for the object's area is:

$$\left| \frac{\hat{A}_{Ob}}{A_{Sc}} - \frac{K_N}{N} \right| \leq \varepsilon_\alpha \tag{16}$$

where ε_α is specified by the equation (13).
It is worth to notice that all the above considerations are correct only for a random number generator with the uniform distribution, which should have as good statistical properties as possible. The discussed algorithm are identical to the method of area estimation of the 2-D object's (expressed in pixels). Nevertheless, the applicability of such approach is limited by the integrating character of the method so an additional modification based on the block approach is necessary, as mentioned earlier.

Such prepared array, considered as a reduced resolution "greyscale" image, can be directly utilised in some typical control algorithms. Further decrease of the computational cost can be obtained by the additional binarization of this image leading to the ultra-low resolution binary image considered in further discussions, where each of the blocks can be classified as a representative of an object or the background. For the line following robot control purposes the objects is equivalent to the path. The control accuracy of the mobile robots corresponds to the quality loss of the data present in this image. Such quality can be treated as the quality of the binary image assuming the knowledge of the reference image (without any distortions). Unfortunately, most of the image quality assessment methods, even the most recent ones, can be successfully applied only for greyscale or colour images and the specific character of the binary images is not respected by them. For this reason only the methods designed exclusively for the binary images can be used. Such binary image quality assessment methods can be used as the optimisation criteria for the proposed algorithm. In the result the proper choice of the block size, binarization threshold and the relative number of randomly drawn pixels can be chosen.

Depending on the logical condition used for the statistical analysis (the construction of the 1-D binary vector), the algorithm can be implemented using various colour spaces (or only some chosen channels), with the possibility of utilising independent logical conditions for each channel (chroma keying).

3.2 The Monte Carlo based motion detection

The real-time navigation of an autonomous mobile robot based on the machine vision algorithms requires a fast processing of images acquired by the camera (or cameras) mounted on the robot. In some more sophisticated systems the data fusion based on the additional informations acquired by some other sensors, such as PIR sensors, ultrasound detectors, a radar technique equipment or optical barriers, is also possible. Most of such sensors are responsible for the motion detection, especially if the presence of some moving obstacles in the robot's surrounding is assumed. The vision based sensors can be divided into three main groups: analogue (i.e. comparing the luminance with a given threshold value), digital (obtained by the sampling and quantization of acquired analogue signal) and the digital ones acquired directly from digital video cameras. There are also some restrictions of typical alternative solutions, e.g. passive infrared detectors are sensitive to temperature changes and they are useless for the motion detection of objects with the same temperature as the background. Some more robust solutions, such as ultrasound and radar detectors, are usually active (energy emitting) and require an additional power supply. The application of a vision based motion detection procedure utilising the Monte Carlo method for the mobile robots control purposes is caused mainly by the fact that no additional hardware is necessary, since the images used by the motion detector are acquired by the same camera as in the classification procedure.

Assuming the scene with the constant light conditions without any moving objects, it can be characterised by a constant value defined as the number of black (or white) pixels of the respective binary image, which usually represent the object visible on the image obtained from the camera. The algorithm of the motion detection utilises the rapid changes of the number of such pixels (treated as "ones") when an object moves respectively to the light sources. Such change is caused by the dependence of the image pixel's luminance (and colour) on the angle between normal vector to its surface and the direction of the light ray passing the pixel, as well as the influence of the surface's reflection coefficient, shape, roughness etc. Real objects usually have a heterogeneous structure of their surface so the number of analysed points changes dynamically (assuming constant threshold or chroma keying range). For slow changes of the light conditions, especially for outside environment, the binarization threshold can be updated. It it worth to notice that in the proposed approach the necessity of the storage of the single value only (the number of pixels corresponding to the "ones") allows the reduction of the required system's operating memory size.

A typical well known video based algorithm of motion detection is based on the comparison of two neighbouring frames of the video signal but it requires the storage of the images used during the comparison so relatively large amount of memory is needed. Some more robust algorithms of background estimation, directly related to motion detection, usually based on moving average or median filter, are also more computationally demanding and require the analysis of several video frames (Cucchiara et al., 2003; Piccardi, 2004). Even a comparison of all the pixels from two video frames can be treated as a time consuming operation for an autonomous mobile robot with limited computational performance and relatively small amount of operating memory, so in the real-time applications some high performance processing units would be required. Instead of them the statistical Monte Carlo approach described above can be used for the reduction of the computational complexity, where only some randomly chosen pixels are taken into account. Therefore the comparison of full frames is reduced to the comparison of only two values representing the number of "ones" in each frame. Any rapid change of that value is interpreted as the presence of a moving object.

A slightly modified method based on the additional division of the image into smaller blocks can be applied for a more robust detection. In the presence of moving objects the changes can be detected only in some of obtained blocks so the integrating character of the Monte Carlo method is eliminated.

The fast motion detection based on the comparison of two values representing the estimated areas of the object in the two consecutive binary frames is based on the following formula:

$$\hat{A}_{Ob}(i+\delta) - \hat{A}_{Ob}(i) > threshold \tag{17}$$

where $threshold >> \varepsilon_\alpha$ and $\delta \geq 1$ denotes the shift between both analysed frames.

In the proposed block based approach the application of the formula (17) for each of the $r \times r$ pixels blocks allows proper motion detection, using the array of the Monte Carlo motion detectors, also in the situation when the motion of the mobile robot causes the relative motion of the objects observed by the integrated camera. For the perspectively looking camera, typical for such applications, even if the objects do not change their size on the image, the changes can be observed among the blocks. The objects moving towards the camera always cause the increase of their estimated area in the image used by the motion detector. The estimation of the geometrical parameters of the objects, often necessary for the proper navigation of the controlled robot, can be performed for the frame with the maximum value of the estimated area occupied by the object on the image with eliminated background, assuming that robot is not moving directly towards the object. If the size of the object increases in consecutive frames, such estimation should be performed immediately preventing possible collision with the object.

3.3 The estimation of geometrical parameters

The extraction of some geometrical parameters, such as perimeters or diameters, using the discussed approach is also possible. For this purpose the analysed binary image has to be divided into $T \times S$ blocks of $r \times r$ pixels each using a square grid. Then the area of each object's fragment in the elementary square elements (blocks) is calculated and the results of such estimation can be stored in the array P containing $T \times S$ elements.

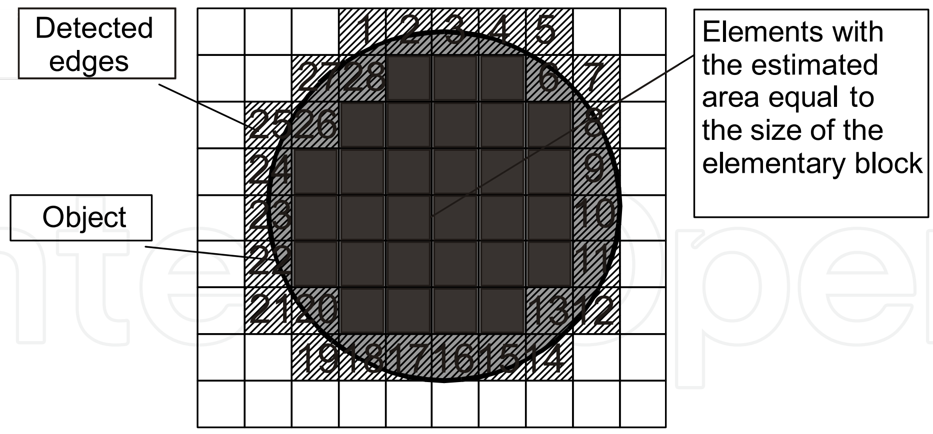


Fig. 3. The idea of the block based edge estimation

The simplest approach to the edge detection, which is necessary for further processing, is based on the array *P*. For this purpose the second array *K* with the same size is created and its elements have the following values:

- zero if the corresponding element's value in the array *P* is equal to zero (background),
- zero if the corresponding element's value in the array *P* is equal to the size of the elementary block (all pixels represent the inner part of the object) and none of its neighbouring blocks (using the 8-directional neighbourhood) has the zero value,
- one for the others (representing the edge).

The idea of the edge detection is illustrated in Fig. 3
The obtained array *K* is the projection of edges detected from the source image, so the number of its non-zero elements represents the estimated value of object's perimeter expressed in squares of $r \times r$ pixels. For a better estimation the number of square elements can be increased (smaller values of the parameter r) and then the previously discussed operation should be repeated. In such case the analysis of the blocks in the array *P* with the zero values (representing the inner parts of the object) is not necessary, so the further analysis can be conducted only for the significantly reduced number of indexed elements classified as the edge in the previous step. Such alternative edge detection method based on the two-step algorithm with the same initial step of the algorithm and another pass of the previous algorithm conducted as the second step with the edge detection calculated using the array *K* (the edge obtained in the first step is treated as the new object without fill) allows more accurate estimation of the edge with slight increase of the computational complexity. All the pixels with the zero values should be replaced from the array *K* and using such corrected image the two curves are obtained as the result (assuming the scale factor of the block greater than $t = 2$ - at least 3×3 elements). As the estimate of the edge the middle line between the two obtained curves can be used, providing slightly lower accuracy but also a reduced computational cost. The estimate of the total object's area can also be determined with similarly increased accuracy as the sum of the values in the array *P*. The limit accuracy of the algorithm is determined by the size of the elementary block equal to 1 pixel what is equivalent to the well-known convolution edge detection filters.
Geometrical parameters of the objects, which can be estimated using the modified Monte Carlo method, can be divided into the two major groups: local (such as mean diameter or the

average area) and global (e.g. the number of objects in the specified area). The estimation of the relative global parameters is not necessary if only a single object is present on the analysed fragment of the image. The most useful parameters used for image analysis, which can be also used for robot's control purposes, are insensitive to image deformations which can occur during the acquisition, as well as to some typical geometrical transformations. In such sense the usefulness of the simplest parameters such as area and perimeter is limited. Nevertheless, the majority of more advanced parameters (such as moments) can be usually determined using the previously described ones, similarly as some motion parameters e.g. direction or velocity (Okarma & Lech, 2008a).

4. Application for the line following robots

One of the most important tasks of the control systems designed for mobile robots is line tracking. Its typical optical implementation is based on a line of sensors receiving information about the position of the traced line, usually located underneath the robot. Such line sensor is built from a specified number of cells limiting the resolution of the optical system. Another important parameter of such systems is the distance from the sensors to the centre of steering, responsible for the maximum possible speed of the properly controlled robot. The smoothness of the robot's motion is also dependent on the spacing between the cells forming the line sensor.

A significant disadvantage of such optical systems is relatively low resolution of the tracking system, which can be increases using some other vision based systems with wide possibilities of analysing data acquired from the front Dupuis & Parizeau (2006); Rahman et al. (2005). Nevertheless the analysis of the full acquired image even with low resolution is usually computationally demanding and time consuming, especially in the presence of some obstacles, some line intersections etc., assuming also varying lighting conditions. Another serious drawback of using line of sensors is the short time for the reaction limited by the distance between the sensors and the steering centre (or wheels). The proposed approach, based on the fast image analysis using the Monte Carlo method, preserves the main advantages of the vision systems allowing the reduction of the amount of processed data. Considering its application for the control of the mobile robots, the camera moves relatively to the static scene, differently than in the primary version of the algorithm, but the working properties of the method are similar.

In order to filter the undesired data, usually related to some contaminations which should not be used for the robot control purposes, the binarization threshold should be properly set. In the conducted experiments the additional "cut-off" value has been set as minimum 20% black pixels possibly representing the object within the block in order to avoid the influence of the small artifacts, especially close to the region of interest. Finally, the simplified binary representation of the image is obtained where such artifacts (the elements which in fact do not represent the followed line) have been removed during the "cut-off" operation illustrated in Fig. 4 where the original binary image and the intermediate result obtained by the Monte Carlo method are also presented.

4.1 A simple proportional control algorithm for controlling a line following robot

The mobile robot control process can be based on the popular proportional control approach. It is typically used for controlling the robots with sensors based on the infra-red receivers grouped into the line containing a specified number of the infra-red cells. In the proposed method the differential steering signals for the motors are obtained using the first (the lowest)

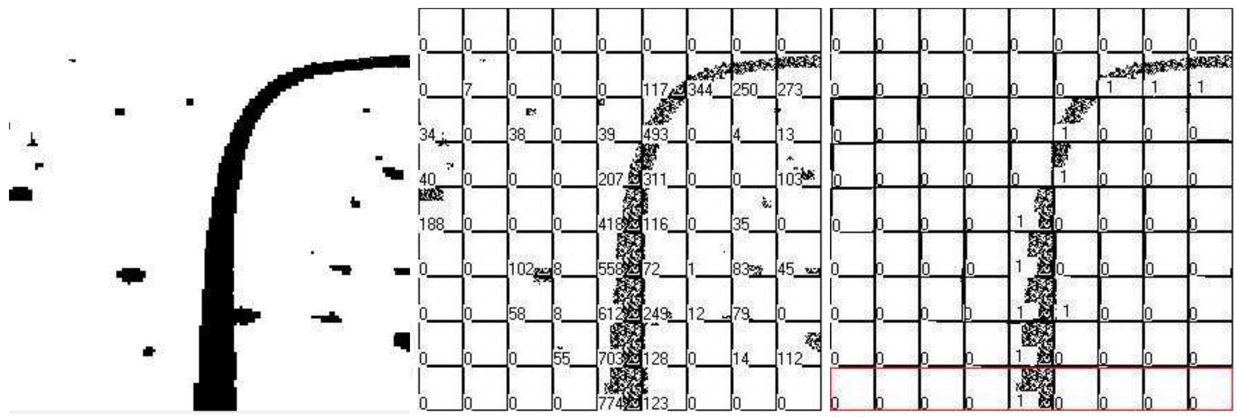


Fig. 4. The idea of the "cut-off" operation

row of the obtained simplified binary image which is equivalent to the typical line of the infra-red sensors.

Nevertheless, the machine vision algorithm can be disturbed by some artifacts caused by the errors during the preprocessing step (e.g. insufficient filtration of some undesired elements on the image). An optimal situation would take place for a single "one" in the lowest row of the image representing the tracked line.

In the simplified version of the algorithm no line crossings can be assumed. The reduction of the possible disturbances can be achieved using the following additional operations:

- limitation of the line thickness by the proper choice of the binarization threshold (lines containing more than a single "one" can be ignored assuming no line crossings),
- orphan blocks removal (applicable for all blocks with the value "one" without any neighbouring "ones"),
- ignoring the empty lines (without any "ones").

4.2 Extension of the robot control by vision based prediction

The control algorithm of the robot should point towards such a position that only a single block of the bottom image row will be filled (representing the line). In the case when two neighbouring blocks represent the line the choice of the block should be based on the simple analysis of the current trend, assuming that the turn angles of the line match the motion and control possibilities of the controlled robot.

The control algorithm for a single frame can be described as follows (Okarma & Lech, 2010):

- binarization of the current frame,
- orphan blocks removal,
- filling the gaps in the detected line using the approximation methods,
- control operation:
  - the detection of the crossing lines
    - if not turning: moving forward with the maximum speed for the symmetrical line-crossing or using the speed control
    - else: turning and using the speed control if the maximum value is not in the middle block
  - speed control: velocity should be proportional to the sum of the values in the middle blocks of each horizontal line (if zero, the minimum speed is set before the turning and the control flag is set to 0).

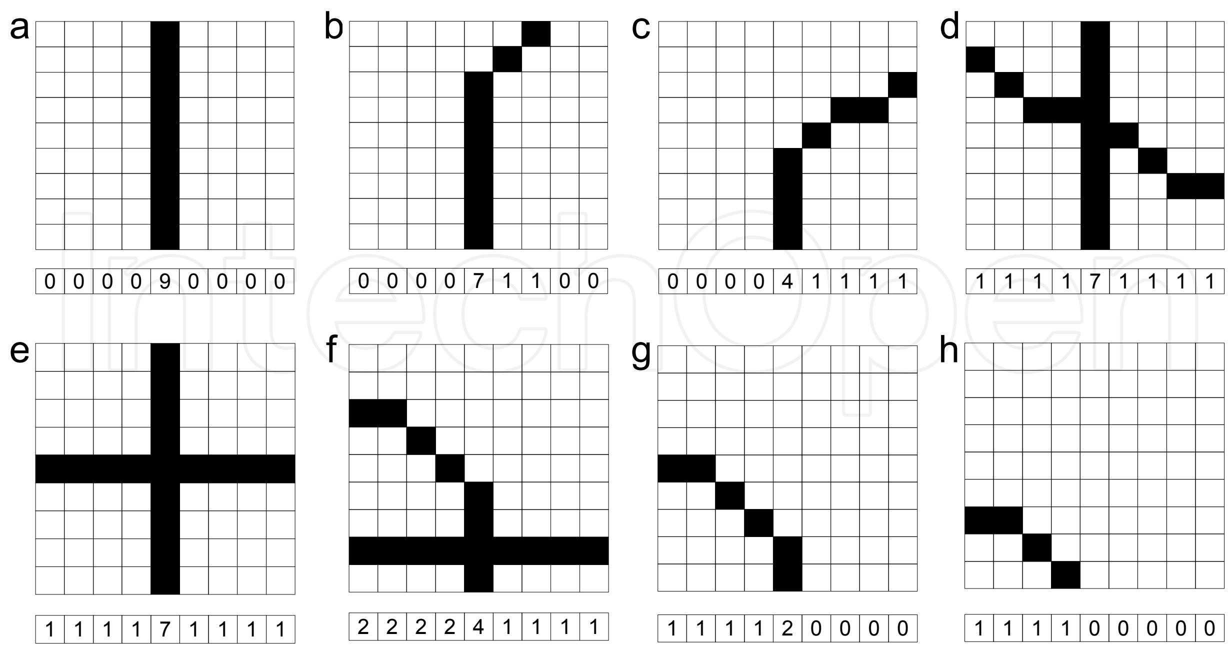


Fig. 5. The illustration of the various configurations of the followed line

The main element of the modified control system is the variable (flag), determined using the previous frames, informing the control system about the current state (e.g. 0 for moving forward or turning, 1 for turning with detected crossing lines).

In the case of detected crossing line the control depends on the symmetry. For the symmetrical line-crossing the robot should move forward with the maximum speed or the additional speed control. If the maximum value is not in the middle block the line-crossing is not symmetrical and the robot should turn left or right and decrease it speed. The speed of the robot should be proportional to the sum of the values in the middle blocks of each horizontal line. If that sum is equal to zero, the minimum speed should be set before the turning. In that case the mentioned above flag should be set to 1. The illustration of the various scenarios is presented in Fig. 5.

One of the typical features of the proportionally controlled systems is the oscillating character of the path for the robot reaching the specified line, causing also the characteristic oscillations of the images acquired from the camera. Nevertheless, the conducted experiments have verified the usefulness of the fast image analysis based on the Monte Carlo method for the efficient controlling of the line following robot in a simplified but very efficient way, especially useful for the real-time applications. The application of the proposed solution for the control of the mobile robot, despite of the simplicity of its control algorithm, is comparable with some typical ones based on the classical approach (proportional steering based on the infra-red sensors).

The main advantage of the proposed method is the possibility of using some prediction algorithms allowing the increase of the robot's dynamic properties and fluent changes of its speed depending on the shape of the line in front of the robot. Another relevant feature is its low computational complexity causing the relatively high processing speed, especially important in the real-time embedded systems.

4.3 Application of the Centre of Gravity

If the robot is located directly over the straight line its central point should be exactly between the two edges obtained using the two-step algorithm of edge detection discussed above. In such situation the sum of the pixel values on its left side should be identical as on its right side. Any significant disturbance can be interpreted as the beginning of a line curve with the assumption that no additional elements, except two line edges, are visible on the image, so the mobile robot should begin turning. An efficient and simple method for controlling the robot which can be used for such purpose is based on the image's Centre of Gravity (Centre of Mass). It can be determined for a grayscale image as:

$$x_c = \frac{\sum_{i,j} i \cdot X_{i,j}}{\sum_{i,j} X_{i,j}} \tag{18}$$

$$y_c = \frac{\sum_{i,j} j \cdot X_{i,j}}{\sum_{i,j} X_{i,j}} \tag{19}$$

where $X_{i,j}$ represents the luminance of the pixel located at position (i, j).
Calculating the coordinates of the Centre of Gravity (CoG), they can be used by the control algorithm as the current target point for the mobile robot. The horizontal coordinate x_c can be used for controlling the turning angle and the vertical y_c for the speed control, assuming the previous filtration of any artifacts from the image.
The approach discussed above can also be implemented using the block based Monte Carlo method using the "cut-off" operation for decrease of the influence of noise, leading to the hybrid control method. For each block the local area estimator $\hat{A}$ is used as the input for the "cut-off" operation leading to the simplified binary representation utilised for steering using the method discussed above. In comparison to the classical Center of Gravity method its implementation using the Monte Carlo approach causes about 8 times reduction of the computation time as has been verified in the conducted experiments.
One of the main modifications proposed in the hybrid method is the switch between the CoG and Monte Carlo based control in order to increase the robustness of the control. For this purpose the sum of each column of the simplified binary image is calculated and stored in an additional vector. If the vector containing the sums of the values, obtained by counted blocks with "ones", has more than one local minimum (e.g. 2 4 0 0 1 6 0 0 1), some additional objects are present on the image. In such situation the calculation of the Centre of Gravity coordinates would be disturbed by a neighbouring object located close to the line, so the motion control should be based on the previous value. Otherwise, the control values determined by the Monte Carlo method can be additionally verified by the CoG calculations.
A more complicated situation may take place for the presence of artifacts together with some discontinuities of the followed line. In this situation the vector containing the sum of the column in the simplified block representation of the image has more than one local maximum. In such case the additional elimination of the blocks which are not directly connected to the followed line (detected as represented by the maximum value in the vector) should be conducted e.g. using the morphological erosion, before the calculations of the Centre of Gravity used for determining the current trend (direction of the line).

5. Simulations in the Simbad environment

In order to verify the working properties of the proposed methods, regardless of some tests conducted using a netbook based prototype, a test application has been created using Java programming language in NetBeans environment allowing the integration of many libraries developed for Java e.g JARToolKit (Geiger et al., 2002) allowing to access the Augmented Reality Toolkit (ARToolKit) functionality via Java and the use of different rendering libraries allows high and low level access. The environment of the mobile robot as well as the robot itself have been modelled using the Simbad simulator (Hugues & Bredeche, 2006), which is available for this platform.

A significant advantage of the simulator is its 3D environment, due to the utilisation of Sun Java3D technology. Nevertheless, an important restrictions of Simbad, in opposite to some other similar products, are the limited working properties for the controlling the real robots die to its internal structure related to the modelling of physical phenomena. However, this results in a simplified and transparent programming source code, so the implementation of the presented algorithms does not cause any considerable problems. The only restriction in the initialisation part of the robot controller is the directing the camera towards the "ceiling", caused by the inability to put the followed line on the ground. Instead of it the line has been located over the robot and the lines in the front of the robot are shown in the top part of the image. The only negative consequence is the reverse transformation of the left and right directions between the robot and the camera. The proportional controlled differential drive robots used in the experiments are typically implemented in Simbad and conducted simulations have confirmed the working properties of the presented steering methods for all control modes with different simulation speed values. The test application used for the simulations is illustrated in Fig . 6.

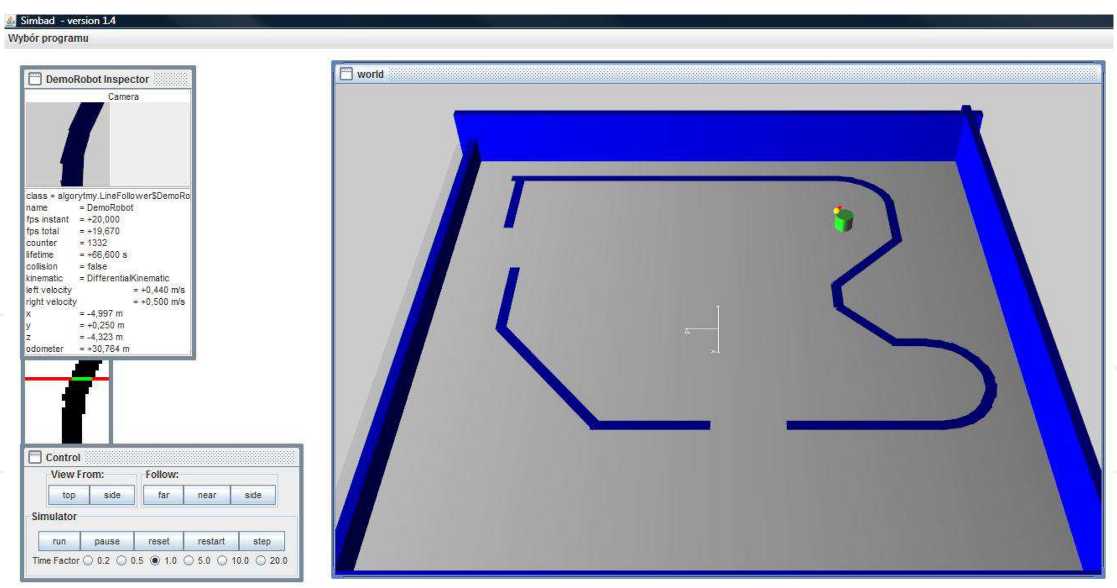


Fig. 6. Illustration of the test application

6. Conclusions and future work

Obtained results allowed the full control of the line following robot (speed, direction and "virtual" line reconstruction). Due to the ability of path prediction, in comparison to the typical line following robots based on the line of sensors mounted under the robot, the "virtual" line reconstruction can be treated as one of the most relevant advantages of the image based motion control. Such reconstruction can also be performed using the ultra-low resolution binary image as the input data. Another possibility, being analysed in our experiments, is the adaptation of the Centre of Gravity method for the real time steering based on the binary image analysis.

The additional robustness of the method can be achieved by the adaptive change of the block size used in the modified Monte Carlo method according to the quality assessment of the binary image and the additional analysis of the line continuity.

Positive results of the simulations performed in Java 3D based Simbad environment allow the proper implementation of the proposed soft computing approach in the experimental video controlled line following robot. Good simulation results, as weel as the low energy consumption by the experimental netbook-based robot, have been also obtained considering the synthetic reduced resolution "grayscale" image constructed using the Monte Carlo method, where the relative areas of the objects in each block have been used instead of the luminance level (after normalisation).

The planned future work is related to the physical application of the proposed approach for some other types of robot control algorithms based on video analysis, e.g. SLAM. Another direction of the future research should be the development of a dedicated binary image quality assessment method, which could be used for the optimisation of the proposed algorithm.

Another interesting direction of the future work is the extension of the proposed method towards direct operations using the colour images. For this purpose a verification of the usefulness of the typical colour models is necessary by means of the chroma keying used for the thresholding in the Monte Carlo method.

7. References

Cowan, N., Shakernia, O., Vidal, R. & Sastry, S. (2003). Vision-based follow-the-leader, *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vol. 2, Las Vegas, Nevada, pp. 1796–1801.

Cucchiara, R., Grana, C., Piccardi, M. & Prati, A. (2003). Detecting moving objects, ghosts, and shadows in video streams, *IEEE Transactions on Pattern Analysis and Machine Intelligence* 25(10): 1337–1342.

Dupuis, J.-F. & Parizeau, M. (2006). Evolving a vision-based line-following robot controller, *Proceedings of the 3rd Canadian Conference on Computer and Robot Vision*, Quebec City, Canada, p. 75.

Geiger, C., Reimann, C., Stocklein, J. & Paelke, V. (2002). JARToolkit - a Java binding for ARToolkit, *Proceedings of the The First IEEE International Workshop on Augmented Reality Toolkit*, Darmstadt, Germany, p. 5.

Hähnel, D., Triebel, R., Burgard, W. & Thrun, S. (2003). Map building with mobile robots in dynamic environments, *International Conference on Robotics and Automation*, Vol. 2, Taipei, Taiwan, pp. 1557–1563.

Hugues, L. & Bredeche, N. (2006). Simbad: An autonomous robot simulation package for education and research, *in* S. Nolfi, G. Baldassarre, R. Calabretta, J. C. T. Hallam, D. Marocco, J.-A. Meyer, O. Miglino & D. Parisi (eds), *SAB*, Vol. 4095 of *Lecture Notes in Computer Science*, Springer, pp. 831–842.

Kalyan, B., Lee, K. & Wijesoma, W. (2010). FISST-SLAM: Finite set statistical approach to simultaneous localization and mapping, *International Journal of Robotics Research* 29(10): 1251–1262.

Mazurek, P. & Okarma, K. (2006). Car-by-light tracking approach based on log-likelihood track-before-detect algorithm, *Proceedings of the 10th International Conference "Computer Systems Aided Science Industry and Transport" TRANSCOMP*, Vol. 2, Zakopane, Poland, pp. 15–20.

Okarma, K. & Lech, P. (2008a). Monte Carlo based algorithm for fast preliminary video analysis, *in* M. Bubak, G. D. van Albada, J. Dongarra & P. M. A. Sloot (eds), *ICCS (1)*, Vol. 5101 of *Lecture Notes in Computer Science*, Springer, pp. 790–799.

Okarma, K. & Lech, P. (2008b). A statistical reduced-reference approach to digital image quality assessment, *in* L. Bolc, J. L. Kulikowski & K. W. Wojciechowski (eds), *ICCVG*, Vol. 5337 of *Lecture Notes in Computer Science*, Springer, pp. 43–54.

Okarma, K. & Lech, P. (2009). Application of Monte Carlo preliminary image analysis and classification method for automatic reservation of parking space, *Machine Graphics and Vision: International Journal* 18(4): 439–452.

Okarma, K. & Lech, P. (2010). A fast image analysis technique for the line tracking robots, *in* L. Rutkowski, R. Scherer, R. Tadeusiewicz, L. A. Zadeh & J. M. Zurada (eds), *ICAISC (2)*, Vol. 6114 of *Lecture Notes in Computer Science*, Springer, pp. 329–336.

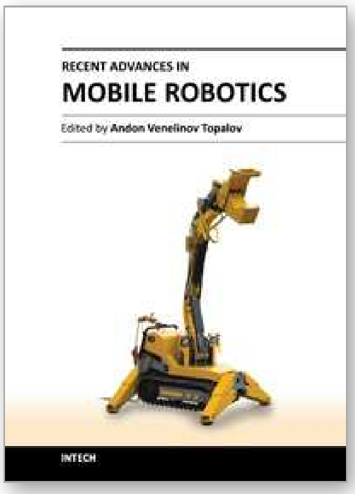
Piccardi, M. (2004). Background subtraction techniques: a review, *Proceedings of the IEEE International Conference on Systems, Man & Cybernetics*, Vol. 4, The Hague, Netherlands, pp. 3099–3104.

Rahman, M., Rahman, M., Haque, A. & Islam, M. (2005). Architecture of the vision system of a line following mobile robot operating in static environment, *Proceedings of the 9th International Multitopic Conference, IEEE INMIC 2005*, Karachi, pp. 1–8.

Se, S., Lowe, D. & Little, J. (2001). Local and global localization for mobile robots using visual landmarks, *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vol. 1, Maui, Hawaii, pp. 414–420.

Se, S., Lowe, D. & Little, J. (2002). Mobile robot localization and mapping with uncertainty using scale-invariant visual landmarks, *International Journal of Robotics Research* 21: 735–758.

Tamimi, H. & Zell, A. (2004). Vision based localization of mobile robots using kernel approaches, *Proceedings of the 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sendai, Japan, pp. 1896–1901.



Recent Advances in Mobile Robotics
Edited by Dr. Andon Topalov

ISBN 978-953-307-909-7
Hard cover, 452 pages
Publisher InTech
Published online 14, December, 2011
Published in print edition December, 2011

Mobile robots are the focus of a great deal of current research in robotics. Mobile robotics is a young, multidisciplinary field involving knowledge from many areas, including electrical, electronic and mechanical engineering, computer, cognitive and social sciences. Being engaged in the design of automated systems, it lies at the intersection of artificial intelligence, computational vision, and robotics. Thanks to the numerous researchers sharing their goals, visions and results within the community, mobile robotics is becoming a very rich and stimulating area. The book Recent Advances in Mobile Robotics addresses the topic by integrating contributions from many researchers around the globe. It emphasizes the computational methods of programming mobile robots, rather than the methods of constructing the hardware. Its content reflects different complementary aspects of theory and practice, which have recently taken place. We believe that it will serve as a valuable handbook to those who work in research and development of mobile robots.