Journal of Achievements in Materials and Manufacturing Engineering
VOLUME 49
ISSUE 2
December
2011

The elements of artificial intelligence and the logic programming in the semi-automatic programming of the robot

K. Foit*
Institute of Engineering Processes Automation and Integrated Manufacturing Systems, Faculty of Mechanical Engineering, Silesian University of Technology, ul. Konarskiego 18a, 44-100 Gliwice, Poland
* Corresponding author: E-mail address: krzysztof.foit@polsl.pl

Received 18.10.2011; published in revised form 01.12.2011

Manufacturing and processing

ABSTRACT

Purpose: The aim of this paper is to show of using some aspects of the artificial intelligence and the logic programming in "programming by expression of will" methods in order to simplify the whole process of programming of the robot.
Design/methodology/approach: The new method uses the "show me the state before and after" approach. The input data have a form of the pictures set: the first set shows the "before" state, while the second one shows the "after" state. These pictures then should be analysed by the computer program and the "skeleton code" for the robot could be created.
Findings: The most important, but also the most error-prone system is the image recognition processor. In the further development of the system it should be taken into consideration, that there is no very effective method for the separation of manipulation objects from the background.
Practical implications: The presented concept of the method of robots programming by the expression of the will (or the intentional programming) is in early phase of development. There are plenty of unsolved problems, which could affect the overall performance and the efficiency of this method. The problem with the highest priority is to create the error-resistant image recognition system, which could work with popular types of cameras.
Originality/value: The presented method is the extension of programming by demonstration method, but there is no need "to teach" the robot – the program is generated almost automatically.
Keywords: Automation engineering processes; Robotics; Mechatronics

Reference to this paper should be given in the following way:
K. Foit, The elements of artificial intelligence and the logic programming in the semi-automatic programming of the robot, Journal of Achievements in Materials and Manufacturing Engineering 49/2 (2011) 383-390.

1. Introduction

Since ancient times, people wanted to create smart tools that could do the work without human intervention. This need is deeply inscribed in the history of robotics, particularly in its origins and the construction of the first automatons.

Over the centuries, people have changed the perception of the role of the robots and automatons in a daily life: starting from the typically entertainment applications, through the assistance in carrying out daily activities related to the human work, to the complete replacement of the human at its workplace. Today, automatons and robots replace human, working in difficult, dangerous to the health and life environment, while ensuring high accuracy and repeatability of the performed operation.

One of the most absorbing problems is programming automata and robots. The most desirable form of the "dialogue" with the machine has been the use of voice commands, like when talking to the other man. For centuries, the programming has been done by proper selection of cams and levers, and the first possible application of certain aspects of human perception appeared in times of electricity, and still being developed today.

Nowadays, most of the industrial robots are programmed using specialized, high-level programming languages. Often the off-line programming systems are used, which are based on a PC class computer equipped with the appropriate software that allows editing the program code and often helps the operator by using creators and standard code blocks. The software also allows simulation of the robot behavior in the workplace. The on-line programming model does not give the same comfort of program editing like the off-line programming. The operator is limited by the functions offered by the robot's operating system. For this reason, often the mixed method is used – so called hybrid programming. Recent studies are conducted in order to eliminate the inconvenience of using a teach pendant to program the robot on-line. The aim is to eliminate (at least partially) the need of using the programming language, introducing instead so-called programming by demonstration, which in general could be defined as programming by "showing" arrangement of individual elements of the program to be performed (positions, tool state etc.), using the alternative techniques, such as computer vision systems, force control, voice commands, etc.

The aim of this paper is to show some aspects of the artificial intelligence and the logic programming in "programming by expression of will" methods in order to simplify the whole process of programming of the robot.

2. History and the present day of programming of the automata

2.1. The Antiquity and The Middle Ages

In the fourth century BC, Aristotle wrote in "Politics":

"For if every instrument could accomplish its own work, obeying or anticipating the will of others, like the statues of Daedalus, or the tripods of Hephaestus, which, says the poet, of their own accord entered the assembly of the Gods; if, in like manner, the shuttle would weave and the plectrum touch the lyre without a hand to guide them, chief workmen would not want servants, nor masters slaves." [1].

This sentence is often seen as the first in the history reference to the robotics. Aristotle describes the intelligent tools, which could do the work without human interference. At the same time he refers to the statues of Daedalus and the tripods of Hephaestus – mythological archetypes of the robots. Daedalus was regarded as the master capable to breathe life into their works (statues), while the tripods of Hephaestus were to be small tables, performing the role of mechanical servants. Descriptions of these machines are similar to the modern constructions, but there is no evidence of their existence.

The one of more famous and well described ancient automaton, which should be mentioned here, is the mechanism of automatic opening and closing the door of the temple (Fig. 1), whose constructor was Heron. Another work of Heron was a vending machine located at the entrance to the temple, selling water after inserting a coin [2, 3].

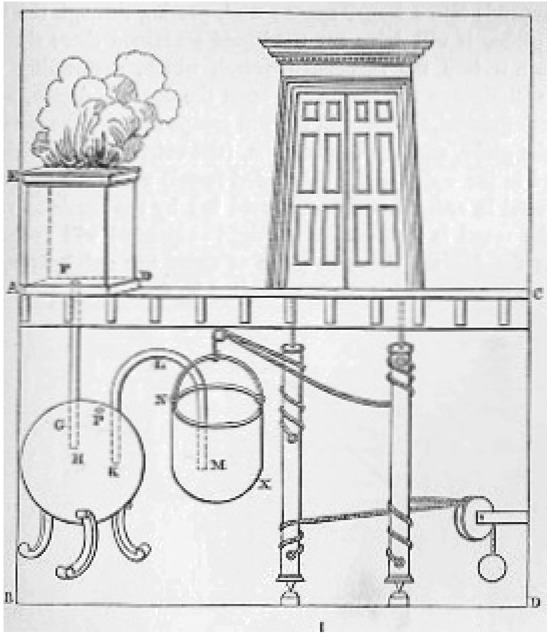


Fig. 1. The Heron's automatic temple door [2]

Most machines that were invented in ancient times had no possibility to change the program, since it was determined by the construction of a particular structure. On the other hand there was no need to change the program, because at that time automatons played the informational role (often being coupled with clock, like the giants on the top of the tower in Piazza San Marco or the mechanical cock atop the cathedral in Strasbourg), or the entertainment one. The exception was the mechanical orchestra which was constructed around 1200 by Al-Jazari [4]. In this case it was possible to change the rhythm of the drums through the appropriate placement of the pegs on the driving wheel.

2.2. Robotics and the period of the Enlightenment

The significant progress in the field of robotics coincided with the Enlightenment era. It was connected with the fact, that around 1500 spring-driven clockwork was invented. Thus, the drives of the automatons have become smaller and more accurate. At the same time, the number of programmable machines has been increased. One of the first was "The Flute Player", which was developed and presented in 1738 by Jacques de Vaucanson. The humanoid machine had the possibility to play twelve different melodies on the flute. Further machines of his authorship were "The Tambourine Player" and the famous "Duck" [4, 5].

Another famous constructor of automatons was Pierre Jaquet-Droz. Among the other things he built three well-known machines: “The Writer”, “The Draughtsman” and “The Musician” (Fig. 2) [6, 7]. Jaquet-Droz automata have been very complicated machines. The most complex mechanism, "The Writer" has been built using about 6,000 parts – this was the source of its advanced capabilities. The other two machines ("The Player" and "The Draughtsman") had a less complicated structure (respectively 2500 and 2000 parts). The construction of "The Player" was based on the cylinder with pins (pegs), which was the carrier of the “program” used for playing the melody, while "The Draughtsman" has had four programs that have resulted in four different drawings. "The Writer" uses a code disk which allows the machine to write any text to the maximum length of forty characters. This property has distinguished the automaton from others, because it could be said, that it executed a program with variable parameters – the text to write – while the other allowed only switching between different programs.

Fig. 2. The three famous automata of Jaquet-Droz: “The Draughtsman” on the left side, “The Player” in centre and “The Writer” on the right side [Wikimedia CC license]

Another interesting invention, which has been made for textile industry, is punched cards. This idea is attributed to J. M. Jaquard, but the punched cards were first used around 1725 by Bouchon and Falcon as a more reliable medium than paper tape. In 1832 this idea was used by Korsakov to store, retrieve and search the data [8].

2.3. Robotics in 20th century

Tremendous progress in the field of robotics has been made along with the development of electricity. A new kind of energy gave new opportunities to build drives that keep small size, but allowed to exert high torques and forces. On the other hand, the electric current was a good carrier of information, and because of the application of first electronic components (electron tubes, relays), the processing of the information became easier. These events again awakened dreams of having a "mechanical servant", what could be seen in the literature written in this period. In the drama “R.U.R.”, written by Karel Čapek, the word “robot” appeared for the first time and over the years shaped the imagination of people, forming associations with the humanoid machine. The leader in the manufacturing of complex robot in that time was the Westinghouse Electric Company [4, 9], which built a series of three robots: “Mr. Televox”, “Willie Vocalite” and “Electro”. These machines, were intended to act as “electronic servants”. The first of its, built in 1927, “Mr. Televox” basically was a remote control device, which – for better effect – has been placed in the humanoid-like case. The control process has been done by using the pitch pipe, which generated a certain pitch of sound. The transmission of the sound has been realised using standard telephone line. The automaton responses have taken a form of chirps or buzz – later the 78 RPM record has been used. The “Willie Vocalite” in comparison with “Mr. Televox” has gained mobility: the machine has the possibility to stand up, sitting, moving his arms, shoot the gun and smoking a cigarette. The robot could also “identify” the “taste” (it has been equipped with the acidity sensor) and has been able to speak (using the 78 RPM record). His successor – “Electro” – was built in 1937 and presented at the New York World's Fair in 1939 (Fig. 3). The machine also appeared in several films. “Electro” has inherited all the features of its predecessors, was able to walk and to distinguish between green and red light, using optical sensors. The specific feature of this robot, which distinguished it from the others, was the possibility to control them by using voice commands. The control system did not recognize the individual words spoken by the operator, but only reacted to the particular pattern of the operator’s voice. However, it was a successful attempt to “program” a robot using natural language, using the voice to describe actions which should be done.

The evolution of robotics after the World War II was heavily influenced by Asimov's prose. His vision inspired George Devol to develop a prototype industrial robot in 1954. The first machines have taken a job at the General Motors factory in the sixties of the last century. The development of electronics, particularly the use of transistors and integrated circuits allowed the rapid development of computers. Ongoing research are directed at increasing the efficiency and functionality of robots, thus the lasting for a long time artefacts of “iron man” had been abandoned. The style of programming the robots also has been changed, using a computer or dedicated terminals, which are then converted to Teach Pendants.

The off-line programming does not solve all the problems associated with programming of robot manipulator’s motion. Some parts of the program must be written into memory using on-line method, and this in turn involves the exclusion of the robot from the production process. Current studies are being focused on the possibility to minimize the on-line programming time, using advanced method like force control or voice control. These methods have a lot of in common with the visions of Aristotle or Asimov, which concern the communicating with the machine in a concise manner, using natural language. This requires endow the robot with systems which allow not only the perception of the environment, but also reasoning. The high level of abstraction of the human’s order should result in an automatic choice of the proper commands and – after putting them in the correct order, which allows to achieve the aim of the task – the complete program should be generated.

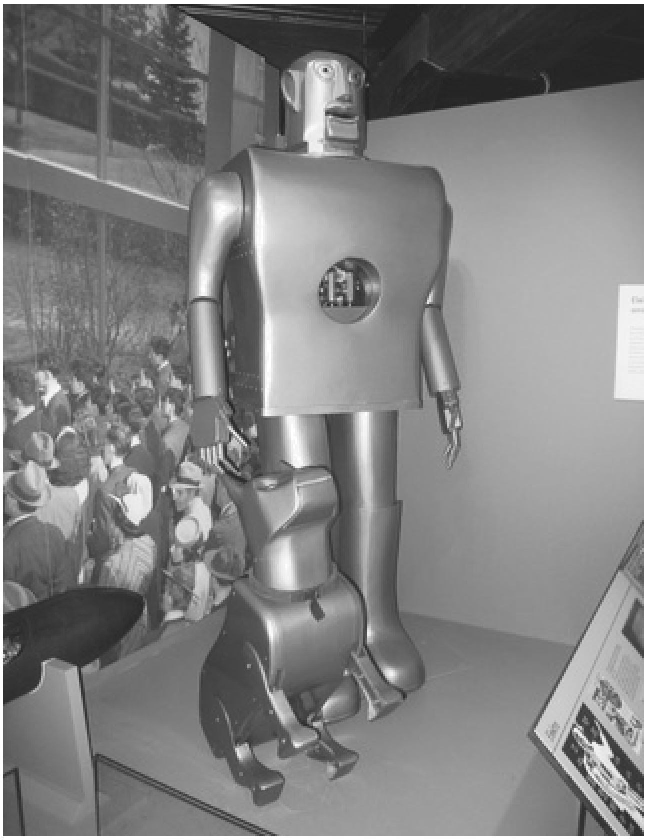


Fig. 3. “Elektro The Moto-man” and “Sparko The Moto-dog” robots in the Senator John Heinz History Center [Wikimedia CC linence]

2.4. Programming by demonstration

During doing some on-line programming tasks, from the practical point of view it is enough to automatically generate the “skeleton” of the robot’s program – the operator should only complete them entering all of the needed parameters. Because the modern control systems of the robots are primarily focused on the speed and precision and are not intended to perform complex computational or the reasoning programs, therefore external computers are used for doing these tasks. In such configuration, the computer processes operator’s commands given in some form of natural expression (voice commands, gestures etc.) and in that manner extends the capabilities of the robot’s controller.

Programming by demonstration [10] in its simplest form is done by force control of the robot. The other way is to use special control arm, which is similar to the other ones used in telecheric robotics. The third method is to use the “phantom” robot (which could be a real or scaled down model of the manipulator equipped with all of the needed sensors). The programming is done in such a way, that the operator moves the links of the kinematic chain of the “phantom” robot while the real robot repeats each move. More complex way involves the use of virtual reality or processing of the operator's hand movements on the appropriate trajectory. Any of the mentioned methods could be complemented with the voice control interface.

In the case of using the programming by demonstration method, there is exact reproduction of the operator’s intentions with modest degree of utilization of the artificial intelligence techniques.

3. Intentional programming of the robot (programming by the expression of the will)

The new way of on-line programming of the robot, which is proposed and generally described in this paper, is the method which could be called “programming by the expression of the will”. It combines the use of artificial intelligence, logic programming and programming by demonstration. It is very close method to the “intentional programming”, which is used in computer software creation, but in the robot’s world the “expression of the will” is more proper description. The main difference between presented method and the programming by demonstration is that the first one requires less involvement of the operator.

3.1. The problem

Let’s imagine the situation where a supervisor assigns the employee a job to do. Let it be, for example, unloading of boxes from the car and setting them in the right place at the warehouse. To clarify this task it is only needed to specify the object (what? – Boxes) and the path as the starting point and the end point (from ? to? – From the truck to the warehouse). Any other details, as the manner of unloading, planning of the path to avoid the obstacles etc. are specified by the employee. Similarly, when somebody leaves his car at the service, saying, "Please change the engine oil", the task is also completely described. The car service worker uses the information given by the customer (what should be done) in order to define all of the activities needed to complete the task. The employee can also have adequate instruction sets that contain a set of basic operations, performed during the realization of the task. Manuals, catalogues and the own knowledge of the worker are the kind of database, which is used in order to create the algorithm of realization of the task.

Having regard to the above description the following question could be asked: is it possible to program the robot in a similar way? This concept is close to the idea of the robot, which was popular at the beginning of the twentieth century: a servant, which is waiting for the next order. The contemporary state of the knowledge could allow at least the partial realization of this idea in relation to on-line programming of the industrial robots.

3.2. The implementation of the method

The key to implementation of the new method is the appropriate definition of the task. Unless the commands mentioned in the previous paragraph could be easily interpreted and it is not difficult to prepare a clear plan for their implementation, there are some commands that – being obvious for the human – are not clearly interpretable by a machine. For example, the machine will not be able to interpret the command "Clear object" without knowledge of the criteria of purity evaluation, method of cleaning and the type of movements it should perform. It is therefore necessary to use a knowledge database, but the time needed to work out the solution, which ensure the proper implementation of a command, may be longer than the direct programming by demonstration. Therefore it should be taken into consideration, that the use of logic programming and artificial intelligence method under certain conditions could be more time-consuming. On the other hand, the stationary industrial robots usually perform the transportation tasks, where the complexity of such a task is relatively small in comparison to the mobile or humanoid robots. The working environment of the industrial robot is rather static, so there is no need to cope with some unusual situations. Also the set of the tasks will be very limited. There is also exception from this rule – more complicated tasks realized by painting or welding robots.

The very useful way of task's description is the demonstration, but in the case of "programming by the expression of will" it is not done by the "show me how to do this / watch what I do" method. The new method uses the "show me the state before and after" approach. The input data have a form of the pictures set: the first set shows the "before" state, while the second one shows the "after" state. These pictures then should be analysed by the computer program and the "skeleton code" for the robot could be created, which in turn could be processed by the operator in order to write the complete program. The difference between "programming by demonstration" and "programming by the expression" of the will is shown in Figure 4. In the first case the operator should "show" the robot how to move the orange box from the "start" point to the "end" point. The path could be described by moving the object closed in the robot's gripper (the operator moves the whole kinematic chain of the robot's manipulator) or by moving the special marker that would be observed by the camera connected to the computer or the control unit. In the second case, the operator shows only the "start" and the "end" position and gives the command like for example: "move from start to end". The path is then planned by the computer according to possible limitations, like e.g. obstacles situated between given points. It could be derived using one of the many of existing method, e.g.:

- BUG method,
- Braitenberg algorithm,
- Visibility graph method,
- Potential fields method
- 2½D algorithm,
- Q-learning and Markov chains.

The methods of trajectory planning, which were listed above, will not be described there, because they are well depicted in the other sources [11-14].

3.3. The computer vision system

As it was mentioned earlier, one of the proposed ways of giving information about the task is to show the state of the workspace before and after the realization of the work. It could be realized using the set of three cameras (Fig. 5):

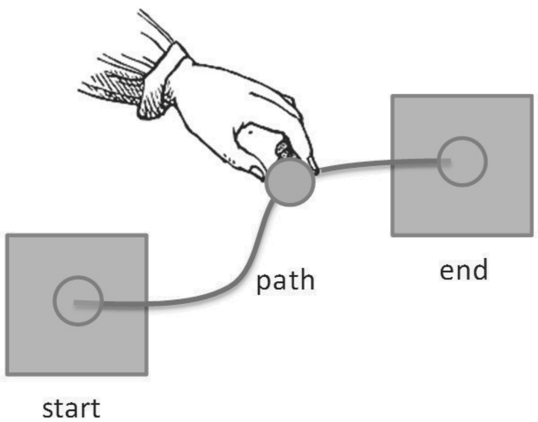
- The first above the workspace, which observes the X and Y coordinates,
- The second at the workspace border, which observes the X and Z coordinates,
- The third at the workspace border, which observes the Y and Z coordinates.

Of course two cameras will be enough in this case, but three cameras causes that every coordinate is measured twice. This could help to set the proper positions of the cameras and calibrate the measurement system.

The role of the cameras is to take the snapshot of the workspace – first when the operator has arranged the "before" state, and the second when the "after" state has been arranged. There is no need to use the cameras all the time, so in the case of a small robotic system, there could be only one camera, which will be mounted on the special stand.

The image recognition processor could be calibrated using the photo ruler (Fig. 6). This is very simple, but quite effective method, because it allows positioning the camera at any (rational) distance. In this manner the origin of the coordinate system could be marked.

a)



b)

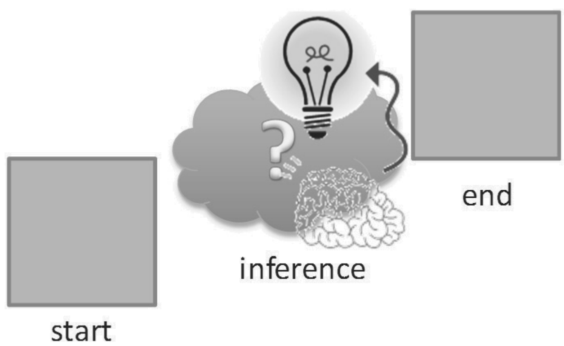


Fig. 4. The difference between programming by demonstration (a) and programming by the expression of the will (b)

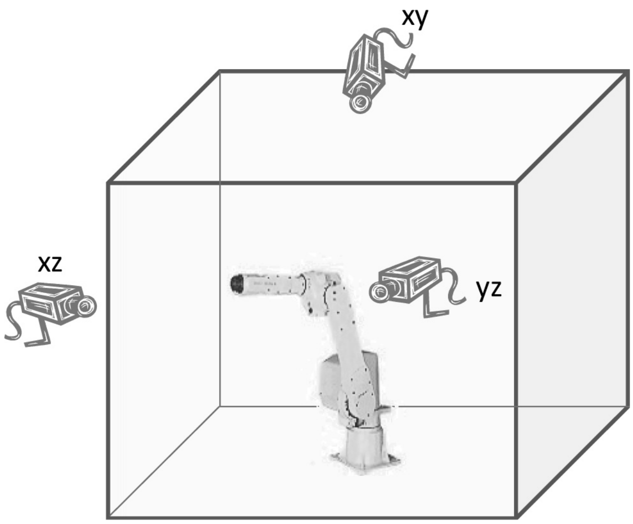


Fig. 5. Positions of the cameras for observing the robot's workspace

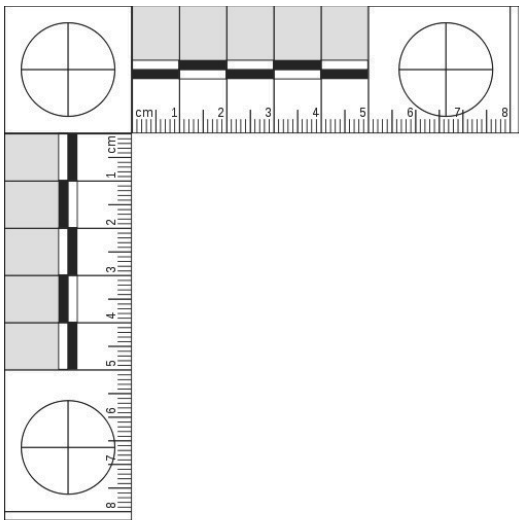


Fig. 6. The kind of photo ruler, which could be used as coordinate system origin marker

The photo ruler is very characteristic, and could be easily recognized during calibration.

3.4. Limitations of the proposed method and some technical aspects for future overworking

The most important, but also the most error-prone system is the image recognition processor. In the further development of the system it should be taken into consideration, that there is no very effective method for the separation of manipulation objects from the background. There could be many disturbing factors, for example:

- a sunbeam, which could disturb the photo exposition by giving the non-uniform, moving illumination,
- a person or the other robot working on the next stand, other moving objects (e.g. line or curtain on the wind),
- objects that have complicated shapes or are painted in complex patterns,
- places, which are not well illuminated,
- shining and reflective objects of manipulation.

In many cases it may help, when the system is "informed" about the static environment. Because the method does not involve the manipulator arm during presentation of the "before" and the "after" state, it is enough to take photo of the empty workspace, before the presentation. Also separation screens, painted in neutral colours, could help to "remove" the unwanted objects from the background.

There could be also the situation, where one of the workspace states could not be correctly taken by the camera. For example some of objects could be hidden behind the manipulator's body. To solve this problem, the operator could divide the whole workspace for some smaller parts, which could be treated separately. In this manner every sub-space could be photographed individually, and because its smaller dimensions, there is no need to install separate cameras for them. This method should also give better results during image processing, because of better illumination and higher visibility of details.

In the Figure 7 the bad quality image is shown. First of all the camera is taking photo from the angle, which give the incorrect information about the object shape (Fig. 8). The other problem is the glass table. Transparent and reflective surface gives the undesirable information noise:

- because of transparency, the image is overloaded with the unneeded information about objects situated under the table,
- because of the reflective surface there are artefacts of the objects, they are clearly visible in the Figure 8, under the silver cylinder and the red cylinder situated in the centre.

There are also visible lightbeams, which have negative influence on the exposition.

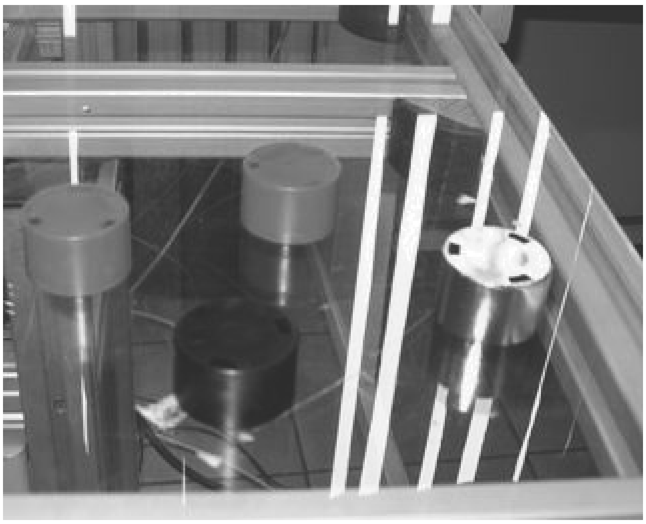


Fig. 7. The bad quality of the image

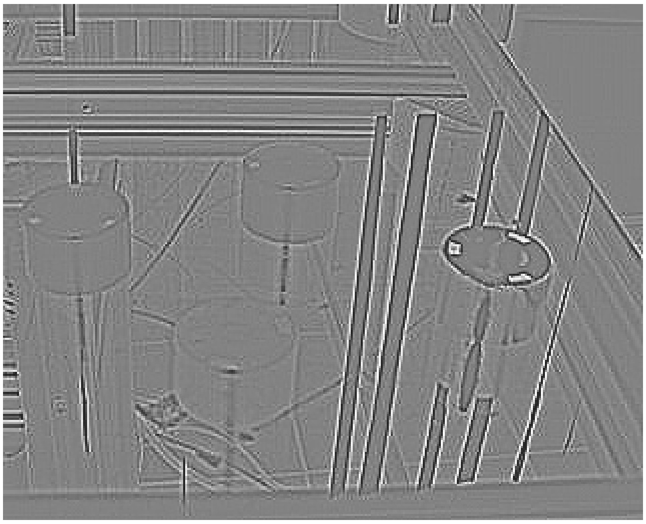


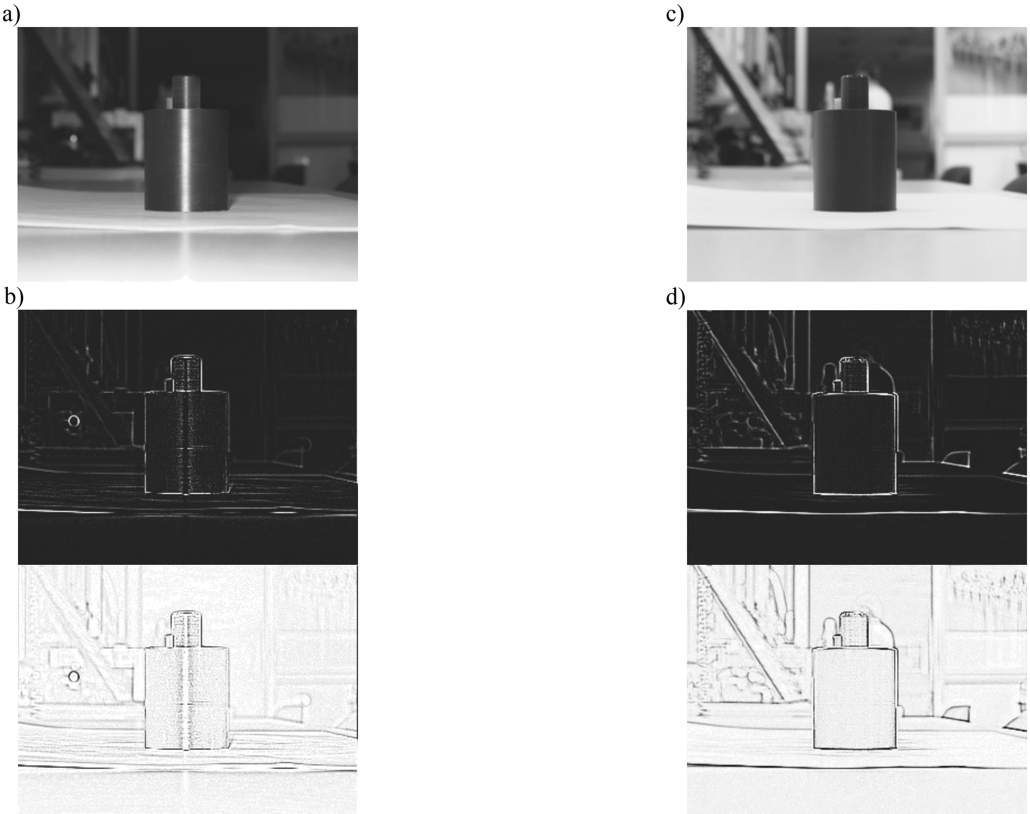
Fig. 8. Border detection has been done on image from Fig. 7. The shape of the objects cannot be properly classified

The other problem of the vision system is to properly recognize the objects, which colours are very close to the background. The situation is the same, when there is no good illumination or objects are backlighted. So, it is very important to properly adjust the light source. In the Figure 9 an example of two different light conditions is shown. Using the bright and sharp source of the light (Fig. 9a) gives the better detection of the objects on the first plane, while the background is dark (Fig. 9b). There is a problem with reflections in the object surface, which should be properly filtered. On the other hand, the use of the smooth light (Fig. 9c) could result in better object detection, but more unneeded background object are also detected, which should be removed from the image (Fig. 9d).

The other issue to discuss is to cope with the variety of shapes of the objects of manipulation (Fig. 10). The question is where and how to grasp the object in order to realize the manipulation task properly? First of all, the solution depends on the specific task of manipulation. Having the final structure, as shown in Figure 11, there is no possibility to grasp the black cylinder from the side – it must be done from the top. The other cylinders could be taken from the top or aside, but in this particular cause it could be unified, so every object would be taken from the top. In addition, every object could be inscribed into a cuboid, where the height, width and length are determined on the basis of maximum outer dimensions of the object. The manipulation is done in such a way that the geometric centre of the cuboid coincides with the characteristic point of the manipulator. Unfortunately, this assumption does not exclude cases where the grasp would be uncertain (Fig. 12). In such cases, the operator intervention is required.

a)

c)



b)

d)

Fig. 9. The influence of illumination and the light source on the effectiveness of object detection algorithm (the Laplace edge detection algorithm has been used)

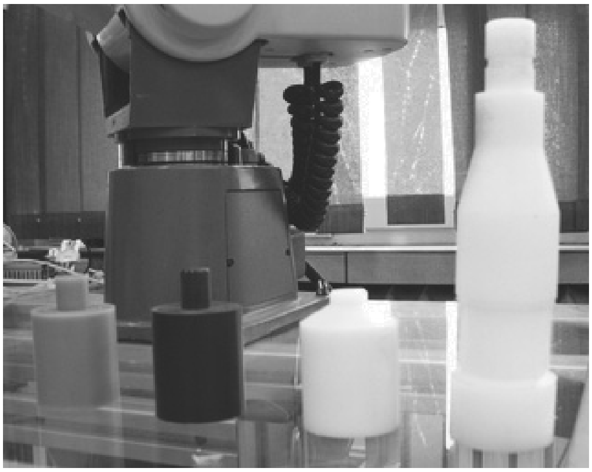


Fig. 10. The example of variety of colours and shapes of the objects of manipulation

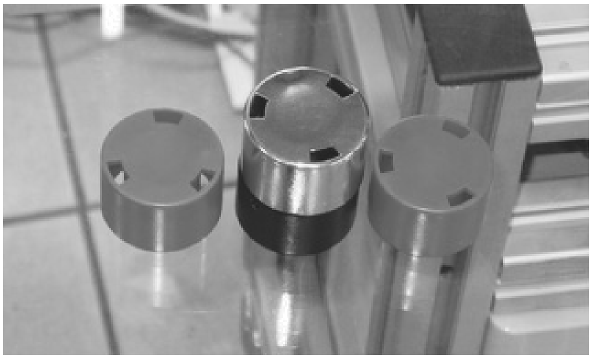


Fig. 11. The example of final effect of the manipulation task

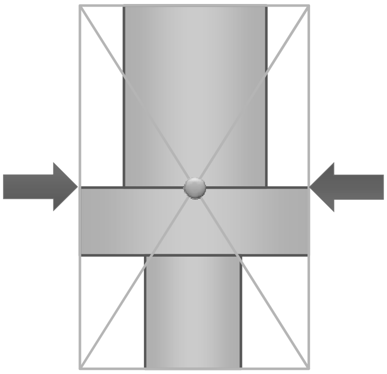


Fig. 12. The case of the uncertain grasp

4. Conclusions

The presented concept of the method of robots programming by the expression of the will (or the intentional programming) is in early phase of development. There are plenty of unsolved problems, which could affect the overall performance and the efficiency of this method. The problem with the highest priority is to create the error-resistant image recognition system, which could work with popular types of cameras. This application should run efficiently on most modern computer systems, including portable computers.

The implementation of the system will require the use of the logic programming language. There already are some implementations (e.g. Prolog), which may be used directly, but some portions of the system will be developed from scratch.

For now, there is no plan to include Virtual Reality or voice control of the robot, but the system will be designed for further equipment with new interfaces.

References

[1] Aristotle, Politics, Harvard University Press, 1977. Acquired from the Google Books, http://books.google.pl/.

[2] B. Woodcroft, The pneumatic of Hero of Alexandria, Taylor Walton and Maberlye, London, 1851. Acquired from http://ww.history.rochester.edu/steam/hero/

[3] M. Lahanas, Heron of Alexandria, Inventions, Biography, Science (1/2). http://www.mlahanas.de/Greeks/ Heron Alexandria.htm, accessed 10.11.2011.

[4] R. Malone, The Robot Book, Push Pin Press, 1978.

[5] W. Clark, S. Schaffer, The sciences in enlightened Europe, University of Chicago Press, 1999. Acquired from the Google Books, http://books.google.pl/.

[6] M.E. Rosheim, Robot evolution: the development of anthrobotics, Wiley-IEEE, 1994. Acquired from the Google Books, http://books.google.pl/.

[7] N.B. Abbas, Thinking machines: discourses of artificial intelligence, LIT Verlag, Münster, 2006. Acquired from the Google Books, http://books.google.pl/.

[8] Korsakov's "Intellectual Machines", http://homeoscope.ru/ eng/index.html. Accessed 10.11.2011.

[9] Electro, http://davidszondy.com/future/robot/elektro1.htm. Accessed 10.11.2011.

[10] S. Calinon, Robot programming by demonstration: a probabilistic approach, EPFL Press, 2009. Acquired from the Google Books, http://books.google.pl/.

[11] G.G. Kost, The use of the Q-learning algorithm for programming the industrial robots, Proceedings of the 12th International Scientific Conference "Achievements in Mechanical and Materials Engineering" AMME'2003, Gliwice – Zakopane, 2003, 505-508 (in Polish).

[12] D. Reclik, G. Kost, The comparison of elastic band and B-Spline polynomials methods in smoothing process of collision-less robot trajectory, Journal of Achievements in Materials and Manufacturing Engineering 29/2 (2008) 187-190.

[13] G.G. Kost, R. Zdanowicz, Modeling of manufacturing systems and robot motions, Journal of Materials Processing Technology 164/165 (2005) 1369-1378.

[14] K. Foit, G.G. Kost, D. Reclik, Automatic programming and generation of collision-free paths for the Mitsubishi Movemaster RV-M1 robot, Journal of Achievements in Materials and Manufacturing Engineering 47/1 (2011) 57-65.