



I8 - Környezeti tér feltérképezése sztereo kamerakép alapján

*Mérési útmutató az
Irányítástechnika és Képfeldolgozás
Laboratóriumhoz: 8. mérés*

*Dr. Vajda Ferenc
3D érzékelés és Mobilrobotika kutatócsoport
Irányítástechnika és Informatika Tanszék
2015. október 7.*

Jelen dokumentum a Budapesti Műszaki és Gazdaságtudományi egyetem Irányítástechnika és képfeldolgozás laboratórium 1. c. tárgyához kapcsolódó mérési útmutató. A mérés és az útmutató a laboratórium 8. méréséhez készült a tanszék 3D érzékelés és mobilrobotika kutatócsoportjában.

A mérések és a mérési útmutató kidolgozásában részt vettek a kutatócsoport tagjai:

Vajda Ferenc
Takács Tibor
Srp Ágoston Mihály

A dokumentum szabadon letölthető, és átadható a tartalom módosítása nélkül. A dokumentumból történő idézések esetén a forrás megjelölendő.

Verziók:

Dátum	Verzió	Módosította	Leírás
07/10/09	V0.1	Vajda Ferenc	Első változat
03/09/15	V0.2	Szemenyei Márton	Második Változat

Tartalomjegyzék

1. Objektumfelismerés, pozíciómeghatározás	4
1.1 Háromdimenziós pozíció meghatározása	4
2. 3D pozicionálás virtuális valóságban történő navigációhoz	8
2.1 Célkitűzések	8
2.2 A mérés során alkalmazott eszközök	8
2.3 Feladatok	8
3. OpenCV referencia	10
3.1 Kép létrehozása	10
3.2 Megjelenítése	10
3.3 Bináris kép előállítása	10
3.4 Színtérkonverzió, szétválasztás komponensekre	11
3.5 Műveletek objektumokkal	11
3.6 Műveletek képrészekkel	11
3.7 Template matching	11
4. Ellenőrző kérdések	12
5. Irodalomjegyzék	13

1. Objektumfelismerés, pozíciómeghatározás

A kamerarendszerek egyik legfontosabb felhasználási célja a környezetben található objektumok szétválasztása és a lényeges elemek kiemelése, más szóval szegmentálás. Tekintettel a fontosságára, illetve a feladat meglehetősen nem egzakt jellegére, számos jó és kevésbé jó algoritmust alkalmazhatunk. Legtöbbjük csak különleges körülmények között használható, vagy olyan mértékű futásidőigénye, hogy a legtöbb alkalmazásban szóba se kerülhet. Emiatt az alkalmazások nagy részénél valamilyen módon egyszerűsíteniünk kell a környezetet, és olyan módon kell átalakítanunk, hogy az algoritmus számára könnyedén feldolgozható legyen. Erre megoldás, ha a megkeresendő objektum színét megváltoztatjuk olyan módon, hogy a környezettől elüssön.

A szegmentált objektumok megtalálása után annak releváns tulajdonságait is meg kell határoznunk. Szinte minden esetben szükséges a pozíció és/vagy orientáció, de gyakran van szükség méretre stb. A lényeges tulajdonságok kiemelése általában egyszerűbb feladat, mint a szegmentálás, de sokszor ezek meghatározása is szinte lehetetlennek tűnik.

Az alábbiakban ismertetünk néhány fontosabb algoritmust, amely segítségével a mérésben szükséges képfeldolgozási feladatok elvégezhetők. Ez természetesen nem jelenti, hogy a hallgató nem használhat szofisztikáltabb megoldást.

1.1 Háromdimenziós pozíció meghatározása

Objektumok háromdimenziós elhelyezkedésének meghatározására számos módszert ismerünk. Az elterjedtebb megoldások többnyire arra épülnek, hogy ugyanazon objektumot több szögből is felvesszük – ismerve a felvételkészítés paramétereit (kamera elhelyezkedése, egyéb paramétereit). Talán a legfontosabb eljárás a sztereo kameraképpár alapján történő pozíciómeghatározás. A mérés során a hallgató is ezt alkalmazza.

A háromdimenziós objektumoknak a kamera érzékelőjére történő vetítése az 1. ábrán látható módon ún. perspektív transzformáció segítségével történik. A

transzformáció homogén koordináták segítségével az alábbi módon fogalmazható meg:

$$\begin{bmatrix} x_w \\ y_w \\ w \end{bmatrix} = \mathbf{T}_{\text{persp}} \cdot \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (1)$$

ahol $\mathbf{T}_{\text{persp}}$ a kamera és a viszonyítási koordináta-rendszer közötti forgatást és eltolást, illetve az egy ponton keresztüli síkra történő vetítés transzformációs együtthatóit tartalmazza. Ebből a képpontok az alábbi módon származtathatók.

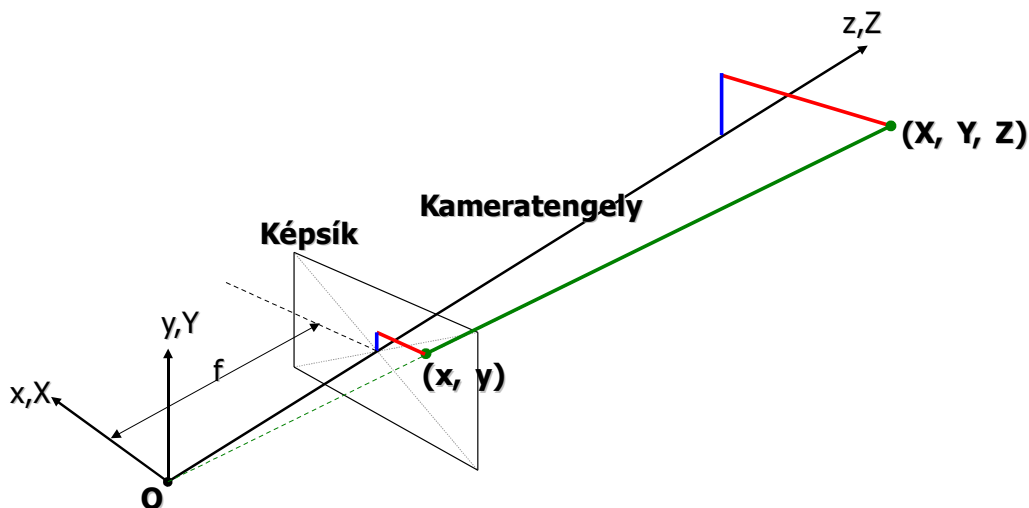
$$x = \frac{x_w}{w}, \text{ ill. } y = \frac{y_w}{w} \quad (2)$$

Amennyiben a viszonyítási koordináta-rendszer a kamera koordináta-rendszere (l. ábra), a transzformációs mátrix az alábbi módon egyszerűsödik:

$$\mathbf{T}_{\text{persp}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1/f & 0 \end{bmatrix} \quad (3)$$

Így

$$x = f \frac{X}{Z}, \text{ ill. } y = f \frac{Y}{Z} \quad (4)$$



1. ábra A képpalkotás

A képpontokat tipikusan nem a kép középpontjától számítjuk, hanem a bal

felső sarokból, ezért a koordinátákat ennek megfelelően el kell tolni.

Jól látható a fenti ábrán, hogy a 3D információ egyik eleme (z) a leképezés során elveszik. Ez érthető is, hiszen a 3D-ban meghatározott pont 3 koordinátája helyett a képen csak 2 értéket kapunk. A probléma kiküszöbölésének egyik módja, hogy két kameraképet használunk (sztereo geometria), és az így kapott többletinformációk alapján határozzuk meg a hiányzó koordinátát (l. 2. ábra).

Legegyszerűbb esetben az ábrán látható elrendezést alkalmazhatjuk, ahol a két kamera koordinátarendszerének tengelyei egymással párhuzamosak, vagyis

$$Z = Z_1 = Z_2, \text{ ill. } Y = Y_1 = Y_2 \quad (5)$$

Különbség csak az X koordináták meghatározása során keletkezik, amely megegyezik a koordinátarendszerek origóinak távolságával:

$$X_2 = X_1 - b \quad (6)$$

A leképzési egyenletekből (4) a következőt kaphatjuk:

$$x_1 = f \frac{X_1}{Z}, \text{ ill. } x_2 = f \frac{X_1 - b}{Z} \quad (7)$$

Ebből

$$x_1 - x_2 = \frac{f}{Z} (X_1 - (X_1 - b)) = \frac{fb}{Z} \quad (8)$$

Így:

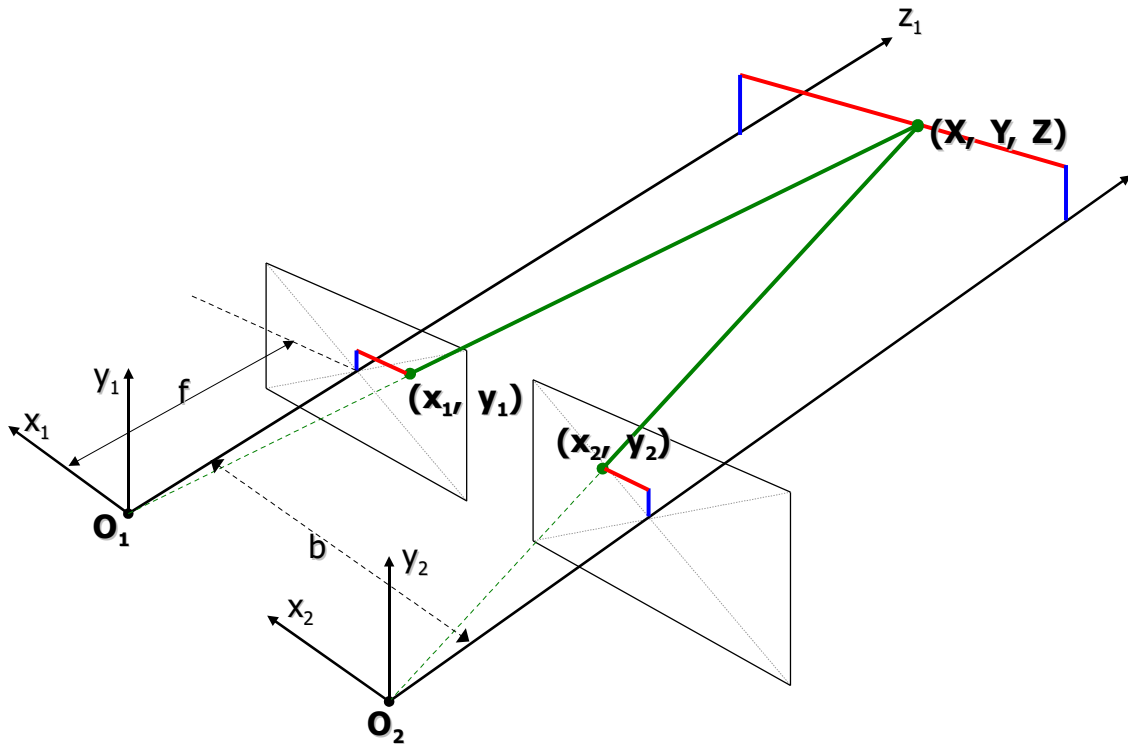
$$\boxed{Z = \frac{fb}{x_1 - x_2} = \frac{fb}{\Delta x}} \quad (9)$$

Valamint az egyes kamera koordinátarendszerében

$$\boxed{X = \frac{x_1 Z}{f} = \frac{x_1 b}{\Delta x}} \quad (10)$$

És

$$\boxed{Y = \frac{y Z}{f} = \frac{y b}{\Delta x}} \quad (11)$$



2. ábra A standard (ideális) sztereó elrendezés

A gyakorlatban nem fordul elő, hogy a két kamera geometriai viszonya a fentieknek megfelelő legyen, ráadásul a kamerák lencsési többnyire nemlineáris geometriai torzítást is okoznak amelynek egy viszonylag összetett matematikai művelet (ún. rektifikáció) segítségével korrigálhatunk. Az eljárás túllépi a mérés kereteit, így ezzel a továbbiakban nem foglalkozunk.

2. 3D pozicionálás virtuális valóságban történő navigációhoz

A mérés során a hallgató megismerkedik a sztereó képfeldolgozás alapjaival, illetve betekintést nyer a virtuális valóság rendszerekben való alkalmazásaiba.

2.1 Célkitűzések

A mérés során a hallgató egy virtuális térben működő navigációs alkalmazást valósít meg. A navigáció eszköze alapvetően – a hagyományos beviteli eszközök kiegészítéseként – egy (vagy több) térben mozgatható marker. A marker mozgásával különféle funkciókat valósíthatunk meg.

A hallgató feladata, hogy a marker háromdimenziós koordinátáit meghatározza egy sztereó-kameraképpár segítségével. A pozíciómeghatározó rutint egy meglévő virtuális valóság alkalmazáshoz kell illeszteni, amelynek eredményeképpen a marker segítségével a virtuális térbe lehet beavatkozni.

2.2 A mérés során alkalmazott eszközök

Eszköz	Típus	Paraméterek
Kamera	Bumblebee	Technológia: 2 × Sony progressive scan CCD Felbontás: 1024×768 Bázistávolság: 12cm Lencsék: 4mm fókusztávolság Kommunikáció: IEEE-1394 (Firewire)

2.3 Feladatok

- a) Írjon kalibrációs rutint OpenCV segítségével (a jegyzőkönyv végén talál egy rövid összefoglalót az OpenCV-ről), amely segítségével meghatározhatjuk azt a színt, amelyet a színalapú szegmentálás során használni fogunk. A feladat, hogy a képernyőn látható marker színét rögzítsük. Vegye figyelembe, hogy a marker színe a kamerán nem állandó.

- b) Írjon rutint, amely megtalálja a markert a két kameraképen. Az első kamera képen keresse meg azokat a pontokat, amelyek a színük alapján a markerhez tartozhatnak, majd ezek közül válassza ki azokat, amelyek a marker egyéb tulajdonságai (elsődlegesen a mérete) alapján valóban a markerhez tartoznak. A második kameraképen való kiválasztásnál feltételezheti, hogy a két kamerakép rektifikált, vagyis az első képen megtalált pontok y pozíciója megegyezik a pontnak második képre vetített y pozícióval (l. még 1.1).
- c) Határozza meg a marker háromdimenziós pozícióját az ismert paraméterek (fókusz távolság, bázis távolság, képközéppont koordináták) segítségével.
- d) Tesztelje az eljárást a mérésvezető által biztosított rendszerrel.
- e) Készítsen a mérésről jegyzőkönyvet. Ne felejtse el, hogy a jegyzőkönyv alapján a mérésnek rekonstruálhatónak kell lennie, ezért jelenjenek meg benne a felmerült problémák, azok megoldása, a végső rendszerben található hibák stb.

3. OpenCV referencia

Az alábbiakban néhány példán keresztül bemutatjuk az OpenCV használatát. Részletesebb információ az OpenCV dokumentációjában [1] található.

3.1 Kép létrehozása

```
// szürkeárnyalatos, egycsatornás kép ( a kép alapvetően cv::Mat )
cv::Mat image( cv::Size( width, height ), CV_8UC1 );

// színes, háromcsatornás kép
cv::Mat image( cv::Size( width, height ), CV_8UC3 );

// másik képpel megegyező méretű és típusú kép
cv::Mat image( other.size(), other.type() );

// betöltés fájlból
cv::Mat image = cv::imread( "filename.ext" );

// kamerából
cv::VideoCapture capture( 0 );
cv::Mat frame;
while ( ... ) {
    capture >> frame;
    if ( !frame.data ) { /* no image */ }
}
```

3.2 Megjelenítése

```
// kép megjelenítése
cv::imshow( "Ablak neve és egyben fejléce", image );

// várás billentyűzetre
int key = cv::waitKey( 0 ); // várás ideje ms-ban. Ha 0, akkor végtelen
// waitKey LEGALÁBB EGYSZER KELL!!!
```

3.3 Bináris kép előállítása

```
// szürkeárnyalatos kép előállítása színes képből
cv::Mat gray, binary;
cv::cvtColor( image, gray, CV_BGR2GRAY );
// BGR és nem RGB formátum az alapértelmezett!!!

cv::threshold( gray, bin, threshold_value, 255, cv::THRESH_BINARY );
// 255 a küszörbértéket meghaladó pixelek új értéke
```

3.4 Színtérkonverzió, szétválasztás komponensekre

```
// váltás szinterek között
cv::Mat hsvImage;
cv::cvtColor( rgbImage, hsvImage, CV_BGR2HSV ); // más konverziók is lehetnek
// szinterek: RGB, BGR, RGBA, BGR555, BGR565, GRAY, YCrCb, HSV, HLS, Luv, Lab
//           XYZ, BayerBG, BayerGB, BayerRG, BayerGR

// komponensek szétválasztása
std::vector< cv::Mat > hsv( 3 );
cv::split( hsvImage, hsv );
```

3.5 Műveletek objektumokkal

```
//objektumok keresése
cv::FindContours( binImg, contours, CV_RETR_EXTERNAL, CV_CHAIN_APPROX_SIMPLE );

// momentum kiszámolása
for( int i( 0 ); i < contours.size(); i++ ) {
    cv::Moments moments = cv::moments( cv::Mat( contours[ i ] );
    float zerothOrderMoment = moments.m00;
}
```

3.6 Műveletek képrészletekkel

```
cv::Mat contourImage;

// kontúrok segítségével maszk kép készítése
for( int i( 0 ); i < contours.size(); i++ ) {
    cv::drawContours( contourImage, contours, i, cv::Scalar( 255, 255, 255 ),
    thickness );
    // thickness a vonal vastagsága, ha -1, akkor kitölti a kontúr belsejét
}
```

3.7 Template matching

```
// A kimenet tárolására szolgáló kép
cv::Mat errors;

// template matching végrehajtása
cv::matchTemplate( image, template, errors, cv::TM_SQDIFF );
```

4. Ellenőrző kérdések

1. Foglalja össze néhány mondatban, a mérés célját és feladatait.
2. Adja meg a kamera képalkotásának egyenletét.
3. Ábrázolja a képalkotás folyamatát.
4. Milyen módon állítható vissza a képalkotáskor elvesző 3D információ?
5. Rajzolja fel az ideális sztereó elrendezést.

5. Irodalomjegyzék

- [1] OpenCV - Open Source Computer Vision Library,
<http://docs.opencv.org/index.html>