

OBJECT RECOGNITION METHODS IN COMPUTER VISION USING IMAGE PROCESSING LIBRARY EMGU CV *OBJEKTU ATPAZĪŠANAS METODES TEHNISKAJĀ REDZĒ IZMANTOJOT ATTĒLU APSTRĀDES BIBLIOTĒKU EMGU CV*

Autors: **Andris Balodis**, e-pasts: shalunn@inbox.lv

Zinātniskā darba vadītājs: **Pēteris Grabusts, Dr.sc.ing.,prof.**, e-pasts: peteris.grabusts@rta.lv
Rēzeknes Tehnoloģiju akadēmija, Atbrīvošanas aleja 115, Rēzekne

Abstract. *This paper describes how object recognition is being done in delta robot station using computer vision which functionality is based on image processing library Emgu CV. In this paper there are developed some algorithms that determine rectangle form object position, color, angle of rotation and area. All these parameters allow to position and sort objects that move on conveyor belt. Each algorithm testing is being done on delta robot station that already contains video camera and artificial light source. The results show that algorithms do all necessary parameters determination and in addition these algorithms allow determining parameters for square and round objects too. Developed algorithms can be used in several disciplines, for example, in food and stationery products sorting. Application of computer vision simplifies and makes more efficient every monotone process especially when this process is being done by robots which can increase process speed and precision multiple times.*

Keywords: *computer vision, delta robot station, Emgu CV, image processing, object recognition.*

Ievads

Mūsdienās ikvienu ražošanas procesu mēģina automatizēt ar mērķi paaugstināt darba ražīgumu un izvairīties no cilvēkresursu kļūdām.[1] Preču salikšanas procesa automatizācijai bieži pielieto vairākas tehnoloģijas, kuras aizvieto cilvēkresursus. Viens no automatizācijas variantiem ir izmantot preču sortēšanas staciju, kas sastāv no delta robota, tehniskās redzes un konveijera.[2] Priekšmeti, kas nāk pa konveijera līniju un tiek tehniskās redzes darba zonā, tiek atpazīti un datorvadības sistēma, vadot delta robotu, nodrošina priekšmeta satveršanu un pārvietošanu uz noteiktu pozīciju. Šajā darbā tiek veikta vairāku tehniskās redzes algoritmu izstrāde.[3]

Darba mērķis ir izstrādāt algoritmus taisnstūrveida objektu pozīcijas, krāsas, pagrieziena leņķa un platības noteikšanai ar tehniskās redzes palīdzību.

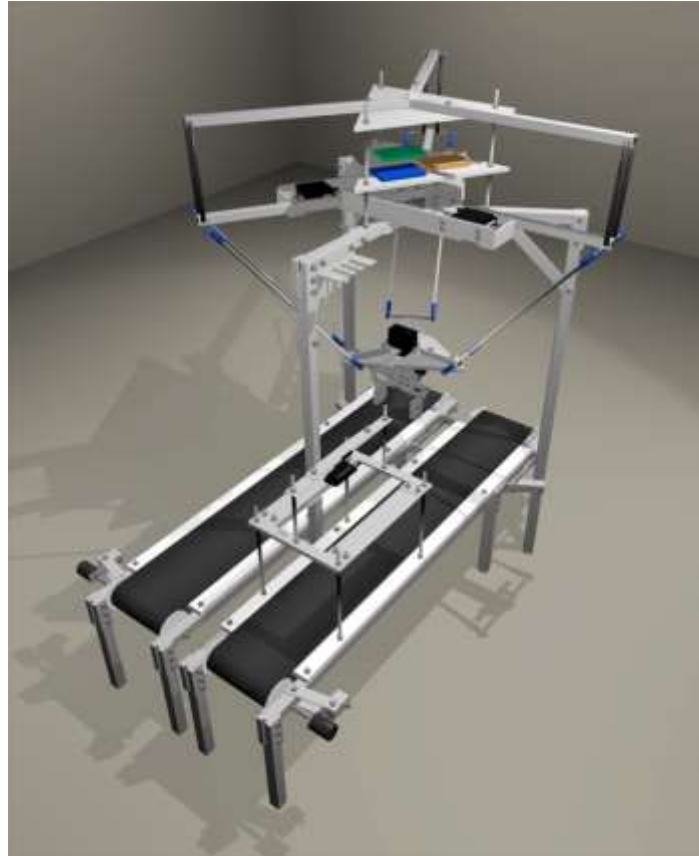
Darbā tiek izpildīti sekojoši uzdevumi:

1. Tiek nodrošināta video kameras attēlu iegūšana datorprogrammā.
2. Tiek nodrošināta attēlu apstrāde objektu pozīcijas, krāsas, pagrieziena leņķa un platības noteikšanai.
3. Tiek piedāvāta metode, kā uzlabot attēlu apstrādes sistēmas darbību.

Objekti un metodes

Lai atvieglotu šo algoritmu izstrādi, ir pieejama delta robota stacija, kura satur video kameru un mākslīgo gaismas avotu. Datorprogrammas izstrāde tiek veikta izstrādes vidē Visual Studio 2012, izmantojot bibliotēku Emgu CV.

Emgu CV ir brīvi pieejama attēlu apstrādes bibliotēka, kuru izstrādāja ar mērķi dot ikvienam cilvēkam iespēju veidot savas attēlu apstrādes sistēmas.[4] Bibliotēkā ir integrētas vairākas gatavās metodes, kas dod iespēju ātrāk un efektīvāk veikt attēlu apstrādi, netērējot laiku bāzes funkcionalitātes izstrādāšanai.[5] Šajā darbā tiek aprakstīts tas, kādā veidā tiek īstenota objektu atpazīšana delta robota stacijā, izmantojot Emgu CV bibliotēku. Izmantojamā delta robota stacija tiek parādīta 1. att.



1. att. Delta robota stacija ar tehnisko redzi

Datorprogrammā attēlu iegūšanai no video kameras, ir nepieciešams izveidot atsevišķu datorprogrammas plūsmu (*Thread*), kas paralēlā režīmā veic datu nolasīšanu no video kameras. Šī datu nolasīšana tiek veikta ar metodi, kura tiek izsaukta 30 reizes sekundē, nodrošinot šādā veidā nepārtrauktu attēlu iegūšanu un apstrādi. Līdz ar to, katru reizi 30 sekundēs datorprogramma veic katra attēla kadra apstrādi, nosakot objektu raksturlielumus reālajā laikā. Zemāk dotā koda rinda saglabā katru video kameras kadru mainīgajā „frame”.

```
Image<Bgr, Byte> frame = _capture.QueryFrame().Clone();
```

Objektu atpazīšanai ir svarīgi nodrošināt objektu labu redzamību attēlā. Šajā darbā ir jānolasa taisnstūrveida kastītes, kas pārvietojas pa konveijera lentu. Kastītes atšķiras viena no otras pēc pozīcijas, krāsas, pagrieziena leņķa un platības. Konveijera lenta ir melnā krāsā. Līdz ar to krāsainās kastītes var būt ļoti novērojamas uz melna konveijera lentes fona.

Kameras izšķirtspēja ir 640x480 pikseļi, no kuriem 640 ir x ass, bet 480 ir y ass. Objekti pārvietojas pa konveijeru x ass virzienā. Objekta centrs var atrasties jebkurā pozīcijā y ass virzienā. Katra objekta garums ir 150 pikseļi, bet platums 100 pikseļi.

Attēls, kas nāk no video kameras, sākotnējā variantā tiek automātiski apstrādāts ar video kameras palīdzību. Šo automātisko apstrādi ir jāizslēdz, lai padarītu attēlu krāsas viendabīgākas. Tomēr tāpat attēls satur digitālos trokšņus, kas rodas mākslīgā gaismas avota iedarbības rezultātā. Šos digitālos trokšņus var noņemt, pamainot attēla kontrastu, spilgtumu, toni un citus parametrus. Tas var būt izdarāms no pašas datorprogrammas, jo Emgu CV bibliotēkā ir iekļautas metodes, kas dod iespēju mainīt video kameras attēla iestatījumus. Eksperimentāli mainot dažādus parametrus tiek iegūts normalizēts attēls, kur ir asi kontūri starp objektiem un fonu. Zemāk dotais kods iestatāta video kameras parametrus.

```
_capture.SetCaptureProperty(CV_CAP_PROP_EXPOSURE, parameters.exposure);
_capture.SetCaptureProperty(CV_CAP_PROP_GAIN, parameters.gain);
_capture.SetCaptureProperty(CV_CAP_PROP_BRIGHTNESS, parameters.brightness);
_capture.SetCaptureProperty(CV_CAP_PROP_CONTRAST, parameters.contrast);
_capture.SetCaptureProperty(CV_CAP_PROP_SATURATION, parameters.saturation);
_capture.SetCaptureProperty(CV_CAP_PROP_WHITE_BALANCE_BLUE_U,
parameters.whiteBalance);
```

Nākamais posms skar attēla apstrādi, jo ir nepieciešams veikt katrā attēlā objektu kontūru noteikšanu. To palīdz izdarīt Emgu CV iebūvētā metode, kas automātiski apstrādā attēlu un visos kontūros starp objektiem un konveijera lentu saliek punktus, tādā veidā katra objekta kontūrs tiek pārvērsts punktētās līnijās. Zemāk dotais kods atrod objektu kontūrus attēlā.

```
MemStorage stor = new MemStorage();
Contour<Point> contours =
frame2.FindContours(CHAIN_APPROX_METHOD.CV_CHAIN_APPROX_SIMPLE,
RETR_TYPE.CV_RETR_TREE, stor);
```

Katrs kontūrs satur ļoti daudz punktu, ar atstatumu ne vairāk par 5 pikseliem starp punktiem. Šos punktus tālāk ir jāapstrādā tā, lai katrs taisnstūrveida objekts saturētu tikai četrus punktus, pa vienam punktam uz katru stūri. Katra kontūra punktu skaita samazināšanu palīdz veikt cita Emgu CV bibliotēkas metode, ar kuras palīdzību punktu kopa tiek optimizēta, un tie punkti, kas atrodas vienā taisnē starp jebkuriem diviem citiem punktiem, tiek izņemti no kopējā punktu saraksta. Tādā veidā visi starppunkti tiek izņemti, un rezultātā katrs kontūrs satur tikai 4 punktus. Katra objekta 4 punkti tiek glabāti atsevišķos List<Point> sarakstos. Zemāk dotais kods izveido optimizēto kontūru punktu sarakstus, analizējot kontūrus ar lielu punktu skaitu.

```
List<List<Point>> contourPoints = new List<List<Point>
contourPoints.Add(new
List<Point>(contours.GetConvexHull(ORIENTATION.CV_CLOCKWISE).ToArray()));
```

Nākamajā posmā ir jānosaka objekta pozīcija uz konveijera lentes. Lai to noteiktu, var izmantot objekta kontūra punktus, atrodot centru starp šiem punktiem. Centra atrašana tiek veikta, nosakot minimālās un maksimālās punktu koordinātes, un tad, saskaitot un izdalot tās ar 2, tiek iegūta to vidējā vērtība. Datorprogramma dod iespēju veikt nepārtrauktu sekošanu pēc objektiem pa visu attēlu, tomēr tas pavisam nav nepieciešams. Pietiek nofiksēt tikai to pozīciju tad, kad tie pārvietojas tieši pāri attēla centram. Šajā momentā ir jāpievieno šī objekta pozīcija kopējā pozīciju sarakstā. Lai viens un tas pats punkts netiktu atkārtoti pievienots katros nākamajos kadros, ir jāveic punktu nepārtraukta salīdzināšana, lai tikai unikālie punkti varētu būt pievienoti kopējā punktu sarakstā. Tiek paredzēts tas, ka jauns punkts katrā iepriekš pievienotajā pozīcijā var būt pievienots tikai pēc noteikta laika intervāla. Pēc šī laika intervāla iepriekš pievienotie punkti maina savu trigeru stāvokli un dod iespēju to y ass pozīcijā pievienot jaunus punktus.

Lai noteiktu objekta platību, var izmantot metodi, kura var apstrādāt iepriekš atrastos kontūrus un izvadīt kontūra platību skaitliskā vērtībā. Šajā gadījumā tiek pielietota trigonometrija, kad starp katriem trim punktiem tiek aprēķināts laukums un pēc tam visi laukumi tiek kopā saskaitīti. Zemāk dotais kods dod iespēju noteikt kontūra platību.

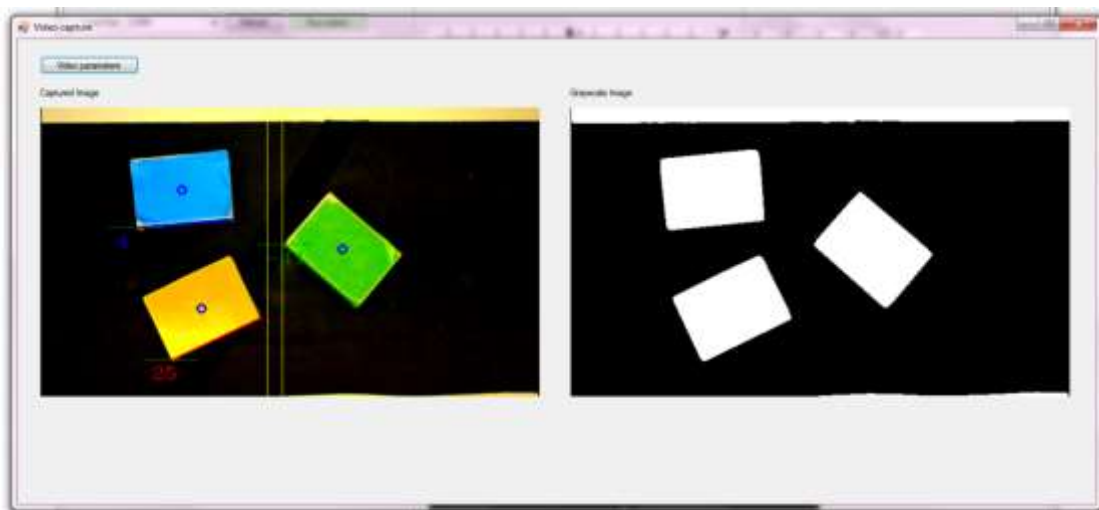
```
contour.ContourArea();
```

Lai atrastu objektu pagrieziena leņķi starp apakšējo garāko objekta malu un video kameras x asi, ir nepieciešams atrast tos divus kontūra punktus, kas atrodas zemāk par pārējiem. Kaut arī katrs objekts satur 4 punktus, nav iespējams viennozīmīgi pateikt, kuri no šiem punktiem atrodas attēla apakšā vai augšā. Līdz ar to ir jāveic vairākas pārbaudes, kuras atlasīs apakšējos punktus pa taisnstūra garāko malu, un tikai tad var aprēķināt objekta pagrieziena leņķi. Zemāk dotais kods dod iespēju aprēķināt objekta pagrieziena leņķi.

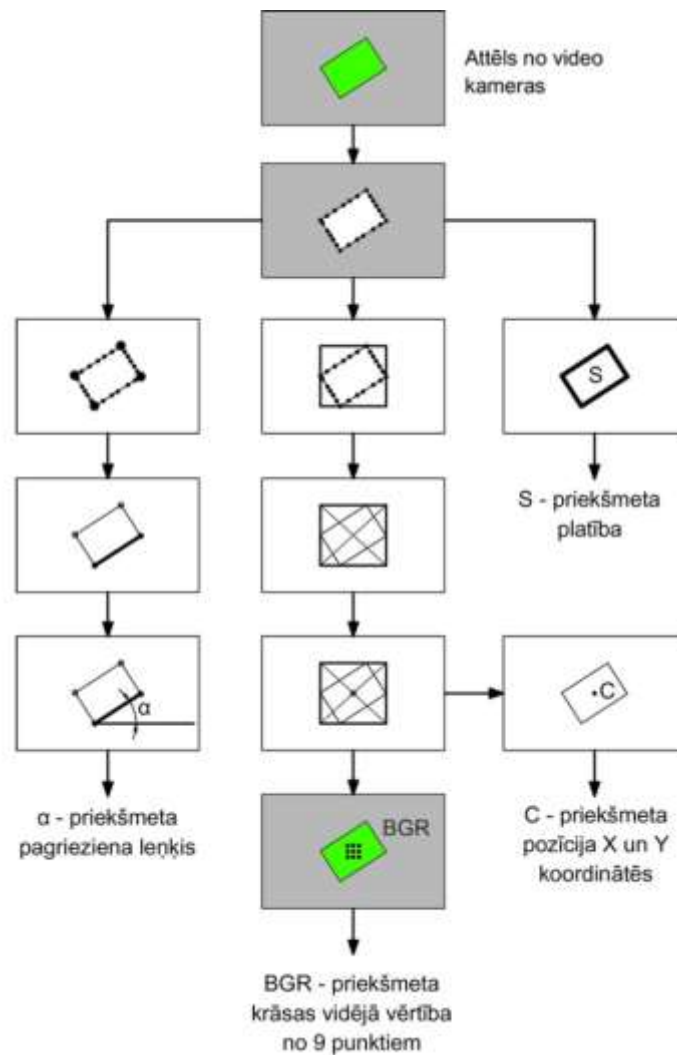
```
double angle = (Math.Atan2(p1.Y - p1.Y, p1.X + 5 - p1.X) - Math.Atan2(p4.Y - p1.Y, p4.X - p1.X)) / ((float)(Math.PI / 180f));
```

Lai noteiktu objekta krāsu, ir jāizmanto objekta centra pozīcija. Eksperimentāli ir noteikts, ka objekta krāsa dažreiz var saturēt nelielus digitālos trokšņus, kas ietekmē krāsu spilgtumu. Līdz ar to kvalitatīvāki rādījumi ir tad, ja izmanto vidējo krāsu starp 9 punktiem ap šo centra pozīciju. Šī krāsa tiek salīdzināta ar standartkrāsām un tiek noteikts tās nosaukums.

Izpildot visas augstāk minētās darbības, tiek iegūti visi nepieciešamie objekta raksturlielumi. Tā kā objekti pa konveijera lentu nepārtraukti pārvietojas ar konstantu ātrumu, tad katrā jaunajā attēla kadrā ir jāveic katras iepriekš atrastās pozīcijas x vērtības palielināšana, lai delta robota satvērējs varētu zināt to, kurā momentā tam ir jāsatver objekts un kur šis objekts atrodas. Apstrādāts video kameras attēls tiek dots 2. att. kreisajā daļā, bet normalizēts attēls ar asiem kontūriem tiek dots 2. att. labajā daļā. Visu algoritmu darbības principu kopsavilkums tiek dots 3. att.



2. att. Apstrādāts un normalizēts video kameras attēls



3. att. Objektu raksturlielumu noteikšanas algoritmi

Rezultāti un to novērtējums

Izstrādātie datorprogrammas algoritmi dod kvalitatīvus rezultātus veicot objektu atpazīšanu. Kaut arī algoritmi ir paredzēti tieši taisnstūrveida objektu satveršanai, tie arī pareizi strādā ar kvadrātveida un apaļiem objektiem. Tika arī eksperimentēts ar trīsstūrveida objektiem, tomēr ar tiem vairākos gadījumos radās problēmas, jo satvērējs nevarēja tos kvalitatīvi satvert.

Iespēja satvert arī apaļos objektus radās tāpēc, ka algoritms apaļajā kontūrā atrada nevis 4 stūra punktus, bet apmēram 16, kas bija vienmērīgi salikti pa visu kontūru. Tādā veidā tika precīzi noteikts apaļā objekta centrs, krāsa un platība. Bet pagrieziņa leņķis šajā gadījumā bija ap 0 grādiem, tāpēc satvērējs to satvēra kā jebkuru taisnstūrveida objektu, kas ir horizontāli novietots.

Pēc eksperimentu datiem tika noteikts tas, ka objekta krāsa attēla vidū pa visu x asi ir spilgtāka, nekā tā ir attēla apakšā vai augšā. Līdz ar to programmā tika pievienota metode, kas automātiski paaugstina krāsas spilgtumu, ja objekts atrodas tuvāk apakšai vai augšai. Šādā veidā tika kompensēta krāsu daudzveidība un tās kļuva viendabīgākas. Krāsu daudzveidību var arī samazināt, ja mākslīgos gaismas avotus novietot nevis x ass virzienā, bet gan y ass virzienā. Tādā veidā visa centrālā daļa būs vienādā spilgtumā un, līdz ar to, visu objektu krāsas būs viendabīgākas.

Secinājumi

Dotajā darbā ir detalizēti paskaidroti datorprogrammas algoritmi, ar kuru palīdzību delta robota stacija var automātiski veikt taisnstūrveida objektu atpazīšanu uz konveijera lentes. Rezultātā tiek iegūti algoritmi, kas veic visu nepieciešamo objektu raksturlielumu noteikšanu un papildus arī dod iespēju veikt kvadrātveida un apaļo objektu raksturlielumu noteikšanu. Izstrādātos algoritmus var pielietot vairākās nozarēs, piemēram, konfekšu vai kancelejas preču šķirošanā.[6][7] Tehniskās redzes pielietošana būtiski atvieglo monotonu darbu, it īpaši tad, kad tehnisko redzi izmanto kopā ar robotiem, kas var vairākkārtīgi paaugstināt procesu veikšanas ātrumu un precizitāti.[8]

Literatūra

1. Ayub, M. A., Mohamed, A. B., Esa, A. H. (2014). In-line Inspection of Roundness Using Machine Vision. Malaysia: Faculty of Mechanical Engineering.
2. Chauhan, V., Surgenor, B. (2015). A Comparative Study of Machine Vision Based Methods for Fault Detection in an Automated Assembly Machine. Canada: Department of Mechanical and Materials Engineering Queen's University.
3. Do, Y. (2012). Intelligent Worm Sorting Using Robot Vision. South Korea: Major of Electronic Control Engineering.
4. Emgu.com (2015). Main Page. http://www.emgu.com/wiki/index.php/Main_Page, sk. 10.03.2017.
5. Fanuc.eu (2015). Vision functions for industrial robots. <http://www.fanuc.eu/es/en/robots/accessories/vision>, sk. 10.03.2017.
6. Wu, W., Wang, X., Huang, G., Xu, D. (2015). Automatic gear sorting system based on monocular vision. China: Research Center of Precision Sensing and Control.
7. Yuan-yuan, T., Si-yang, L., Qing-chang, T. (2012). Application of Detecting Part's Size Online Based on Machine Vision. China: School of Mechanical Engineering.
8. Zhang, W., Mei, J., Ding, Y. (2012). Design and Development of a High Speed Sorting System Based on Machine Vision Guiding. China: School of Mechanical Engineering Tianjin University Tianjin.