VISION RECOGNITION AND APPLICATION TO MPS 500 SYSTEM

РАЗЛИЧЕНИЕ КАРТИНЫ И ЕЁ АППЛИКАЦИЯ В СИСТЕМ MPS 500

Ing. Kamil Židek, PhD., Ing. Vladislav Maxim, PhD., prof. Jozef Kováč, PhD
Technical University – Košice, Slovak Republic,
E-mail: kamil.zidek@tuke.sk, vladislav.maxim@tuke.sk,

Abstract: *This article deals with replace sorting module system of MPS 500 based on sensors system by camera system. The project discusses about development system and programming language, in which is software for separating development. Compendium and analyze of components in which is camera system based on (camera, computer – control software, parallel port, electrical circuit, pneumatic parts). Description of block diagrams that describe point of work main program and algorithm of picture recognize. Last part describes main formulation of software's source code created by programming language C# in development system MS Visual Studio.*

KEY WORDS: *VISION RECOGNITION, VIDEO CAMERA SYSTEMS, MPS500.*

1. Introduction

This article deals with replacing of standard sorting system to System witch is based on Video Camera recognition for modular system MPS 500. Present solution of sorting module is based on sensors. Alternative solution is created on Video Camera system. The Task was created system from cheapest components, for example by quality web cam, where the price is about 20 times cheaper as industrial video-cam. These industrial components offers frame frequencies about 60 or more per second. For ours task is not a very important high frames fervencies per second neither high resolution, because parts on belt don't have so much speed. There is enough 11 to 25 frames, that it is possible to use cheap alternative as webcam. In the article are described components of video camera system, main features of software with detailed description of algorithms and possibilities of optimalization.

2. Principe of Vision Recognition.

All objects (computer vision too) are perceptible thanks to light, which they produce or which is reflected from its. Light is characterized by several attributes:

- Colour hangs on electromagnetic waves and contains:
  - o Brightness – representing intensity of light,
  - o Saturation – color representing cleanness of color.

From physical aspect is light electromagnetic oscillation in field from 380 THz to 770 THz.. Every frequencies belong some color. In visible spectrum we can differ more than 400 000 color tone. Reception of color tone is almost subjective. For creating color picture we need set of tones. Method of obtaining these tones from class of basic color declares models of color creating.

For work with color is necessary to obtain:

- Basic set of colors, with which we will be work,
- Method of combining basic color.

There are two basic way of combination color:

- Additive combining: addition of some tone arise lighter color, combination of all basic colors arise white color.
- Subtractive combining: addition of some tone arise darker color. Combination of all basic color arise black color. Typical model is CMY.

For working with tone was developed several systems, which differ in set of basic colors a rules of its combination. Common for all models is effort approach to human reception. By work with color on Computer it is necessary to expressly digitalization all attributes of colors. [1]

Basic models of color combination:

- model RGB,
- model CMY(K),
- model HSB, HLS, UWB

Subtractive model CMYK, is suitable for color coding for printers, but for usability of picture analysis isn't suitable. Model HSB are used when RGB is not able describe enough color with is coding. RGB model is basic model, which is in common used and supported in developing environment for software. When we will want use by analyze, other model as RGB, we must retransform it, what does it take time during analyze and slow down whole process. Further now use of others models as RGB for analyses is unpractical and don't offer better coding of colors. I choose RGB model as the best for speed color recognition in my software.

2.1 Model RGB.

Color model RGB "fig.1" is additive model, where red, green and blue color are combined in differ of quantities for display of other colors. Name of this model "RGB" is created from 3 basic color of red, green and blue. Colors in RGB model can by represented by number values in interval <0,1> or percent form for every basic color. Intensity (Brightness) of basic colors is interval <0,1> too, whereby 0 – is a minimum and 1 is maximum of color intensity. In regard to sensitivity of human eye is enough divide this interval to 100 parts. In computer graphics it is preferable divide it from 0 to 256. We can code intensity by 8 bits. Every color tone in this case can be represented by 3*8 = 24 bits, then we can create 16 777 216 differ colors. According this coding is every color coded by 24 bits, what are 3 memory places in computer. For example 255 0 0 is saturated red, 0 255 0 is saturated green color, 0 0 255 is saturated blue color, 0 0 0 is black color and 255 255 255 is white color.

24-bits picture with measure 640 x 480 pixels:
24 x 640 x 480 = 7,372,800 bits
7,372,800 / 8 = 921,600 bytes [1]

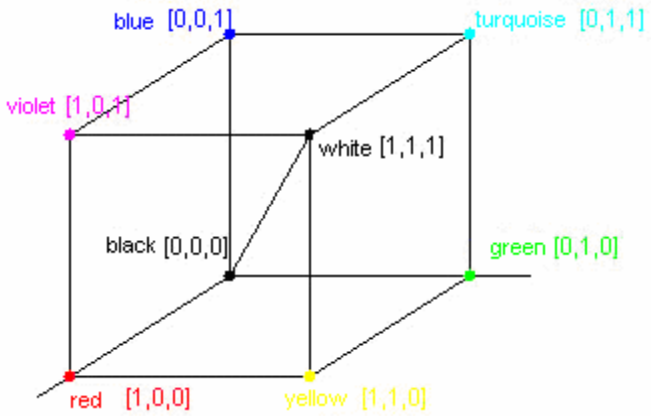


Fig. 1 RBG model

3. Preprogrammed libraries for picture processing.

Developing environment as Microsoft Visual Studio or Borland Builder don't offer any included components of libraries for picture color recognition, therefore it is necessary create your own or use some which are in disposition. We can found two types of libraries or components:

- for recognition shapes and colors.
- special libraries of motion recognition.

One of components from first group is Pegasus CapturePRO, offer methods for video processing and audio too. Pegasus Capture PRO offers possibility works with wide amount of video equipments which support USB, FireWire, WDM or DirectShow. CapturePRO contains .NET, COM and VCL components (CapturePRO a TwainPRO), which offers tools for picture and video processing from video device. Primarily is suitable for creating software for color and shape recognition. For free is available as trial version and limited time is full functional. [9]

Source code for motion recognition we can download there [10]. These source codes are open source and we can use it unlimited times with any restriction. Recognized are movable objects with differs methods, and there are implemented many ways of algorithm optimalization. Furthermore there is special algorithm for recognition more object separately on the picture in real time.

4. Design of recognition module.

System we can divide to two groups: recognition and action part.

Description of recognition chain: Recognition parts construct of Video Camera connected to computer thru USB port, which communicated with created software. This software recognizes color of the parts.

Description of action chain: In base of recognized color of parts then program send signals thru parallel port to amplifier, which is an able control electropneumatics block. Electro pneumatics blocks then control movable piston in sorting module and control of parts flow to 3 magazines.

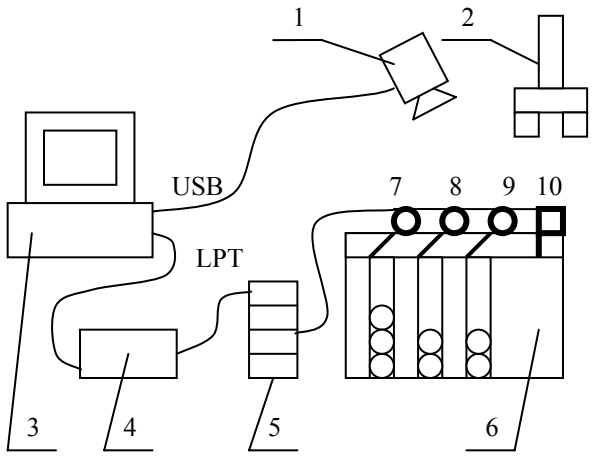


Fig. 2 Scheme of Recognition System

Description of scheme:

1. Usb Web Camera
2. Pneumatics gripper
3. PC with program for picture recognition
4. Amplifier of signals from LPT (based on optoparts)
5. Block of electro-pneumatics distributor
6. FMS 500 – Sorting modul
7., 8., 9. Pneumatics rotable piston
10. Pneumatics piston

Next parts is describing in detail all parts both created chains.

4.1. Usb web camera

By choosing of web camera it was necessary make provision for these parameters:

- Its shape (for easy fit),
- Quality and max number of frames per second (Video resolution, fps)
- Picture resolution
- Price

I choose Microsoft Lifecam NX-6000 “fig.3”. It offers small measures and very quality picture in high resolution.

Fig. 3 Usb Web Camera Nx-6000

Basic parameters:

- High resolution of static photograph - 7,6Mpx (3200x2380px)
- Video resolution: 2Mpx
- Usb 2.0
- 3x digital zoom
- Wide-angle objective - 71 grade
- Price with tax: 2 500 Sk

4.2. Communication with parallel port.

Parallel port is well known as LPT (from eng. Line Printer Terminal) is specific with 25 pin connector. LPT port is a standard part of PC for parallel communication (parallel transform of bit signals) offer 17 digital links, which we can divide to eight pins (2-9) for direct write, four for reading (10-13 and 15), four for both operation (1,14,16 and 17) and last eight pins (18-25) are ground fig.4.

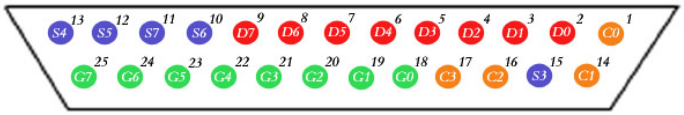


Fig. 4 Usb Web Camera Nx-6000

This interface offer 8 data lines for electro-pneumatics blocks. In our solution we used only first four data lines from D0 to D3, last four can be used later for next extension.

Output signals are defined by classical logic TTL, does it means. log. 1. is +3.5V to +5V and log. 0 is 0V to +0.4V.

4.3. Signal amplifier.

For connection of computer thru parallel port and electro pneumatics distributor was necessary to create auxiliary circuit. Voltage, which can parallel port offer on output is 3,4 V. This Voltage is too small for control of electro pneumatics distributor, which then can on or off pneumatics piston on sorting system in system Festo MPS 500. For control electro pneumatics distributor it is necessary voltage 24 V. Design of circuit “fig. 5.”

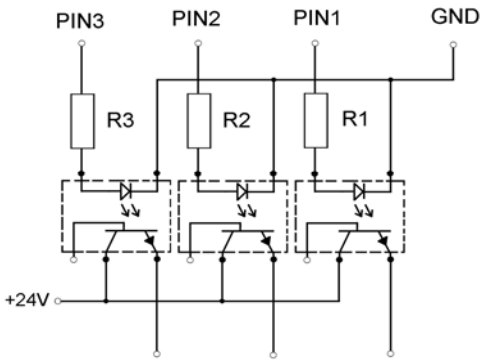


Fig. 5 Signal amplifier

Pins PIN1, PIN2, PIN3 are output from parallel port of Computer connector. Three wires on other side are connected to electro pneumatics distributor.

4.4 Program for sorting of objects in system FESTO

Whole program contain these two basic parts:

- Control part of program(main program)
- Algorithm for optical recognition

4.4.1 Control part of program

This part of program is superior to algorithm for vision recognition. When program is launched all value are set to default, then run connection with camera. This part of program offer user friendly interface and setup of basic value for example sensitivity of color. Main part of program launch timer, which run commands for vision recognition every 100ms (this value we can change according of request how many pictures we need to analyze in one second).

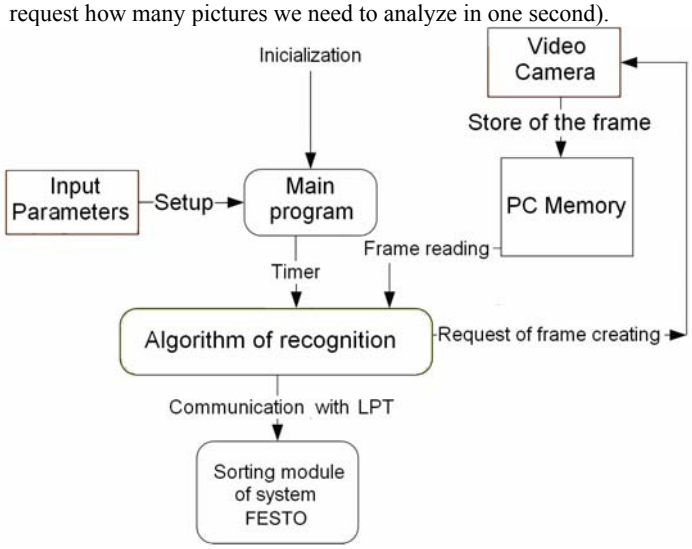


Fig. 6 Algorithm of main program.

Fig.7 shows graphical interface of program, it is possible setup parameters during recognition. Program shows following panels, which are divided according function and possibilities of setup:

- Scanning density for vertical and horizontal rastering.
- Parts counter.
- Color palette of sensitivity.
- Camera window with graphical location of the part.

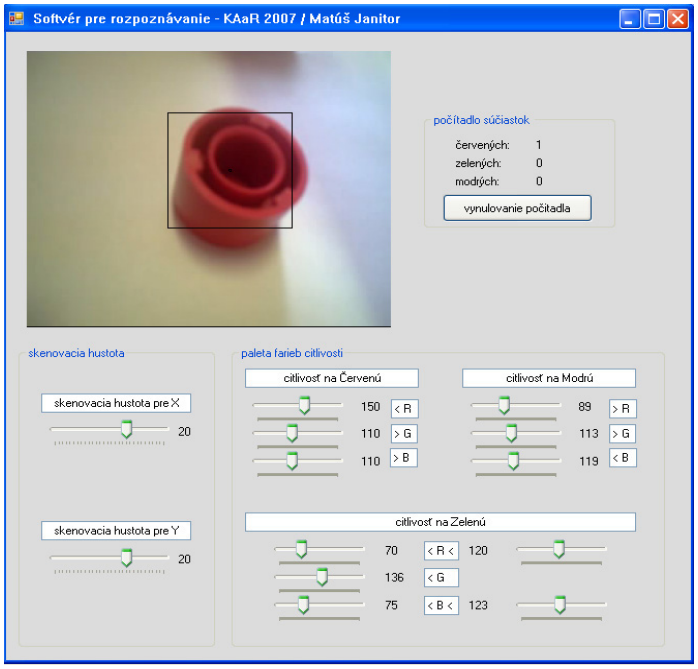


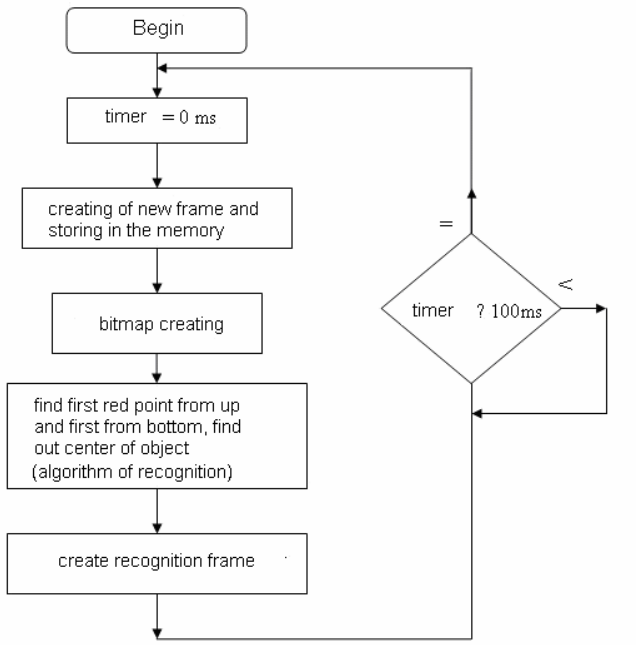
Fig. 7 Algorithm of main program.

4.4.2 Algorithm for vision recognition.

Program algorithm creates main part of optical recognition on base of color recognized object. It is mandatory design this algorithm with high reliability and mainly high speed, because this algorithm will be run in loop according to request of number of frames per second.

If frame has 300 pixels on width and 300 pixels on high, then by analyze must algorithm run for every of pixel from 90 000 pixels which the picture contain. By period of 10 analyzed frames for one second it is about million of analyzed pixels. For recognition of color it is enough, if algorithm run for every third pixel (this value is not fixed, we can setup thru value of „scanning density X or Y“) in column and in row of frame. This optimalization offers high speed to algorithm it run only 100 000 times for second, what introduce cut-down run of algorithm about 90 %. Next method for speedup of algorithm is lowering of resolution created by video camera system.

Fig. 8 Algorithm of parts color recognition



4.4.3 Principe of recognition.

Algorithm analyses frames from video camera in bitmap format, scanning color of every pixel, begin with pixel position data: x = 0 , y = 0 (left up corner of frame) continue with x pixel position incremented value 1 (or more which hang on scanning density setting) to the end of width of picture. If maximum x position in the cycle is reached, value of x location is setup to 0 value and y location is increased to 1(or more). This cycle repeating to first pixel which satisfied condition about part color for example red color: R > 210, G < 40, B < 40, these condition introduces sensitivity to deviation for red color of pixel. Fig.9, pixel with location X1, Y1 fulfills condition for red point. In this point the scanning end and to the memory are stored values X1, Y1 and color which algorithm found as first in the frame. Scanning continuing with same principle as from up but now in direction from down to up(right down corner of frame). If scanning of pixel reach the pixel which met necessary condition for color which are found in first scan, then are stored to memory location of pixel X2, Y2. There scanning end and next are counted Xt=X1+(X2-X1)/2, Yt=Y1+(Y2-Y1)/2, it is a center of density of object between pixel 1 (X1,Y1) and pixel 2 (X2, Y2). Whole algorithm is repeated according of requirement of analyzed frames per second.

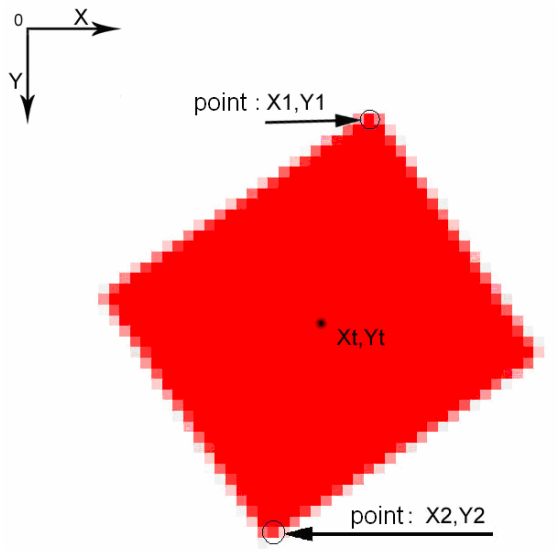


Fig. 9 System of detection part in picture

4.4.4 Palette of color sensitivity.

Palette of color on which will be program react by analyze can be very big, but disadvantage will be longer source code, there will be more commands and program run much more slowly. This program is especially written for system FESTO, where parts have only basic color, I choose that program will be react only to three basic color of RGB model: red, green, blue. Usability of this model is easier for works with colors by programming, which are basic color of RGB model. Any parts of color pixel will be not contain value of RGB ideally, because was necessary established condition for every part of RGB, which are the color in required interval.

If pixel is red, must its part met these condition:
pixel-R >190, pixel-G <110, pixel-B <110
If pixel is green, must its part met these condition:
pixel-R <110, pixel-G >190, pixel-B <110
If pixel is blue, must its part met these condition:
pixel-R <110, pixel-G <110, pixel-B >190

When we assign easy condition, for example red color on frame: R=255, G=0, B=0, then algorithm during analyzing don't found any pixel in this ideal condition, because there are some distriburbance. We need still use interval for every recognized color.

These condition bounds „from, to'' is problematic to obtain by computing because light condition change in every room during whole day. These bound can be easier find out experimental.

5. Realization of vision module.

Fig.9 shows module of system MPS 500 before of implementation of video camera recognition system.

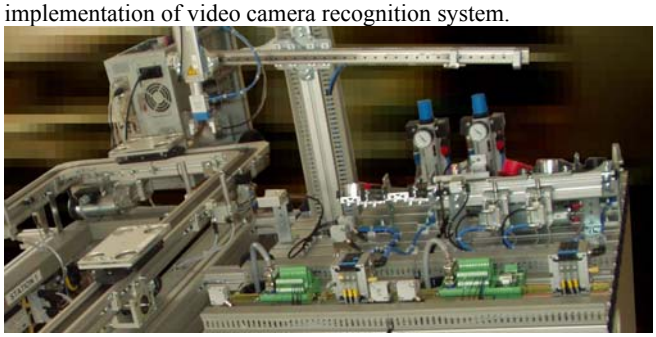


Fig. 10 Sorting station MPS500

Fig.10 shows module with implemented extension. It is possible use new solution of recognition and sorting of parts with video camera system or it is possible switch off this extension with help of switch on amplifier block change to basic system of sorting based on sensors. Then will be activated based control with PLC system.

Demonstration video you can be download on link [11].

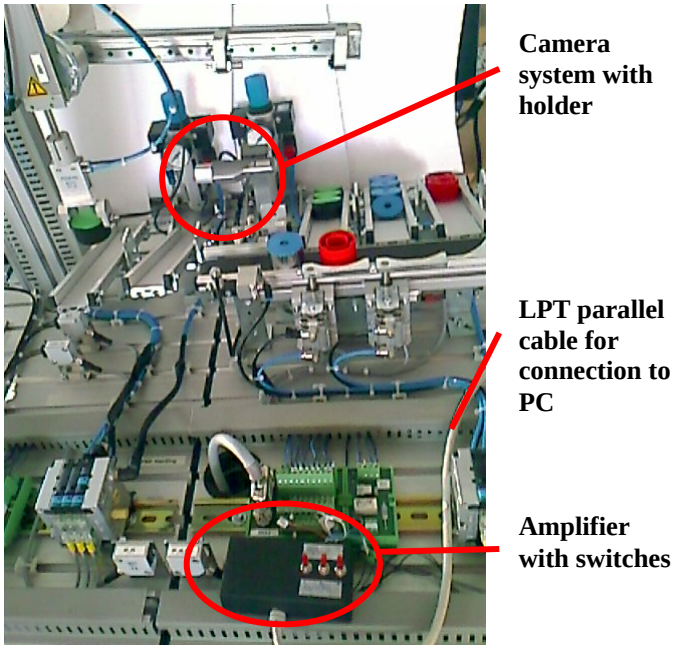


Fig. 12 Sorting station MPS500 with Picture recognition.

5. Conclusion

This article introduces of solution for part differs based on color recognition for system Festo MPS 500 and is integrated to sorting module. Solution include choose of suitable video camera, design of electronic circuit for amplifier component, developing software for recognition of color with parallel communication interface and optimalization of source code.

Result is connected and working video camera system controlled by developed software. This system will be used for teaching of student and will be based for creating other similar systems.

Next extension of software can be recognition of parts shape, or inspection works which compare produced part with comparing etalon and find out distortions. If we setup enough resolution we can very precisely measure dimensions of parts, as alternative to standard methods.

This article was written with support of solving granted project VEGA No.1/2178/05 and KEGA č. 3/120103 Virtual programming modules systems with artificial intelligence.

6. Literature

[1] Pgrafika, Počítačová grafika [online]. Aktualizované 2003 [cit.2007-04-30], http://www.pgrafika.host.sk/index1.html
[2] Wikipédia, Kódovanie farieb[online]. Aktualizované 2007 [cit.2007-04-30], http://sk.wikipedia.org/wiki/CMYK
[3] Sharp J.: Microsoft Visual C# 2005 - krok za krokem, Brno, Computer Press, 2005.
[4] Virius M.:C# - Hotové řesení, Brno, Computer Press, 2006, ISBN: 80-251-1084-2
[5] HW Server s.r.o., Dokumentace-paralelny port, http://hw.cz/Teorie-a-praxe/Dokumentace/ART819-Paralelni-port---prenosovy-software.html
[6] Kučera M., Rosulek M.: Web kamery pro E228 [online], Aktualizované 2006 [cit.2007-04-20], www.pcelf.cz/studium/vsb/arl/sempra/index.htm
[7] Berthold K. Horn: Robot Vision (MIT Electrical Engineering and Computer Science), MIT Electrical Engineering, 2003.
[8] Nagel,Ch.: C# 2005 – Programujeme profesionálně, Brno, Computer Press, 2005.
[9] Pegasus CapturePRO. http://www.pegasusimaging.com/capprodownload.htm
[10] Motion detecition. http://www.codeproject.com/cs/media/Motion_Detection.asp
[11] www.kaar.sebsoft.com/jomla/Color_recognition.mp4