

Szakdolgozat

Varga Zoltán

Debrecen

2009

Debreceni Egyetem
Természettudományi és Technológiai Kar
Szilárdtest Fizika Tanszék

Felderítő Robotok
Képfelismerésen alapuló helymeghatározás

Témavezető:

Dr. Szabó István

Egyetemi Docens

Tanszékvezető

Készítette:

Varga Zoltán

Mérnök informatikus

Debrecen

2009

1. Bevezetés.....	5
2. Robotika	6
A robotokról általánosan	6
A robotok történelme, robotgenerációk.....	6
Ipari robotok	8
Kutatásban használt robotok	9
Molekuláris és nanobotok	10
Felderítő robotok	10
Mobilitás, autonómia.....	11
3. Robotok irányítása	12
Látás	12
Sztereo látás	12
Gépi látás (Machine Vision).....	14
Távolság meghatározás	16
4. Képfeldolgozás, Képfeldolgozásról általánosan.....	17
Miért alkalmazunk digitális képfeldolgozást? Mi is az a képfeldolgozás?	17
Képfeldolgozási területek.....	18
Alapfogalmak a digitális képfeldolgozás területén	19
A digitális kép definíciója.....	19
A digitális kép tulajdonságai	20
Képek megjelenítése	21
Képfeldolgozás és elemzés	21
5. Megvalósítás	22
Labview	22
Labview Vision	22
Alapfogalmak	23
Korreláció	23
Mintaillesztés	25
IMAQ Match Pattern 2	26
IMAQ Setup Learn Pattern 2	27

IMAQ Overlay függvénycsalád	28
IMAQ Overlay Text	29
Draw Pattern Match Position	30
6. Program.....	30
Gépigény	31
Futtató környezet	31
Könyvtárszerkezet létrehozása	31
Programegységek	34
Mintakészítő Vi	34
Tavolság Vi	35
7. Összefoglalás	43
8. Irodalomjegyzék	44
9. Köszönetnyilvánítás.....	46

1. Bevezetés

Mobilis robotok ismeretlen környezetben történő navigációjához szükséges valamilyen eljárás a környezet feltérképezésére. Kell egy módszer, melynek segítségével modelt alkothatunk a robotot körülvevő objektumok geometriájáról, ami alapján a robot vezérléséről döntéseket hozhatunk. Ez a modellalkotás történhet a környezet letapogatásával, valamilyen aktív eszköz segítségével, illetve passzív eszközökkel, azon belül is egykamerás, vagy többkamerás rendszerek segítségével.

Jelen esetben a cél képfelismerésen alapuló távolság meghatározás, így a tájékozódást egy kamera segítségével, a nagyítás mértékén alapulva valósítom meg. Adott a feladat, hogy akadályok vannak elhelyezve a kamerától különböző távolságokban és helyzetekben, ezek alapján meg kell határozni a távolságot, és lehetőség szerint az akadályokhoz viszonyított elhelyezkedést, valamint a tárgyakat és a robotot (esetünkben kamerát) el kell helyezni egy koordináta-rendszerben.

Korunk nagy teljesítményű számítógépeinek köszönhetően robbanásszerű fejlődés figyelhető meg a képfeldolgozás területén. Az ott megszerzett ismeretek segítségével szeretném feladatomat megvalósítani.

Munkám során különböző kockák képezték a tárgyak szerepét. A különbség a kockák mintázatában volt. A program tesztelését két kockával végeztem el.

A tárgyak azonosítására elsősorban az IMAQ Vision szoftvercsomag függvényeit használtam, hisz segítségükkel viszonylag egyszerűen megoldhatóak a felmerülő problémák.

2. Robotika

A robotokról általánosan

Kezdjük talán a leglényegesebbel, mi is az a robot? Magának a robot szónak többféle jelentése, definíciója létezik. A műszaki és technikai fogalomkörben általában olyan eszközt vagy berendezést értenek roboton, amely az ember fizikai és/vagy szellemi munkájához hasonló tevékenységet végez, teljesen ember által készített szerkezet, amely önálló mozgásra képes, több szabadsági fokkal rendelkezik valamint újraprogramozható.

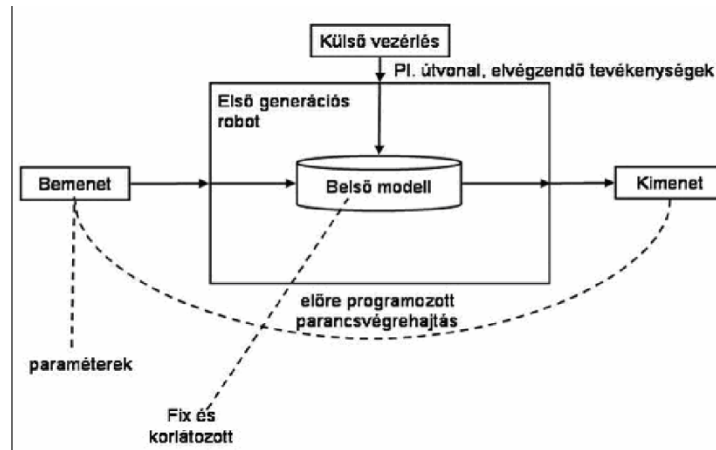
A robotika két okból is kihívást jelentő terület:

- A robotoknak a fizikai világban kell tevékenykedniük, ami sokkal bonyolultabb, mint a legtöbb szimulált szoftvervilág.
- Olyan hardvert (érzékelőket és beavatkozókat) igényel, amelyek valóban működnek.

A robotika részfeladatai: Robot-szerkezetek megépítése. A robot céljának meghatározása. A robot képes legyen tájékozódni valamint, érzékelje a környezetét különböző szenzorok segítségével [1].

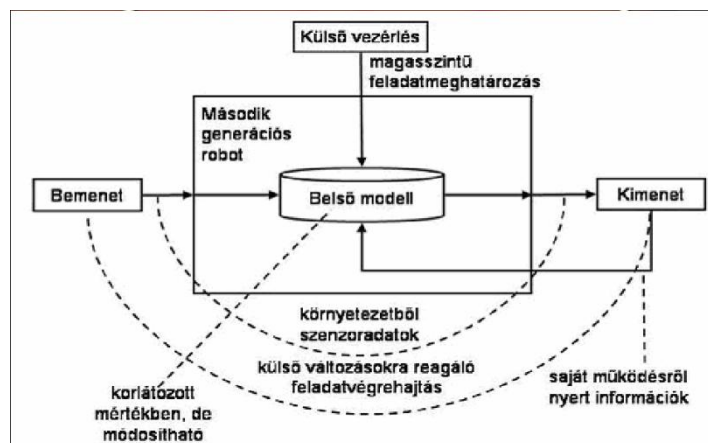
A robotok történelme, robotgenerációk

Az első iparban használt robot 1961-ben állt szolgálatba az amerikai General Motors öntödéjében és a Unimate névre keresztelték. Motorizált karjával a legkülönbözőbb, emberek számára csak nehezen vagy sehogy sem kivitelezhető feladatokat teljesített. Olyannyira sikeres volt, hogy 1969-ben már húsz robotsor végezte a ponthegeztéseket. Az I. generációs robotokhoz tartozott, melyeket főként fizikai feladatokra, leginkább anyagmozgatásra használtak. Ezek a robotok még nem rendelkeztek környezetvizsgáló szenzorokkal, általában csak végezték a feladatokat, amit tervezőjük előre meghatározott.



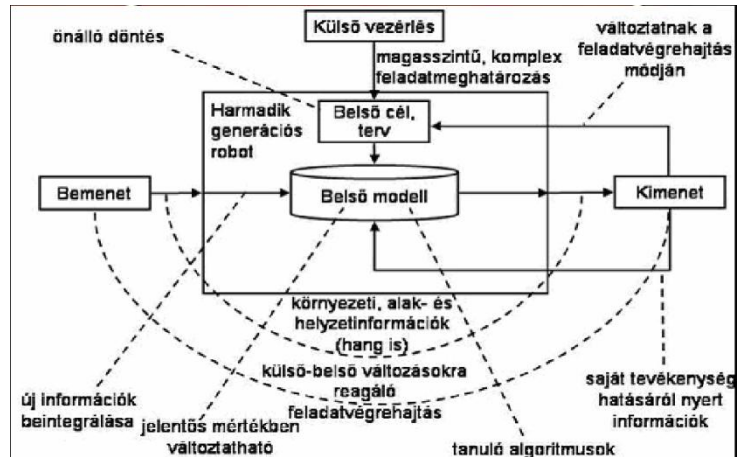
1. ábra: Az I. generáció modellje

A robotok II. generációja, ami az ipari robotok jelentős részét adta a hetvenes években tűnt fel. Ezeknek feladatait már magas szintű programozási nyelveken határozzák meg, szenzoraik segítségével vizsgálják környezetüket valamint bármikor képesek tevékenységük megváltoztatására.



2. ábra: A II. generáció modellje

A következő, vagyis a III. generációs robotoknál immár egyértelmű a mesterséges intelligencia térhódítása. Ezek a robotok már fel is dolgozzák a szenzoroktól érkező jeleket, önállóan változtatnak a magukról és környezetükről tárolt modellen, fel tudják használni az információkat. Jellemzően bonyolult és összetett tevékenységet végeznek.



3. ábra: A III. generáció modellje

Különböző robottípusok léteznek, melyeket alkalmazásuk szerint csoportosíthatunk.

- Ipari alkalmazás szerint:
 - technológiai feladatokat ellátók
 - anyagmozgatók
 - szerelők
- A kutatásban használtak:
 - általános mobil robotok, telerobotok
 - állatok gépi modelljei (animatók)
 - emberszerű robotok (humanoidok)
 - emberarcú robotok (androidok)
- Speciális feladatokra használt robotok:
 - gyógyászatban alkalmazott robotok
 - mikrorobotok, nanorobotok

Természetesen a csoportok között léteznek különböző átfedések. Ez elkerülhetetlen, hiszen egyes gépek több rendeltetésnek is megfelelnek, például a háztartási asszisztensek, szórakoztató célú robotok –focisták és gépállatkák-, valamint a katasztrófa sújtotta térségekben a túlélők után kutató kereső- és mentőrobotok.

Ipari robotok

Becslések szerint a robotok 95-98 %-át az ipari robotok adják. Túlnyomó többségük meglehetősen "buta" - ismétlődően hajtanak végre alapvető cselekedetsorokat. Munkájukat

azért végzik jól, mert a munkakörnyezetük könnyedén előre megtervezhető, rendszerint semmi váratlan nem történik, továbbá minden ott van, ahol lennie kell, a robotnak nem kell megkeresnie semmit, ami a munkához nélkülözhetetlen lenne. Ezek a robotokat a legkülönbözőbb iparágak hasznosítják. Általában csuklós karból állnak, aminek a végén egy készülék, vagy végbeavatkozó, például hegesztő található. Legtöbbjük stabil alapokhoz rögzített, így helyváltoztató mozgásokat igénylő feladatokra nem használható. Fejlesztésüket főként műszaki s nem a mesterséges intelligencia tárgykörébe tartozó kérdések határozzák meg. A technológiai robotok a leghagyományosabbak: futószalag mellett, autógyártásban dolgoznak, tisztítanak, festenek, hegesztenek, de mezőgazdasági munkát is végeznek, például gombákat válogatnak. Az anyagmozgatókat szintén hagyományosan az iparban alkalmazzák. A szerelő robotok alkatrészeket azonosítanak és illesztnek össze. Sok esetben a karok olyan tulajdonságokkal rendelkeznek, melyek lehetővé teszik az olyan munkákat melyek az ember számára veszélyesek.

Kutatásban használt robotok

Az általános autonóm robotok mobilak, feladat-specifikusak, nem szimulálnak állati, vagy emberi viselkedést, általában megfelelnek az intelligens robotokról alkotott általános elképzeléseknek. Előfordul, hogy emberek működtetik őket távirányítással (telerobotok), azonban célszerűbb, ha valamilyen szintű autonómiával rendelkeznek, mert csak így képesek pontosan navigálni, elkerülni az akadályokat, rutindöntéseket hozni, váratlan szituációkban (előre nem jelezhető helyzetekben) helytállni. Alkalmazási területük rendkívül változatos: futárfeladatokat látnak el, rutinmunkákat végeznek kockázatos környezetben (például a Marson, vagy egy atomerőműben), objektumokat őriznek, veszélyes anyagokat gyűjtenek össze, taposóaknákat derítenek fel; lehetnek személyzet nélküli szárazföldi, vízi és légi járművek is. Mozgómechanizmusuk alapján kerekeseket, (mesterséges lábakat használó) lépegetőket és mászókat különböztetünk meg. Az animatok feladata állatok gépi modellezése, valamint az állatok mozgásának elsajátítása, e mozgások előnyeinek kihasználása a különböző alkalmazásokban. Legtöbbjük rovarszerű hatlábú, de sok különbözőt fejlesztenek. Az elmúlt években egyre több kétlábú, emberhez hasonló humanoidot hoztak létre (Honda Asimo, vagy a Toyota trombitása és még sokan mások). Távlabbi cél, hogy ezek a robotok olyan kölcsönhatásban álljanak környezetükkel, mint maga az ember, ami hosszútávon az emberi intelligencia jobb megértése mellett általános rendeltetésű intelligens rendszereket

eredményezhet. (Az emberszerű robotok fejlesztésében az MIT Számítástudományi és Mesterséges Intelligencia, illetve Médialaboratóriuma jár az élen.) Az androidok fejlesztése jelenleg még inkább akadémikus jellegű, konkrét alkalmazásaik inkább látványos kuriózumok, semmint a tényleges hasznosíthatóság. Elképzelhető, hogy a jövőben olyan feladatkörökben számítanak rájuk, mint például a recepciós munka [2,3].

Molekuláris és nanobotok

A gyerekcipőben járó nanotechnológia rendeltetése és célja egyrészt nanométer nagyságrendű tárgyak manipulálása, olyan robotok előállítása melyekkel nanométer nagyságú objektumok manipulálhatók, továbbá olyan mesterségesen előállított kémiai szerkezetek, melyek a szervezetbe kerülve meghatározott feladatokat hajtanak végre (a gyógyszer és a műtét együttese) akár emberi életet megmentve. A nanostruktúrák az orvostudomány és a biokémia termékeinek, mesterséges enzimeinek tekinthető melyek programozhatók és irányíthatók. Tökéletesen működő nanorobotot még nem sikerült előállítani.

Felderítő robotok

A legtöbb felderítő robotokat arra a célra tervezik, hogy olyan környezetben működjenek, amely az élő szervezet, így az ember számára elviselhetetlen. Képesnek kell lenniük bármilyen terepen haladni és bármilyen klímát elviselni, miközben feltérképezi a terület domborzatát, geológiai és kémiai felépítését, valamint rögzít is minden erre vonatkozó információt. A felderítő robotok általában fel vannak szerelve rezgés, mozgás és elektromos szenzorokkal, optikai kamerákkal, infravörös érzékelőkkel és sugázmérőkkel. Valószínűleg a leghíresebb felderítő robotok a Mars-kutatásban orosz-lánrészt vállaló Spirit és Opportunity.



4. ábra: Egy Mars járó robot

Mobilitás, autonómia

Az autonóm robotok külső irányítótól függetlenül is képesek a cselekvésre. Programozásukat az a cél vezérli, hogy a külső hatásokra valamilyen módon reagáljanak. A fejlettebbek sztereo látórendszert használnak: a térbeli érzékelést két kamera, a tárgyak lokalizálását és osztályozását képfelismerő szoftverek biztosítják. A környezet elemzésére használhatnak mikrofonokat és szagérzékelőket is. Egyesek, például a mindennapi életben már megjelent – Magyarországon is kereskedelmi forgalomban lévő – porszívók és a fűnyírók csak megszokott, alaposan körülhatárolt területen működnek, míg mások ismeretlen, nehéz terepeket is elemeznek, és bizonyos terepmintákat bizonyos cselekvésekhez asszociálva alkalmazkodnak hozzájuk. De létezik a véletlenszerűségeen alapuló alternatív tervezői koncepció is: ha probléma merül fel, a robot addig kísérletezik, míg valamelyik próbálkozása szerencsével nem jár.

A való világban teljesen autonóm robotnak a következő adottságokkal kell rendelkeznie:

- Információt tudjon szerezni a környezetéről
- Képes legyen folyamatos munkavégzésre emberi beavatkozás nélkül
- Képes legyen A pontból B pontba jutni, vagyis a helyét megváltoztatni emberi beavatkozás nélkül
- Külső beavatkozás nélküli önjavítás

A robot autonóm tanulás képességével is rendelkezhet, amely az alábbiakból tevődik össze:

- Külső beavatkozás nélkül tanul meg újabb dolgokat

- A környezeten alapuló stratégiát dolgoz ki, a meglévőket újabbakkal egészíti ki
- Külső beavatkozás nélkül alkalmazkodik a környezethez, környezetváltozásokhoz

A robotok sokféle módon irányíthatóak, én azonban csak egy témával foglalkoznék részletesebben, ami nem más, mint a képfeldolgozáson alapuló robotvezérlés kamera segítségével.

3. Robotok irányítása

Látás

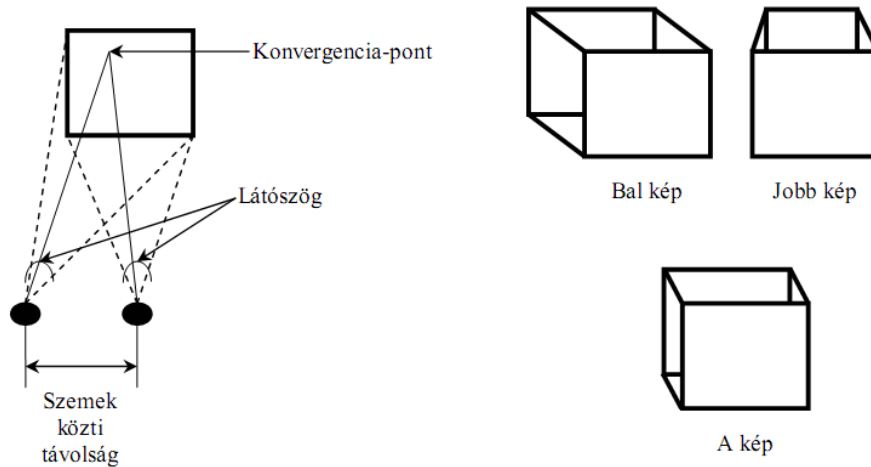
A szem összetett optikai rendszer, amely a tárgyakról az ideghártyán hoz létre valódi, fordított, kicsinyített képet. Egészséges szemmel a különböző távolságú tárgyak éles leképezése a szemlencse fókusztávolságának változtatásával biztosítható. Legtöbbünk számára a látás normális, és hozzá vagyunk szokva, hogy a dolgokat ránézés alapján értékeljük ki. A látás azonban különböző és összetett struktúrákat alkalmaz, melyek nem csak a képek felfogását teszik lehetővé, hanem egyidejű feldolgozásukat és értelmezésüket is. A szem retinájára kivetített képek idegi impulzusokká alakulnak, melyek viselkedésmódot hoznak létre és befolyásolnak: ezt nevezzük látásnak. Így amit látunk, az nem feltétlenül egyezik meg azzal, amit egy másik ember lát.

Mindezekből következik, hogy a gépi látás megvalósítása, vagyis egy olyan program összeállítása mely segítségével önálló mozgásra képes robotot alkothatunk, amely az emberi látáshoz hasonló módon képes tájékozódni nagyon bonyolult feladat, amit eddig még nem sikerült teljes egészében megoldani. Robotok irányításának egyik módja a sztereo látáson alapul.

Sztereo Látás

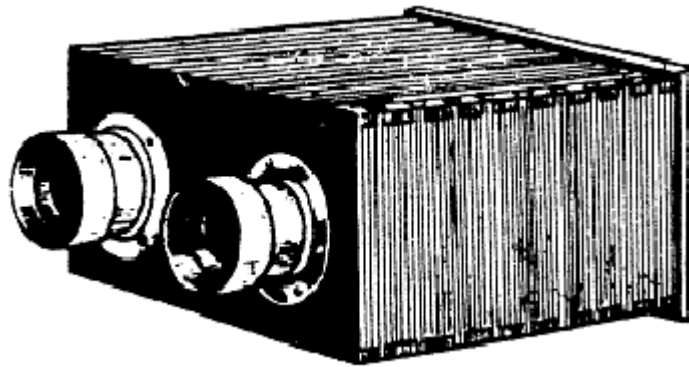
Mi is az a sztereo látás? Az ember két szemmel lát. A két kép tehát hasonló, de nem teljesen azonos. Idegrendszerünk az eltérések alapján állapítja meg a tárgyponatok hozzánk viszonyított távolságát, vagyis a mélységét. Tudjuk tehát, hogy minek alapján teszi ezt. Sajnos az, hogy miként, milyen „számítások útján” valósítja meg, az sokkal bonyolultabb. Ha a dolog nem

lenne ilyen bonyolult, akkor mostanra minden bizonyosan már térlátású robotok vennének körbe minket.



5. ábra: A sztereo, vagyis a térbeli látás

A sztereo megjelenítés úttörője az Charles Wheatstone angol fizikus volt. A londoni Royal Society-ben 1833. június 21-én tartott előadást a sztereo látással kapcsolatos felfedezéseiről. Két, a nézőhöz viszonyítva +45 illetve -45 fokban elhelyezkedő tükörrel érte el azt, hogy mindkét szem csak a neki szánt képet láthassa. A képek egyszerű vonalas ábrák voltak, és a sztereo élmény annak ellenére előállt, hogy nem volt rajtuk se takarás, se árnyékolás, se egyéb mélység meghatározást segítő dolog. Daguerre 1839 augusztusában közzétett fényképezési módszerének elterjedésével immár sztereo fényképek készítésére is lehetőség nyílt. Addig csak sztereo rajzokat alkalmaztak. David Brewster angol fizikus továbbfejlesztette a sztereo fényképezési módszert, és 1849-ben elkészítette az első kétlencsés teljesen sztereo kamerát. 1855-ben a francia Barnard elkészítette az első lencse-előtétként használható tükrös sztereo-előtétet, amelyet az angol Brown továbbfejlesztett. A sztereo nézőkészüléket a német Helmholtz és Pulfricht fejlesztették tovább. Wheatstone tükrös sztereoszkópja nem terjedt el eredeti formájában, helyette az amerikai Oliver Wendell Holmes és Joseph Bates (1862) által továbbfejlesztett „Holmes Stereo Viewer” vált évtizedekig népszerűvé és kvázi világszabvánnyá.[4]



6. ábra: Az első kétlencsés sztereo kamera.

Gépi látás (Machine Vision)

Robotok hatékony irányítására többféle módszer létezik, az általam használaton kívül.

Az utóbbi években, évtizedekben a drasztikusan megnövekedett teljesítményeknek köszönhetően számtalan feladatot tudunk számítógépeinkkel elvégeztetni. Mivel napjaink számítógépei olyan sokféle és nagyméretű adathalmazokat képesek feldolgozni és kielemezni, felmerül a kérdés képesek-e látni? Meg tudja-e mondani egy kameráról betöltött képről vagy videóról a számítógép, hogy az mit ábrázol? A kép- és alakzat-felismerés egyike azoknak a területeknek, amelyeken az ember sokkal tökéletesebb, mint a gép. Ezen területeken felmerülő problémákra keresi a választ a gépi látás (machine vision) tudományterület. Jelenleg az, hogy számítógépeink megértsék, mit ábrázolnak a képek nincs lehetőségünk, egy adott kérdésre azonban sikeresen választ kaphatunk. Mondjuk, ha arra vagyunk kíváncsiak, hogy egy adott tárgy szerepel-e a képen, akkor egy megfelelően elkészített program erre választ tud adni. A gépi látás témaköre igen tág, ezért csak felsorolás szinten említek meg néhányat, bővebben csak a témához kapcsolódó robotok irányításával szeretnék foglalkozni:

- Orvosi képfeldolgozás
- Ipari minőségellenőrzés
- Kamerás megfigyelés
- Robotok irányítása

Felmerülő problémák

A gépnek nagyon sok információra van szüksége:

- az alak pontjainak egymáshoz viszonyított helyzetére,
- viszonyára a környezetéhez,
- térbeli elhelyezkedésére stb.

Ezekből következik, hogy: rengeteg pont koordinátáit kell megadni plusz a színek, árnyalatok, megvilágítási értékek jellemzésére további adatok szükségesek.

Jelenleg a mesterséges, vagyis gépi látás, gyakorlatilag megegyezik a számítógépes képfeldolgozással.

Robotok irányítása gépi látás segítségével

Mélységérzékelés: A mélységérzékelés vagy más néven távolságérzékelés az a képesség, amivel a világot három dimenzióban látjuk. Ennek köszönhetően tudunk pontosan mozogni, távolságokat felismerni, dolgokat megfogni. A retinánkon a környezetünkről egy kétdimenziós, fordított állású, kicsinyített kép jelenik meg. Ahhoz, hogy végül mégis három dimenzióban észleljünk, bonyolult, magasabb szintű folyamatokra van szükség. E folyamat megértése a kulcsa az önálló tájékozódásra képes térlátású robotok megalkotásának.

A robotok önálló tájékozódása már nagyon régóta kutatott terület, melynek eredményei napjainkban kezdenek megjelenni a különféle termékekben. Ezen a területen az alapvető feladatok: guruló, repülő, úszó járművek irányítása, akadályok elkerülése és felismerése vagy az eddig még megoldandó robot vezérelte közlekedés.

Távolság meghatározás

Távolságot meghatározni többféleképpen is lehetséges. Én a geometria optika területét vizsgáltam meg részletesebben, és innen vettem az ötletet, ami alapján sikerült meghatározni a kamera és egy tárgy, estünkben egy kocka közti távolságot.

Ahhoz hogy megfelelően képet kapjunk arról, miként is lehet ily módon távolságot meghatározni, szükségességét érzem egy kis bevezetőt tenni a lencsék képalkotásának módjába.

A lencse képalkotását a leképezési törvény írja le:

$$\frac{1}{f} = \frac{1}{t} + \frac{1}{k}$$

Ahol f a lencse fókusz távolsága, t a tárgytávolság és k a képtávolság. Ha tehát ismerjük a tárgytávolságot és a képtávolságot, a fókusz távolságot könnyen kiszámíthatjuk. Esetünkben a tárgytávolságot szeretnénk meghatározni, ezért az egyszerűség kedvéért én a következő, nagyításra vonatkozó képlet segítségével számoltam ki a távolságot:

$$N = \frac{K}{T}$$

A tárgy és a leképező eszköz távolságát tárgytávolságnak, a kép és a leképező eszköz távolságát képtávolságnak nevezzük. A leképezés a tárgyhoz geometriailag hasonló képet állít elő, amely a tárgyhoz viszonyított állása szerint lehet egyenes állású, vagy fordított állású. A kép nagyságának és a tárgy nagyságának az aránya a nagyítás [5].

Ahhoz hogy meg tudjam határozni a távolságot, először szükség volt a kamera nagyításának meghatározására. A későbbiekben képek kalibrálása című résznél részletesebben kifejtem egy kép kalibrálásának módját, itt csak annyit említenék meg, hogy a nagyítás mértékének meghatározásához szükség van egy kalibrált képre, amely segítségével azt tudjuk mérni, hogy mekkora a képen egy távolság a valós mérethez viszonyítva. Ezt felhasználva, valamint kezdetben ismerve a tárgy távolságát a kamerától, meghatározhatjuk a képtávolságot, amit a kamera használata során kapunk. Tehát:

$$K=N*T, \text{ amiből: } T=K/N$$

Fontos megjegyezni, hogy a kamera képtávolsága néhány tizedet változik a tárgy távolságának függvényében, ezért tizenöt különböző távolságon elvégeztem a szükséges méréseket, és az ez alapján átlagolt számot vettem a képtávolságnak.

E módszer jelentős hátránya, hogy ha más kamerát használunk, akkor sajnos mindig el kell végezni a szükséges méréseket, és képkalibrálási számításokat, viszont ez a módszer a kisebb kellemetlenségektől eltekintve egy elég jó távolság meghatározási mód.

4. Képfeldolgozás, képfeldolgozásról általánosságban

Számítógépes képfeldolgozó rendszereket először az 1960-as években az űrkutatásban és a földfelszínről készített felvételek kiértékelésére alkalmazták. Az azóta eltelt 50 évben azonban a számítógépek kapacitása sokszorosára növekedett, és ezzel egyidejűleg a számítógépes képfeldolgozás alkalmazásai területeinek a köre valamint jelentősége is kiszélesedett. Az internet és a multimédia nagymértékben meggyorsította képfeldolgozáshoz szükséges eszközök fejlesztését és a folyamatosan bővülő felhasználási igények lehetővé tették a gyártási darabszámok növelését, ami jelentős árcsökkenéshez vezetett. Ma már oda jutottunk, hogy otthoni használatra is könnyedén összeállíthatunk olyan számítógépes képfeldolgozó rendszereket, amelyeket tíz-husz évvel ezelőtt a kutatóintézetek is csak igen nagy anyagi áldozatok árán használhattak. Napjainkban a digitális képfeldolgozással szinte az élet minden területén találkozhatunk, azonban most a képfelismerésen alapuló robotirányítással foglalkozom.

Miért alkalmazunk digitális képfeldolgozást? Mi is az a képfeldolgozás?

A számítógépes képfeldolgozás célja és lényege, hogy a környezetünkből származó képi információt a számítógép segítségével feldolgozzuk és kiértékeljük. Ehhez először a valós világ tárgyainak képét számítógéppel kezelhető adatokká kell átalakítani. Ezt az átalakítást nevezzük digitalizálásnak. A digitalizálás logikailag lényegében három lépésből áll, a leképezésből, a mintavételezésből és a kvantálásból, attól függetlenül, hogy a különböző digitalizáló eszközök ezeket a lépéseket nem feltétlenül elkülönítve valósítják meg. A

leképezés azt jelenti, hogy a képkalkoló rendszer – például a kamera vagy a fényképezőgép a háromdimenziós térben lévő tárgyakról érkező elektromágneses jelekből egy kétdimenziós képet hoz létre. A mintavételezés során a kétdimenziós kép egy-egy kis területéhez (képeleméhez) a fényességtől függő számot rendelünk hozzá. A mintavételezett kép már csak a mintavételi helyeken meghatározott szürkeségi értéknek megfelelő számokat tartalmazza. Ezek a számok nem negatív, valós számok lesznek. Az úgynevezett kvantálási művelet a mintavételezett képet alkotó tetszőleges értékű számértékekhez a megengedett világosságkód értékek valamelyikét rendeli. A digitalizálás eredményeként kapott számhalmazt, a digitális képet a számítógép már tudja kezelni, és a programon múlik, hogy ebből a számhalmazból egy adott probléma esetén, mit és hogyan tudunk kiolvasni. Az is nyilvánvaló, hogy a digitális kép minősége, így a további feldolgozás eredményessége szempontjából alapvető fontosságú a digitalizálás. Értelemszerűen a mintavételezés sűrűségének és a kvantálási szintek számának a növelése javítja a digitális kép minőségét, de egyidejűleg a szükséges tároló helyet is növeli. Színes kép esetében a mintavételezés és a kvantálás szín-összetevőnként, egymástól függetlenül valósul meg [6].

Felmerülhet a kérdés, hogy mi értelme van a számítógépes feldolgozásnak? A választ nem lehet egy mondatban megadni, hiszen az indok esetenként más és más lehet, de például:

- A számítógép nem fárad el, ha órákon át ugyanazt a mérést kell elvégeznie.
- A számítógép objektív, azonos körülmények között mindig ugyanazt az eredményt adja.
- Végül a képfelismerés célja a képen rögzített valós világ leírása, a tárgyak felismerése és azonosítása, tudásbázisból már ismert alakzatokkal. Ennek alapján valamilyen feldolgozási célkitűzés teljesítése. A végső cél a látás automatizálása, a vizuális információt közvetlenül megértő, ezzel vezérelt robotok előállítására.

Számunkra ez az utóbbi a leglényegesebb szempont, hiszen számítógépes képfeldolgozó rendszerrel ellátott eszköz oda is elküldhető, ahová ember nem, és itt nem csak úrkutatásra kell gondolnunk, hanem sugárzásnak, vagy káros anyag hatásának kitett helyre is.

Képfeldolgozási területek

A számítógépes képfeldolgozáson kizárólag kétdimenziós (síkbeli), digitális képek gépi feldolgozását értjük.

A digitális képfeldolgozás a természetes képeken foglalt képi információ megértésére irányul. A folyamatot - a rész-célkitűzések szerint - három szintre bonthatjuk.

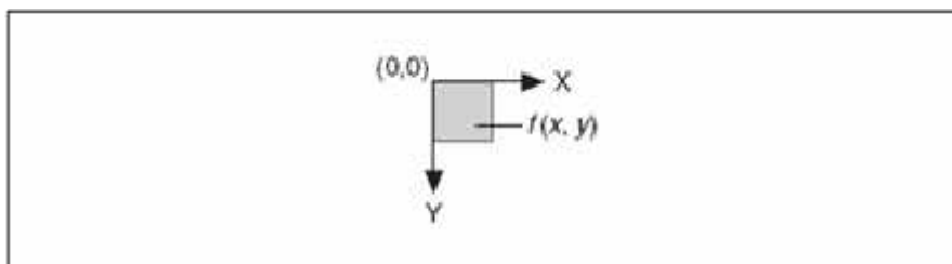
- Elő feldolgozáson a digitális kép létrehozását, valamint a leképzési hibák kijavítását, a jellemző képtulajdonságok kiemelését és a képnek a további feldolgozásra kedvezőbb átalakítását értjük.
- A képosztályozás során megkeressük a képen előforduló alakzatokat, illetve a képet jellemző különféle sajátosságokat. Az ezekhez mindig rendelhető numerikus értékekkel, vagy a köztük lévő relációkkal leírjuk, majd osztályozzuk a képet, illetve egyes részeit.
- Végül számunkra talán a legfontosabb szint a képfelismerés. Célja a képen rögzített valós világ leírása, a képeken szereplő tárgyak felismerése és azonosítása, egy tudásbázisból már ismert alakzatokkal. Ennek alapján valamilyen feldolgozási célkitűzés teljesítése. A végső cél a látás automatizálása, a vizuális információt közvetlenül megértő, ezzel vezérelt robotok előállítás, programozása. Például a feladatban szereplő felderítő robot irányítása kamera segítségével. A robotnak megadhatunk egy alakzatot, tárgyat, majd a robot megkeresi azt a kamerája segítségével és odanavigál.

Alapfogalmak a digitális képfeldolgozás területén

A digitális kép definíciója

A digitális kép nem más, mint fényerősség értékekből álló kétdimenziós tömb, azaz egy mátrix. Egy digitális kép nem más, mint a fényerősség függvénye: $f(x, y)$, ahol f az (x, y) koordinátájú képpont, azaz pixel fényessége. A továbbiakban kép alatt mindig digitális képet értünk. Tehetünk olyan megállapodást, mely szerint a kép $(0, 0)$ koordinátájú pontja a kép

felső bal sarkában található. Mintavételezés során a képszensor mindig diszkrét számú pixellé alakítja az érzékelt képet, majd minden egyes pixelhez hozzárendeli a megfelelő szürkeárnyalatos vagy színes értékét, így meghatározva az egyes képpontok fényintenzitását vagy színét.



7. ábra: A digitális kép (0,0) koordinátája

A digitális kép tulajdonságai

A digitális kép három tulajdonsággal rendelkezik: a kép felbontása, bitmélysége és a komponenseinek a száma. Egy digitális kép nem más, mint egy mátrix. A kép felbontása a sorokban és oszlopokban szereplő pixelek számától függ, egy m oszlopból és n sorból álló kép felbontása: $m \times n$.

A kép bitmélysége nem más, mint a pixelértékek reprezentálásához szükséges méret bitekben megadva. Ha egy kép bitmélysége n , akkor a kép minden egyes pixele 2^n különböző értéket vehet fel. Például ha $n=8$, akkor egy pixel 256 féle értéket vehet fel, 0-255-ig.

A képfeldolgozás során fontos hogy megfelelő bitmélységet válasszunk, mivel bizonyos műveletek során a megnövekedett memóriaigény, és a gyorsabb futási idő miatt felesleges a szükségesnél nagyobb bitmélységű képet használni. Például a feladatomban szereplő objektum alakjának meghatározására, megkeresésére tökéletesen elegendő a 8 bites bitmélység használata, ugyanakkor fényintenzitás mérése során célszerűbb lenne minél nagyobb bitmélység használata.

A komponensek száma megadja a képet felépítő pixelmátrixok számát. Szürkeárnyaltos képek esetén egyetlen komponens létezik, míg színes képek esetén általában három. RGB szintér használata esetén a kép három színkomponensből épül fel, ezek a vörös, zöld és a kék komponensek. Egy n komponensből felépülő kép esetén minden egyes pixel egy egydimenziós vektor lesz.

Képek megjelenítése

A képi megjelenítés általában az egyik legfontosabb rész a képfeldolgozás területén. A képfeldolgozás és a képi megjelenítés egymástól teljesen független terület. Az előbbi a kép létrehozásával és elemzésével kapcsolatos fogalom, míg az utóbbi a kép, mint adat megjelenítésével. Robotok irányításánál nem lenne nélkülözhetetlen a feladat megoldásához a képi megjelenítés. Azonban ha nem jeleníteném meg a kamera képét, akkor a működés közben fellépő esetleges hibákat lehetetlen lenne felismerni. Egy tipikus képelemző alkalmazás is számos meg nem jelenített képet tárol a memóriában.

Labview-ban a képeket kétféle módon jeleníthetjük meg: külső ablakban, vagy a beépített képmegjelenítő kontroll, az Image Display segítségével.

Képfeldolgozás és elemzés

Maga a képelemzés nem más, mint a kép pixeleinek intenzitásértékein végzett mérések és statisztikai számítások. A képelemző függvények segítségével a kép tartalmi tulajdonságait tudjuk elemezni és a további feldolgozás során a kapott eredményektől függően döntést tudunk hozni, hogy mely képfeldolgozó függvényeket használjuk fel [6].

5. Megvalósítás

Mindezek megvalósításához kitűnő fejlesztői környezetek állhatnak rendelkezésünkre. Többek között a National Instruments által kifejlesztett, Labview. Eszközei biztosítják mindazt a teljesítményt és rugalmasságot, amely egy gépi látást és tudományos képalkotást tartalmazó alkalmazások igényelnek.

Következőkben rövid bevezetés a LabView - ba, valamint az IMAQ Vision programcsomagba.

Labview

Labview (Laboratory Virtual Instrumentation Engineering Workbench) maga egy platform és egy grafikus fejlesztői környezet is egyben. A grafikai nyelvet „G”-nek nevezik, ami egy grafikus adatfolyam-programozási nyelv. Eredetileg az Apple Macintosh-ra készült 1986-ban, azóta számos platformra kifejlesztették (Windows, Linux, Unix). A programírás a modulok közti adatkapcsolatok vezetékezéséből áll. Minden program vagy szubrutin (VI, virtuális eszköz) két részből áll, előlapból és hátlapból, vagyis blokkdiagramból. Az előlap a felhasználói felület, beavatkozó szervekkel és kijelzőkkel. A hátlap a kód, az adatáramlási diagram. Az előlapon hozható létre a felhasználói interfész, melynek kialakításakor számos komponens közül válogathatunk. A Labview-ban való fejlesztés alatt az előlapi panel és a blokk diagram szerkesztését értjük [8].

Labview Vision

A képfeldolgozási feladatok megoldásához a Labview hagyományos jelfeldolgozási rendszerét kiterjesztették hardware és software oldalon. Hardware oldalon a megfelelő driverek és csatolóártyák segítségével lehetőség van analóg, USB, firewire és kameranlink csatlakozású kamerák jeleinek fogadására [9]. A speciális képfeldolgozást biztosító függvények a Labview-n belül az IMAQ programcsomagban találhatóak meg. Az IMAQ USB programcsomag segítségével egyszerűen olvashatunk jelet USB portra csatlakoztatott

kameráról. Ezekből, a varázslókkal összeállított képfeldolgozási megoldásokból Labview program generálható, amely így könnyen továbbfejleszhető.

A Labview egy kiegészítő modulja a Vision, ami a képfeldolgozással kapcsolatos függvényeket tartalmaz. A program számos ilyen függvényt használ, így szeretnék ezen beépített függvények közül néhányat ismertetni.

Alapfogalmak:

Az előlapi panel:

Az előlapi panelen elhelyezhető eszközöknek két nagy csoportja van: kontrolok és indikátorok. A kontrolok a bemenő adatok betáplálására, míg az indikátorok a kimenő adatok megjelenítésére szolgálnak.

A blokk diagram:

A blokk diagramon elhelyezhetünk függvényeket, struktúrákat, és SubVi-okat. Ezek az úgynevezett csomópontok, melyeket a programozó huzalokkal köt össze. Egy csomópontnak lehet nulla, illetve egy vagy több bemenete, valamint nulla, egy vagy több kimenete. A csomópontokat összekötő huzalok biztosítják a csomópontok közti adatáramlást. Termináloknak nevezik a csomópontok bemeneti és kimeneti pontjait. Egy csomópont csak akkor hajtódik végre, ha minden adat elérhető a bemeneti oldalon.

Korreláció

A korrelációt akkor használjuk, ha képek közötti hasonlóságot keresünk, vagy a képen egy bizonyos tartomány pontos pozíciójára vagyunk kíváncsiak (mint esetünkben is), esetleg két kép közötti eltolódást, annak irányát és nagyságát kell meghatároznunk. Egy $w \times h$ méretű kernel alkalmazva [(mind w , mind h páratlan és $m=(w-1)/2$, ill. $n=(h-1)/2$ a fél szélesség,

illetve félmagasság)] az f képre, a korrelációs művelet eredménye a következőképp számítható ki:

$$g(x,y) = \sum_{j=-n}^n \sum_{i=-m}^m k(i,j) f(x+i,y+j)$$

Vagyis a k kernel elemei (a konvolúcióval ellentétben) a mátrixelem alatt lévő pixel intenzitásértékeivel szorzódnak. Mivel a műveletek a világos részek fényességét még jobban megnövelik, ezért a fenti kifejezés normált-változatát használjuk [7]:

$$g'(x,y) = \frac{\sum_{j=-n}^n \sum_{i=-m}^m k(i,j) f(x+i,y+j)}{\sum_{j=-n}^n \sum_{i=-m}^m f(x+i,y+j)}$$



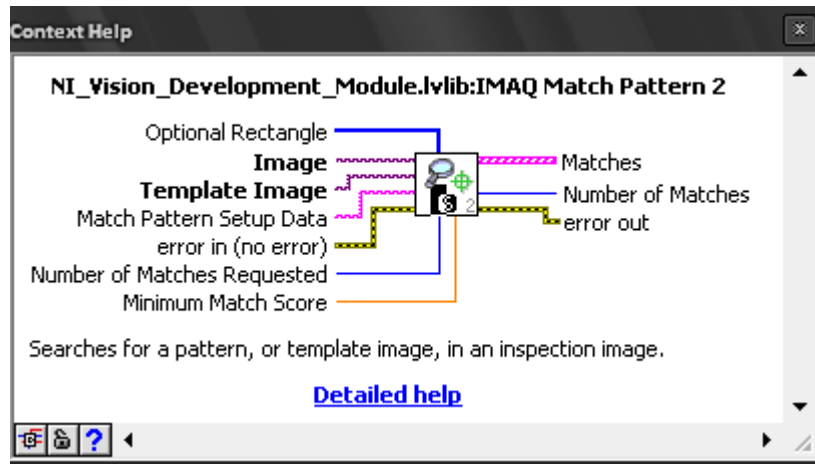
8. ábra: A legjobb illeszkedés koordinátái ($x=149, y=99$).

Mintaillesztés

A mintaillesztés (pattern matching) a képfeldolgozásban egy nagyon gyakran alkalmazott eljárás. Segítségével szürkeárnyalatos képeken tudunk keresni olyan tartományokat, amelyekre az előre definiált mintát - más néven sablont - illeszteni tudjuk. Először a keresendő képi objektumról készítünk egy mintát, majd a mintaillesztő algoritmus a vizsgált képen próbál a mintához minél hasonlóbb előfordulásokat keresni. Ezt pontozással viszi véghez, a nagyobb pontszámú (0 és 1000 között, 1000 a tökéletes illesztés) előfordulások jobban hasonlítanak a keresett mintához. Számos tényező - mint például fényintenzitás, zaj, méret és távolság - befolyásolja a keresés eredményét, ezért nagyon fontos az, hogy ezeket a mintaillesztő algoritmus a sikeres lokalizáció érdekében kezelni tudja. A mintaillesztő algoritmus, ami minden esetben a korrelációs függvény számításán alapszik [6], az egyik legfontosabb függvény a gépi látásban, mert számos terület e függvényen alapul. Az algoritmus főbb felhasználási területei:

- Vizsgálatok: Megadott elemek felkutatása vagy egyszerűen egy olvashatatlan nyomtatvány felismerése.
- Csoportosítás, Osztályozás: Meghatározza egy kiválasztott alakzatnak a pozícióját, ezáltal eltéréseket, elmozdulásokat tudunk vizsgálni.
- Mérés: Távolság, átmérő, forgásszögek mérésére alkalmas. Lehetőség van az objektumok méret szerinti osztályozására is.

Mivel dolgozatom célja a robotvezérlési módszerek bemutatása, megismerése volt, így elkerülhetetlen volt számomra a mintaillesztő függvények használata, hiszen mi más módját is választhattam volna tárgyak felkutatására és azonosítására. A következőkben szeretném röviden bemutatni az általam használt és számomra leghasznosabb mintaillesztési függvényeket.



9. ábra: Az IMAQ Match Pattern 2 függvény leírása

Az IMAQ Match Pattern 2 függvény

Bemenő paraméterek:

- Optional Rectangle: Négyelemű tömb, ami a feldolgozandó területet határozza meg. Ezek sorrendben: bal, fel, jobbra, le. Ha nem adjuk meg ezt a paramétert, akkor a keresés a forrás kép egészére kiterjed.
- Image: A vizsgált kép, amin a keresést végre akarjuk hajtani, amin a mintát keressük.
- Template Image: Az a minta, amit a vizsgálandó képen keresünk.
- Match Pattern Setup Data: Sztring paraméter, ami információkat szállít az IMAQ Setup Match Pattern 2 VI-tól, vagy az IMAQ Advanced Setup Match Pattern 2 VI felől. Ha nem adjuk meg ezt a paramétert, akkor a VI az alapértelmezett paraméterekkel látja el az összes beállítást, ami egy üres sztring konstans.
- Number of Matches Requested: Az a számérték, ami megmondja, hogy mennyi az általunk elvárt találatok száma.
- Minimum Match Score: Ezzel szabályozható a keresés pontossága. Az intervallum 0-tól 1000-ig tart. Magasabb számérték, pontosabb egyezést eredményez.

Kimenő paraméterek:

- Shift Information: Csak a minta eltolt előfordulásait keresi.
- Rotate Information: Csak a minta elforgatott előfordulásait keresi.

Kimenő paraméter:

- Learn Pattern Setup Data: A mintaillesztéshez szüksége beállítások.

IMAQ Learn Pattern 2:

Egy leírást készít a mintaképről, melyet a keresés során a kereső algoritmus használni fog.

IMAQ Setup Match Pattern 2:

Segítségével minden egyes mintát egyéni mintaillesztési beállítással láthatunk el.

IMAQ Match Pattern 2:

Egy tömböt ad vissza, ami a talált egyezések adatait tartalmazza. Ez a tömb természetesen annyi elemszámú lesz, ahány mintával való egyezést kaptunk a képen.

Munkám során azért döntöttem e fajta mintaillesztés mellett, mert a változó fényviszonyok nem kedvezőek a geometriai illesztés használatához.

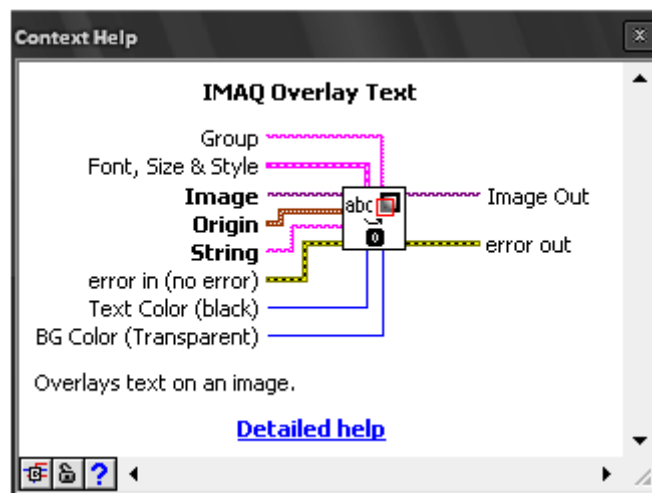
Következzenek hát azon VI-ok, melyek segítségével megjegyzéseket, fontosabb adatokat, paramétereket helyezhetünk el a képeken.

IMAQ Overlay függvénycsalád

Segítségükkel valósítható meg a kijelzőre való írásos szöveg felvitele, alakzatok kirajzolása, keresett minták bekeretezése az egyszerűbb láthatóság érdekében vagy akár címmel is elláthatjuk őket. Távolság, terület mérések során a mért értékeket egyszerűen magára a képre jegyezhetjük, hisz adott esetben sokkal egyszerűbb így leolvasni azokat, mintha egy külön táblázat, esetleg tömb tartozna az értékekhez. Az alakzatok és szövegek színét a fejlesztő tetszőlegesen változtathatja, így ha például két vagy több különböző mintát keresünk egy képen, akkor arra is lehetőség van, hogy minden egyezést különböző színnel keretezzünk be.

IMAQ Overlay Text

Az IMAQ Overlay Text az egyik leghasznosabb beépített függvény az Overlay függvénycsaládból. Fontosabb paramétereit alább ismertetem:

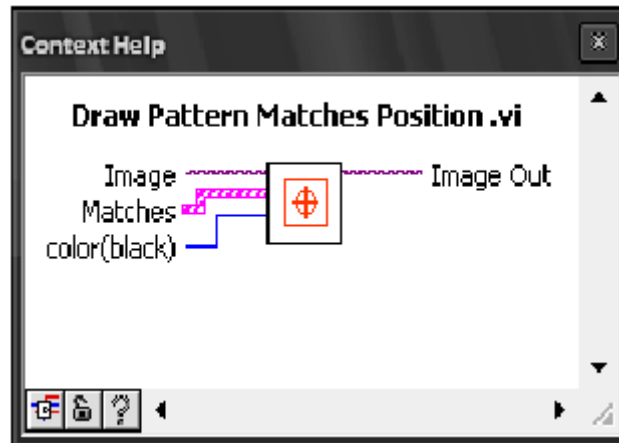


IMAQ Overlay Text függvény leírása

Fontosabb paramétereit:

- Image: A kép, amire a kiíratást végezzük.
- Origin: A képen lévő pozíció, ahová a kiíratást szeretnénk végezni. Vagyis a vízszintes és függőleges pozíciók értékei egy clusterben.
- String: Maga a karaktersorozat, amit ki szeretnénk íratni.

Draw Pattern Matching Position VI



Draw Pattern Match Position függvény leírása

Paramétere:

- Image: Az a kép, amire a kiíratást akarjuk végezni.
- Matches: Az IMAQ Match Pattern 2 kimenete köthető rá. Az illesztés során egy minta többször is szerepelhet a vizsgálandó képen, ezért ennek a paramétere egy tömb, amely minden egyes találatról információt hordoz.
- Text Color: A kiírt szöveg színét állíthatjuk be.

6. A Program

A továbbiakban részletes útmutató olvasható a program megfelelő használatát illetően:

- Gépigény
- Futtató környezet
- Könyvtár szerkezet létrehozása

- Progamegységek és működésük

Gépigény (ajánlott minimum konfiguráció):

1. 3200+ CPU 64bit
2. 2 GB Ram
3. 512 MB VGA

Futtató Környezet:

1. NI. Vision.v8.6.Acquisition.Software Labview 8.6 vagy frissebb verzió
2. NI. Vision. Builder. AI.v3.6
3. NI. Vision.v8.6.Development.Module Labview Vision Developement Modul
4. Labview Drivers
5. Labview 8.6 vagy frissebb verzió
6. Windows XP SP3

Könyvtárszerkezet létrehozása:

Készítsünk egy fő könyvtárat, amiben a szükséges könyvtárakat és a programokat elhelyezhetjük. Legyen a neve: PROGRAM. E mappán belül hozzuk létre a következő alkönyvtárakat:

- MINTA

Ebben a mappában azon tárgyakról készült képeket helyezhetjük el, melyeket szeretnénk, hogy a robot tájékozódásra felhasználjon. Minden egyes tárgyról készíteni kell legalább egy, de inkább több képet, a minél tökéletesebb illesztési beállítások használata miatt. A mintaképek elkészíthetőek pl.: A Vision Assistant szoftverrel, vagy az általam készített egyszerű Mintakészítő programmal, mely segítségével

egyszerűen kivághatunk objektumokat kameránk élő képéről, és elmenthetjük azokat az általunk választott könyvtárba, egyedi névvel ellátva.

- KALIBRATED

Mivel a távolságméréshez elengedhetetlen hogy meg tudjuk határozni egy tárgy mérete mekkora a valósághoz képest a képen, azért szükséges egy úgynevezett kalibrált kép elkészítése, melyen rögzítve vannak a valós világhoz viszonyított távolságok. Ezt az eljárást sajnos minden esetben el kell végezni, amikor egy másik kamera által keresünk egy objektumot, mivel a kamerák felbontása és egyéb tulajdonságaik eltérőek. Tehát egy adott kamerával elkészített objektummintát csak szerencsés esetben fog a program megtalálni egy másik kamera valós idejű felvételén.

Egy kép kalibrálása

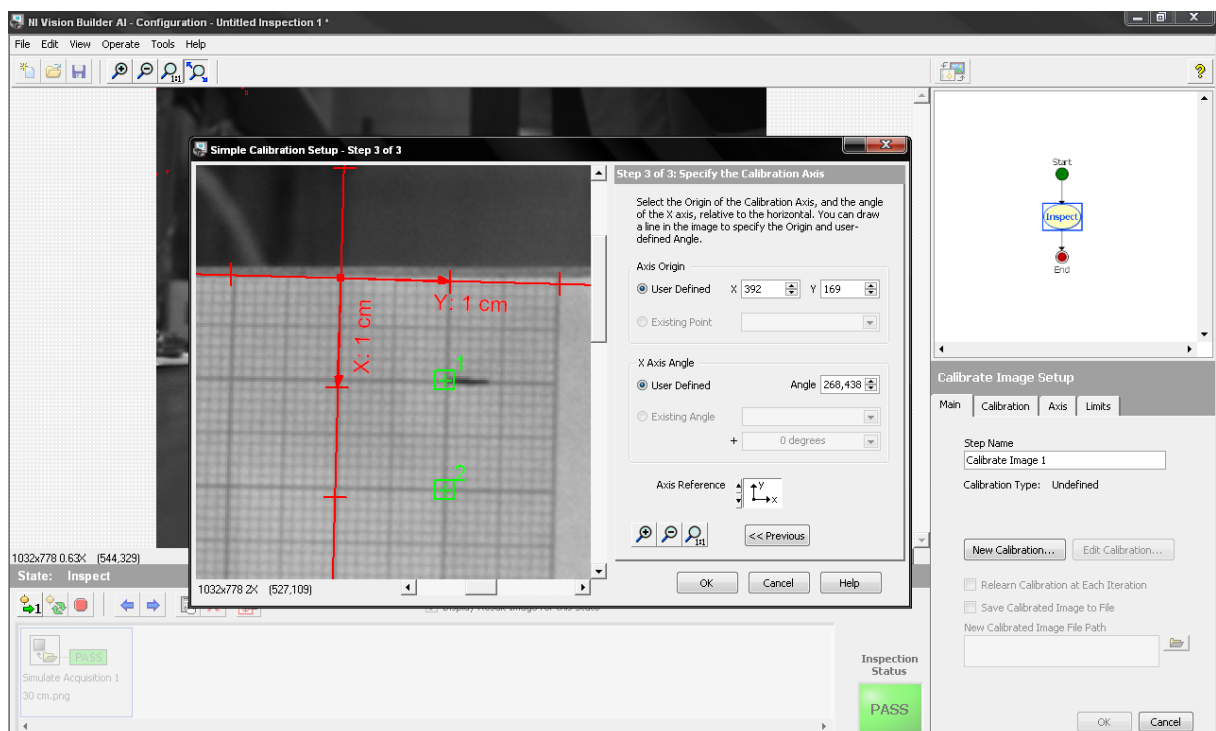
Egy kép kalibrálása egyszerűen kifejezve azért szükséges, hogy meg tudjuk mondani egy pixel hány centiméter. A dolog természetesen ettől azért bonyolultabb, hisz az hogy hány pixel alkot a képen egy centimétert az több dologtól is függ, például a kamera objektívének típusától illetve a fókusz távolságtól. Ezért fontos tehát, hogy minden esetben, amikor kamerát cserélünk, el kell készíteni az adott kamerához igazodó kalibrált képet. Egy kép kalibrálását a legegyszerűbben a Vision Builder AI segítségével tehetjük meg.

A szükséges lépések a következők:

1. Töltsük be a kalibrálni kívánt képet.
2. Menjünk a Calibrated Image ikonra.
3. Válasszuk a New Calibration, majd a felugró ablakon a Simple Calibration lehetőséget.

A 11. ábrán jól látható, hogy tetszőleges pontokat jelölhetünk ki a képen, valamint megadhatjuk az általunk kiválasztott pontok közül kettőnek a távolság a valós világban. A mértékegységek skálája elég széles, csak néhányat sorolnék fel: milliméter, centiméter, méter, inch.

Miután végeztünk a távolságok beállításával, a next gombra kattintva egy koordináta rendszert is definiálhatunk a képhez, mely beosztása az általunk előbb megadott távolság lesz. Célszerű egy olyan képet választanunk a kalibrált minta elkészítéséhez, amely a képen pontosan akkora, mint a valóságban.



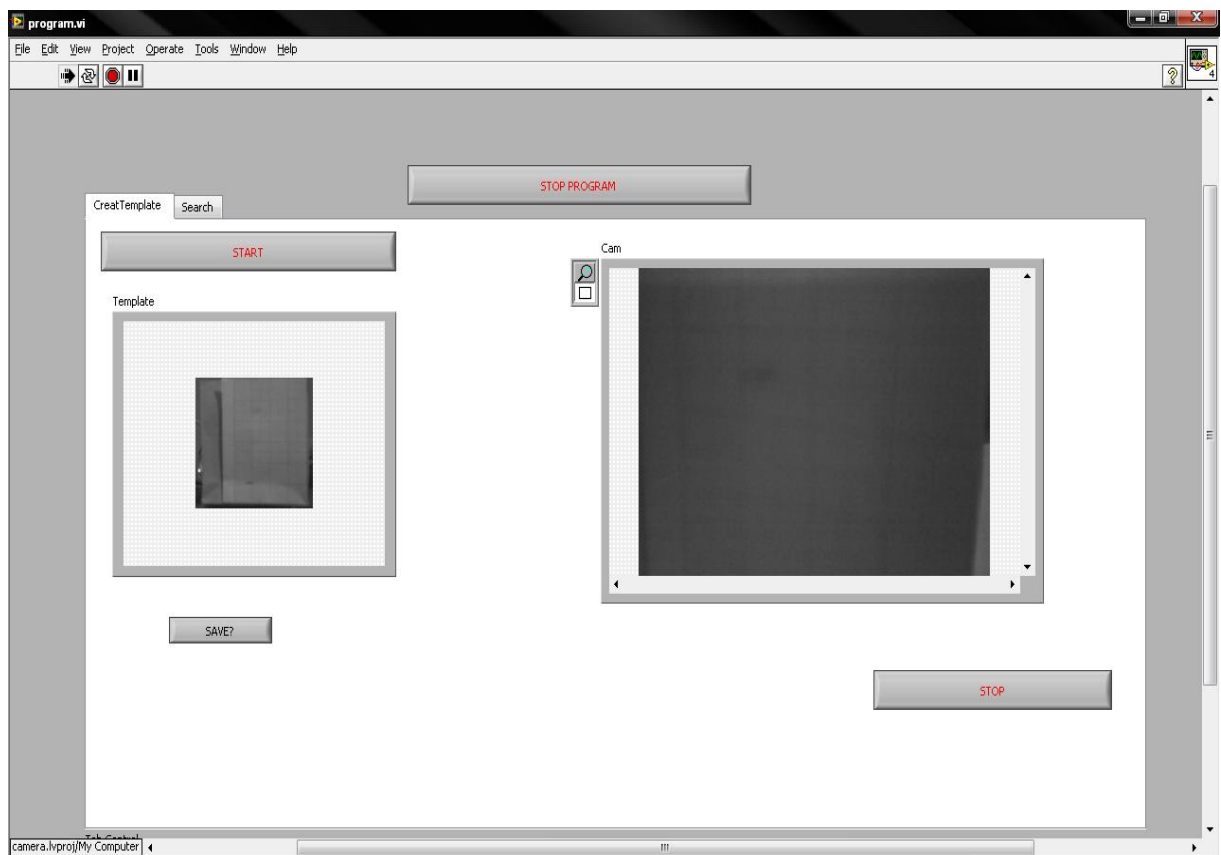
11. ábra: Egy kép kalibrálása Vision Builderben

A kép kalibrálására tehát azért van szükség, hogy a kalibrált képhez viszonyítva meg tudjuk határozni egy másik képen lévő dolgok méretét. Fontosnak tartom újból megjegyezni, hogy valahányszor kamerát váltunk, mindig szükséges egy a kamera adatainak megfelelően paraméterezett kép elkészítése, különben hibás méréseredményeket fog számunkra szolgáltatni a program.

Programegységek és működésük:

Mintakészítő Vi

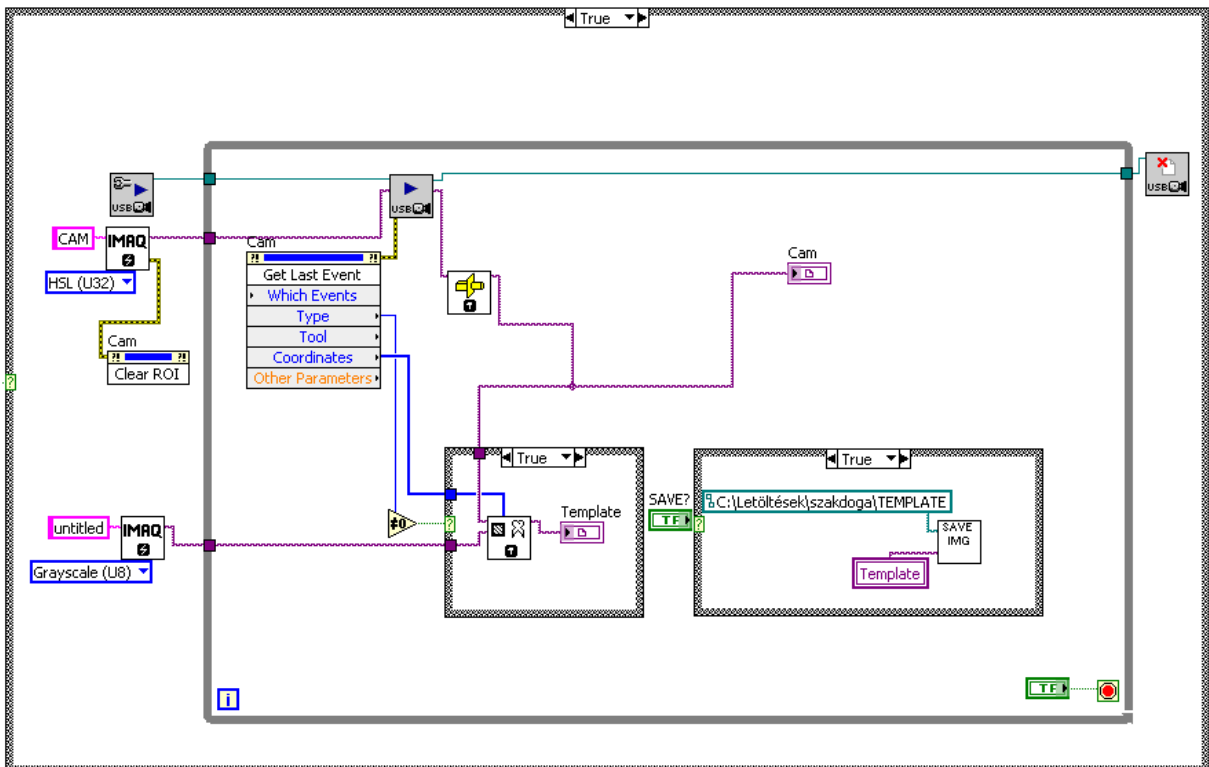
Ez saját készítésű mintakészítő program, melynek segítségével egyszerűen készíthetünk Pattern Matching mintákat. E program segítségével készítettem el a távolság meghatározó program tesztelésére használt mintákat.



12. ábra: A Mintakészítő Vi előlapi panele

A START gomb lenyomásának hatására a program elindítja a számítógéphez csatlakoztatott kamerát, és annak valós idejű képét megjeleníti a kijelzőn, majd a ROI függvény segítségével téglalap alakú területeket vághatunk ki a kamera által mutatott kép tetszőleges területéről. Minden egyes kivágott képet elmenthetünk a SAVE gomb lenyomásával, ezek lesznek a főprogramban használt mintáink.

A következő képen látható a kamera képeről való képrészlet kivágásának megvalósítása.

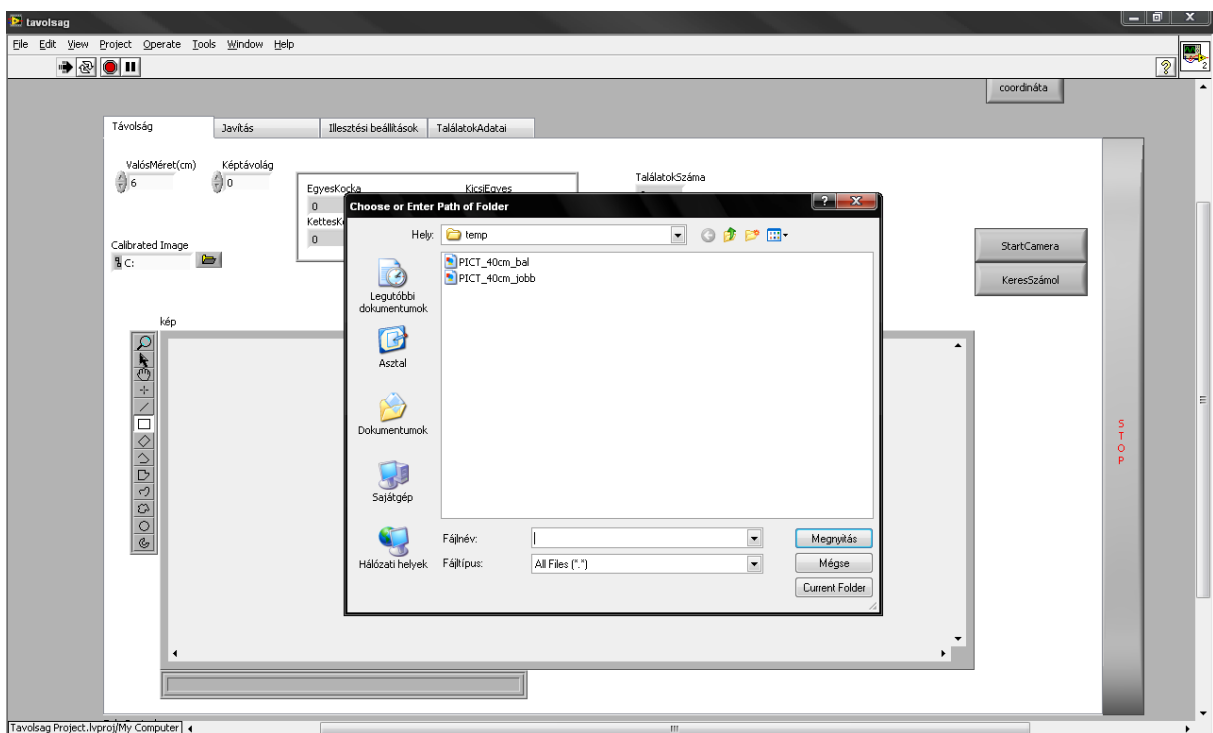


13. ábra: Kép kivágása

Tavolsag Vi

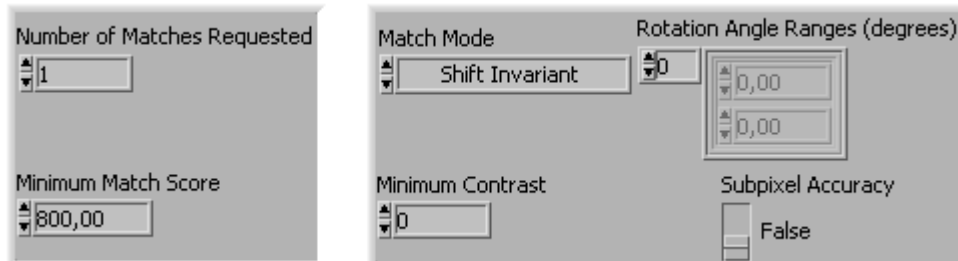
Ez a főprogramunk. A továbbiakban ennek a működése, kicsit részletesebb bemutatása következik.

E program előlapi paneljén adhatjuk meg a kereséshez szükséges beállításokat, beolvashatjuk a keresni kívánt objektumok mintaképeit egy mappából, melyeket a könnyebb kezelhetőség miatt egy tömbben tárolunk. A program indításakor felugrik egy ablak, ahol kiválaszthatjuk, melyik mappa tartalmát kívánjuk beolvasni.



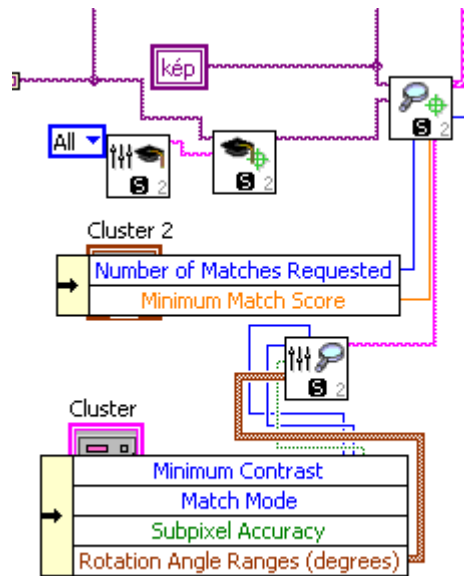
14. ábra: A mintákat tartalmazó könyvtár beolvasása

A következő dolgunk az Illesztési Beállítások fülön elvégezni a szükséges beállításokat, melyek a következők lehetnek:



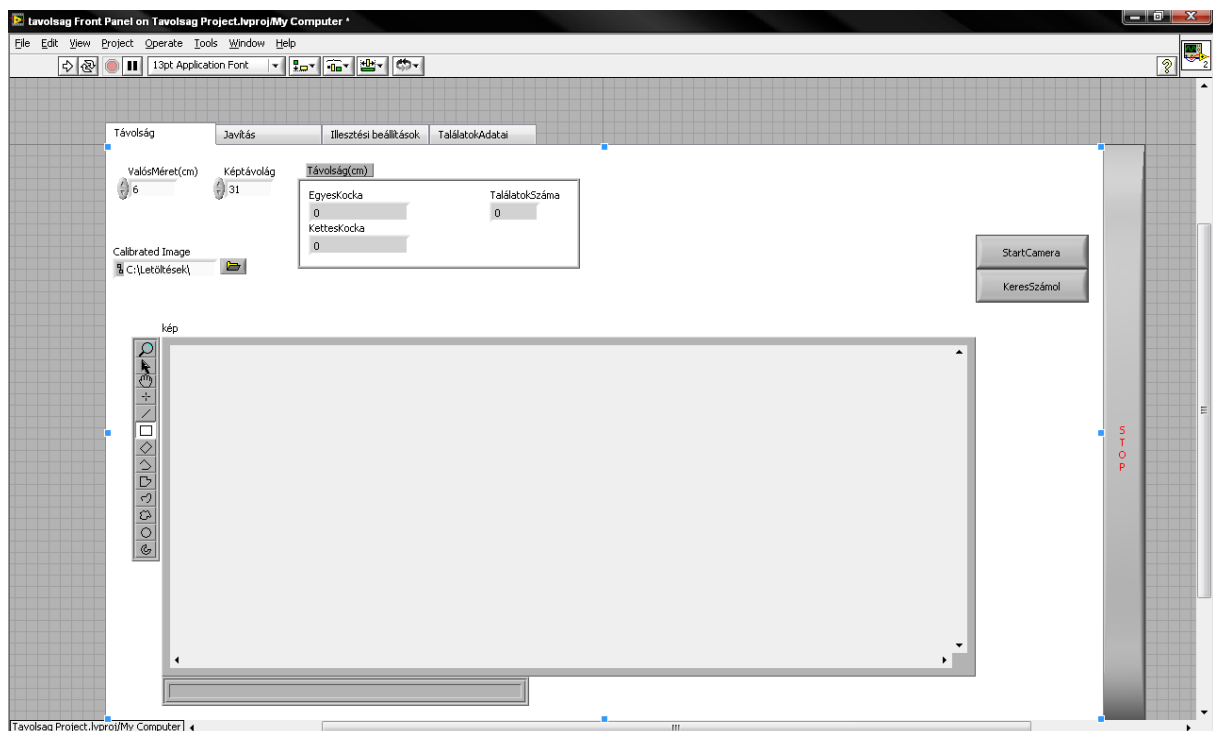
15. ábra: Illesztési beállítások

- Elvárt találatok száma (Number of Matches Requested): Az általunk egy mintára elvárt találatok száma. Tehát minden egyes vizsgált mintára legalább ennyi találatot fog figyelembe venni a program a keresés során.
- Minimális egyezési pontosság (Minimal Match Score): Az illesztés során minden talált alakzat, amire a minta illeszkedik, egy értéket kap, mely maximálisan 1000 lehet, ha az illesztés tökéletes. Ez akkor fordul elő, ha az általunk készített minta jó minőségű, és a kamera helyesen van fókuszálva. Tehát célszerű egy alacsonyabb értéket választani, mert előfordulhat, hogy nem minden esetben fogunk találatot kapni. Az alapértelmezett beállítás 800.
- Illesztés módja (Match Mode):
 - Eltolt (Shift Invariant)
 - Elforgatott (Rotation Invariant)
- Elforgatott illesztés esetén a forgatás mértéke szögben kifejezve (Rotation Angle Ranges)
- Alpixelek vizsgálata (Subpixel Accuracy).



16. ábra: Az illesztési kódrészlet

A Távolság fülön a következő beállításokat szükséges elvégeznünk:



17. ábra: A Távolság fül

Valós méret megadása: A kocka oldalméretének beállítása. Mivel én 6cm-es oldalhosszúságú kockákat használtam, alapértelmezésként ez van beállítva.

Képtávolság: Minden alkalommal, amikor egy másik, eddig még nem használt kamerát használunk, szükséges meghatározni a kamera képtávolságát a **Távolság meghatározás** című résznél leírtak alapján

Calibrated Image: Itt állíthatjuk be melyik kalibrált képet kívánjuk használni. Ez szintén attól függ, milyen kamerával dolgozunk, itt is fontos, hogy minden kamerához szükséges elkészíteni egy kalibrált képet

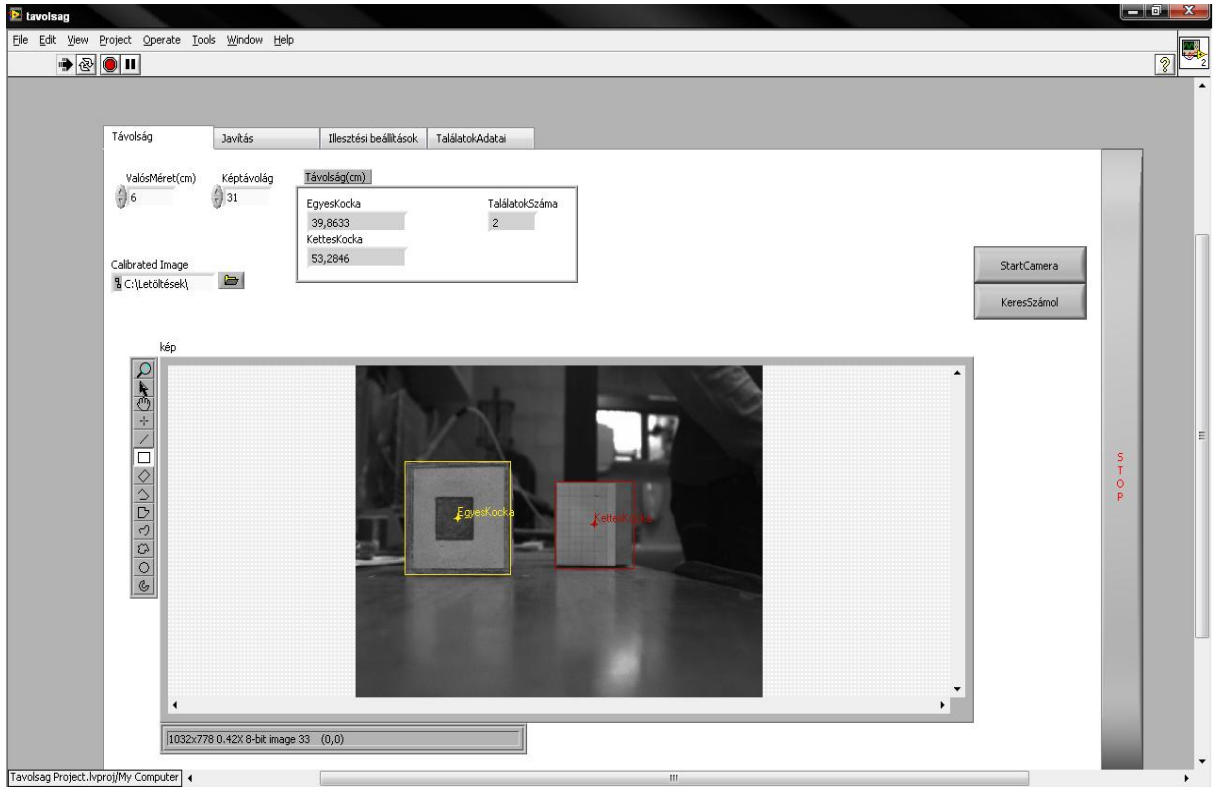
Miután elvégeztük a szükséges beállításokat a StartCamera gomb lenyomására a kamera elkezd képet közvetíteni a program számára, majd KeresSzámol gomb hatására a program a kamera képén elkezd keresni a mintakönyvtárból beolvasott tárgyakat, majd ha nem talált egyezést, akkor elkezd csökkenti a mintaképek méretét és minden egyes méretcsökkentés után elvégzi a mintaillesztést. Ez azért fontos, mert így valósítottam meg, hogy a program azokat a tárgyakat is felismerje, melyek távolabb vannak a kamerától, mint amikor a mintaképet készítettük.

Amint egyezést talál a program, a területet bekeretezi, és kiszámítja a tárgy távolságát, majd kiírja azt a megfelelő indikátorba. Mivel program tesztelése során két kockát használtam, ezért egyelőre ezt a két kockát tudja megkülönböztetni a program. Ennek megfelelően egy kockához egy változó tartozik a mért távolságok meghatározására. Ezek a következő ábrán láthatók:

Távolság(cm)	
EgyesKocka	TalálatokSzama
0	0
KettesKocka	
0	

A TalálatokSzama indikátorba értelemszerűen a megtalált objektumok száma kerül.

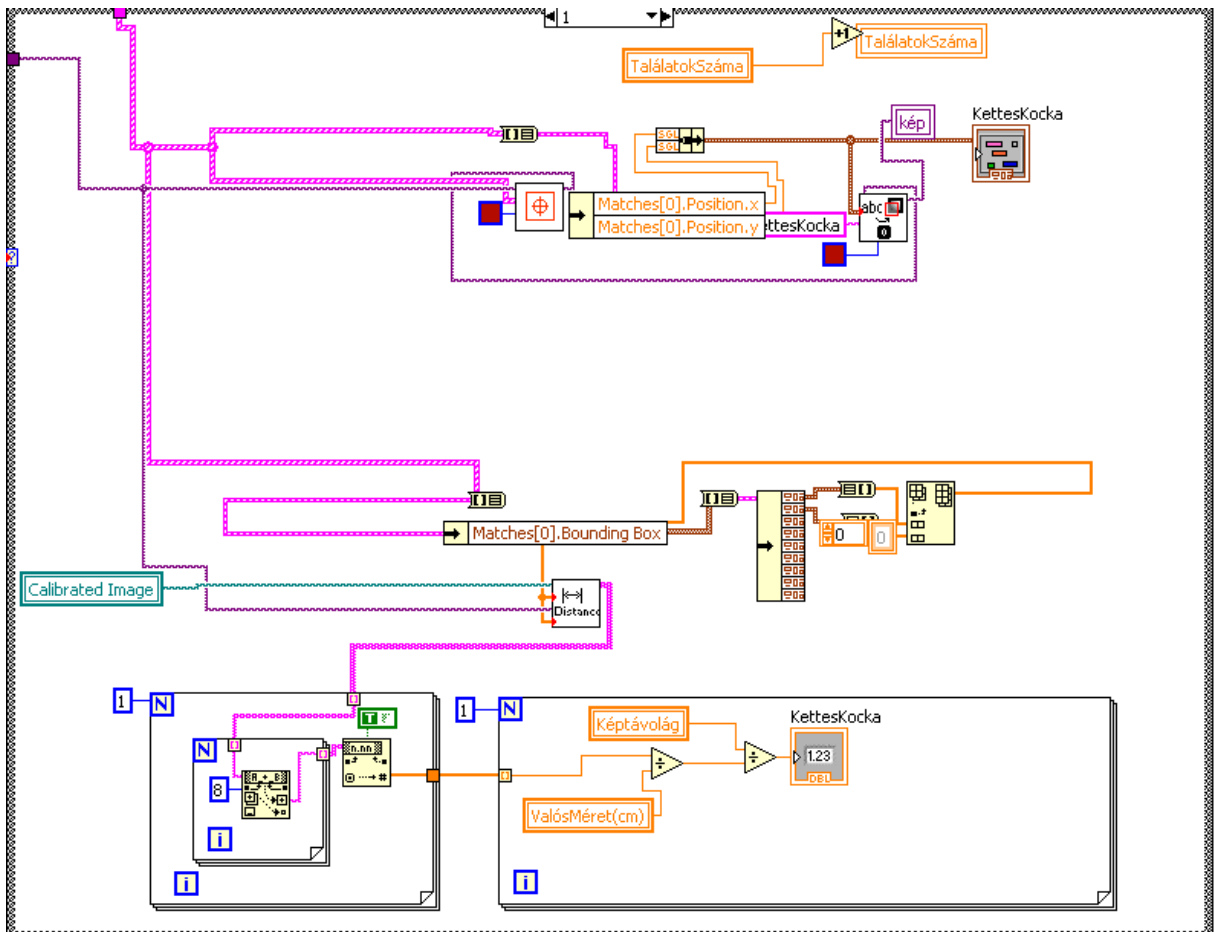
A következő képen a program működés közben látható. Jól látható, hogy mindkét általunk megadott mintát megtalálta, azokat bekeretezte, valamint kiírta távolságukat a megfelelő változóba.



18. ábra: Talált objektumok és távolságuk megjelenítése

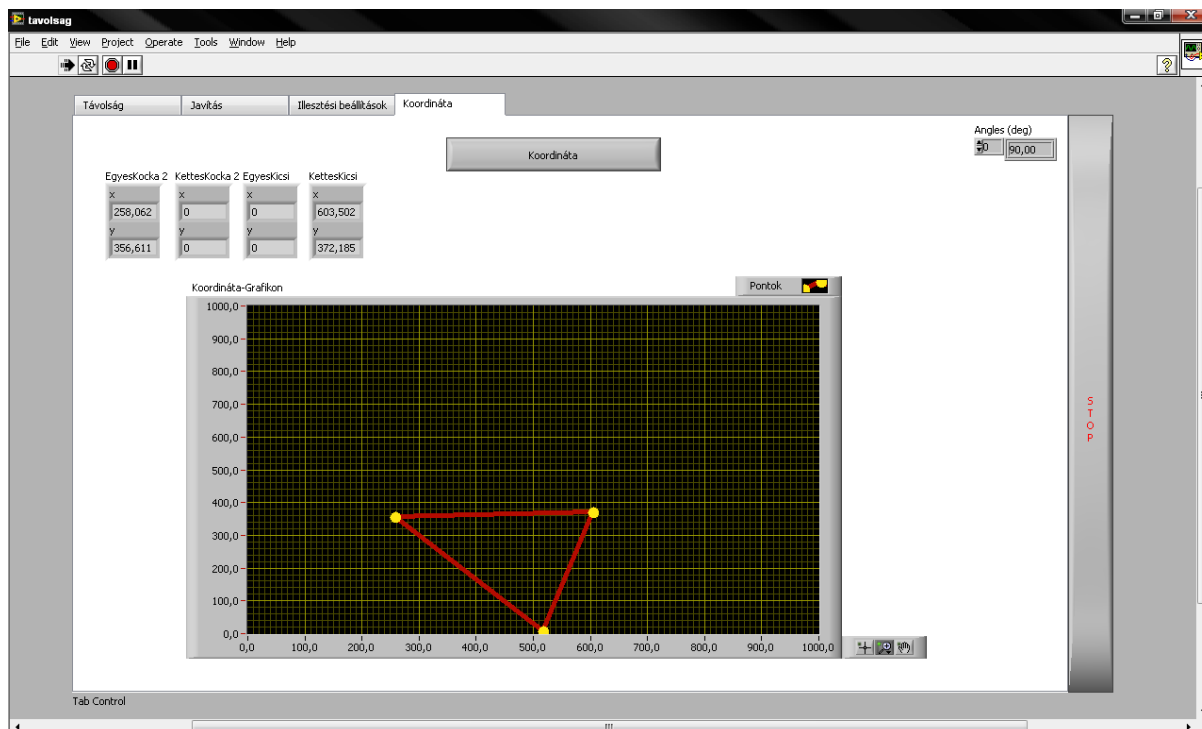
A következő képen a találatok kiértékelésének megvalósítása látható, vagyis itt dönti el a program melyik kockát találta meg. Továbbá itt valósul meg a távolság meghatározás is.

A Distance nevű SubVi végzi a képen talált kocka egyik oldalának a megmérését, majd a mért távolságot egy String tömbben kapjuk vissza, majd ezt megfelelően darabolva és numerikus típusú alaktípusra alakítva kiszámoljuk a távolságot.



19. ábra: Távolság meghatározása és kockák azonosítása

A következő fülön, ami a Koordináta, kirajzolhatjuk a kamera és a kockák elhelyezkedését egy koordináta-rendszerbe. Itt megjegyezném, hogy a kirajzolás még nem tökéletes, ugyanis a kamera vonala az X tengelyen van, ezért térben kicsit nehezen képzelhető el a tárgyak elhelyezkedése, de szemléltetésképpen szerettem volna ezt is beletenni a programba.



20. ábra: Koordinátarendszer

Miután végeztünk, a STOP gombbal leállíthatjuk a programot.

7. Összefoglalás

Szakdolgozatom célja volt, hogy betekintést nyerhessek a robotok világába, a különböző robotirányítási módszerek felépítésébe, azok működésébe.

Véleményem szerint megállapíthatjuk, hogy sikerült egy olyan szoftvert létrehozni, amely a képfeldolgozáson alapulva képes tárgyak azonosítására, elhelyezkedésük valamint a kamerától mért távolságuk kiszámítására, és bár a program még koránt sem tökéletes, azért összességében kijelenthetjük, hogy munkám nem volt hiábavaló és a kitűzött célokat megvalósítottam.

A program tesztelése úgy történt, hogy a kamerától különböző távolságokra valamint más-más irányba helyeztem el a kockákat, és a program által mért távolságokat a valóságban is lemértem. Így sikerült megállapítani, hogy a program hibahatára körülbelül $\pm 5\%$.

Dolgozatom első felében igyekeztem minél rövidebben összefoglalni a különböző robottípusokat, azok felépítését, valamint ismertettem néhány módszert melyek segítségével megvalósítható a robotok navigálása.

A második részben fontosnak tartottam egy kis bevezetést tenni a képfeldolgozás, és a képfeldolgozást megvalósító egyik legfontosabb fejlesztői környezet, a Labview világába, mivel a feladatom nélkülözhetetlen része, mondhatni lelke a képfeldolgozás volt.

Végül pedig jöjjenek személyes véleményeim, tapasztalataim: A robotok világa igen sokoldalú és rendkívül gazdag tudományterület, éppen ezért nagyon sok módon lehet egy-egy problémára választ találni. Elég csak arra gondolnunk, hogy egy robot helyzetének meghatározása is történhet a feladatomban választott képfelismeréses módszer helyett például GPS segítségével vagy akár odometriai méréseken alapulva. Munkám során azt tapasztaltam, hogy ugyan nem feltétlenül a képfelismeréses módszer a leghatékonyabb, mégis számomra ez tartalmazta a legtöbb kihívást, és ez vonzott a leginkább. Feladatom megvalósítása során számos új képfeldolgozási módszerrel valamint függvénnyel és szoftverrel sikerült megismerkednem, ami azt hiszem mindenképpen segítségemre lesz a későbbiekben.

8. Irodalomjegyzék

[1] dr. Gregovics Tibor ELTE IK: Tervgenerálás

people.inf.elte.hu/gt/mi/tervgenbev.pdf

Utolsó elérés: 2009.10.24

[2] IT3 Baráti Társaság weboldala

http://www.nhit-it3.hu/index.php?id=14316&option=com_content&task=view

Utolsó elérés: 2009.11.04

[3] Fazekas Annamária: Robotok, fizikai ágensek

www.cs.ubbcluj.ro/~csatol/mestint/diak_pdf/2005/fazekasannamaria.pdf

Utolsó elérés: 2009.11.07

[4] Sztrancsik Zsolt

users.atw.hu/sztrancsikzs/sztrancsik-zsolt-hf1.pdf

Utolsó elérés: 2009.11.01

[5] Nagyítás

<http://hu.wikipedia.org/wiki/Nagyítás>

Utolsó elérés: 2009.11.11

[6] Ental Kutatási és Fejlesztési Tanácsadó Kft.

ental.hu/data/attachments/file/2009031194132.pdf

Utolsó elérés: 2009.11.19

[7] Cserháti Csaba: Műszaki képfeldolgozás Labview-val elektronikus jegyzet

<http://roller.ttk.unideb.hu/moodle/mod/scorm/player.php?a=30¤torg=eXeDIP46107d3c1bb275ade62&scoid=231>

2009.11.13.

[8] Labview referencia

<http://zone.ni.com/reference/en-XX/help/371361A-01/>

2009.11.17

[9] IMAQ Vision for Labview Users Manual; NI kiadvány (2004)

9. Köszönetnyilvánítás

Szeretném megköszönni Dr. Szabó István Tanár Úrnak, hogy vállalta szakdolgozatom témavezetői szerepkörét, és építő jellegű tanácsival, észrevételeivel valamint szakmai tapasztalataival felügyelte és segítette munkámat.

Továbbá szeretném megköszönni Vámos Dánielnek Debreceni Egyetem Természettudományi és Technológiai Kar Szilárdtest Fizika Tanszék munkatársának, hogy gyakorlati tanácsaival segítségemre volt.