XV. FIATAL MŰSZAKIAK TUDOMÁNYOS ÜLÉSSZAKA

Kolozsvár, 2010. március 25-26.

AZ OPENCV LEHETŐSÉGEI A MÉRNÖKI MUNKÁBAN

TÓTH Kálmán

Abstract

OpenCV means Intel® Open Source Computer Vision Library. It is a collection of C functions and a few C++ classes that implement some popular Image Processing and Computer Vision algorithms. It is free for commercial and non-commercial uses. Applying OpenCV one can take advantage of high speed implementations of functions commonly used in Computer Vision and Image Processing.

Key words:

OpenCV, image

Összefoglalás

Az Intel® cég megalkotta az OpenCV-t, ami egy nyílt forráskódú látvány könyvtár. Ez az összeállítás C függvényekből és néhány C++ osztályból áll, ami megvalósít egy bizonyos elterjedt képfeldolgozó és számítógépes látvány algoritmust. Az OpenCV szabadon felhasználható kereskedelmi és nem kereskedelmi célokra. Könnyen ki lehet használni a nagy sebességet igénylő számítógépes látvány és képfeldolgozási funkciókat is.

Kulcsszavak:

OpenCV, kép

1. Bevezetés

Manapság elég sok helyen szükséges, hogy a számítógépet összekössük a külvilággal, különféle kamerák segítségével. Egy kamera képének megjelenítése és annak feldolgozása elég nehézkes munka. Az OpenCV egyszerűvé teszi ezt a folyamatot, továbbá különféle képfeldolgozási, alakzat felismerési, mozgáskövetési feladatok mellett, az XML fájlkezelést is támogatja. A cikk az 1.0 verzióhoz íródott, viszont már megjelent a 2.0-s verzió is.

Főbb területek, amiket modellezhetünk a könyvtár segítségével:

- Ember, számítógép kölcsönhatások;
- Tárgyazonosítás;
- Szegmentálás és felismerés;
- Felületek, gesztusok felismerése;
- Mozgáskövetés;
- Mozgások szerkezetének, tulajdonságainak vizsgálata.

A fent említett területek közül az alakzatok, gesztusok, mozgások felismerésével Naotoshi Seo [3] nagyon sokat foglalkozott. Honlapján találhatunk leírásokat a témával kapcsolatban.

A cikk további szakaszaiban bemutatom, hogy milyen egyszerűen lehet az OpenCV-vel képet megjeleníteni, és kamera képét feldolgozni.

2. Függvénykönyvtárak

Az OpenCV 6 fő függvénykönyvtárat tartalmaz, amelyeknek mindnek megvan a maga fő feladatköre:

1. ábra Az OpenCV funkcionális részei

CvCam: Kamera képének, és képek feldolgozásával foglalkozó függvénykönyvtár. Ennek segítségével tudjuk beállítani a képek és kamera felbontását, a minőséget és sok más paramétert.

HighGUI: Öt nagyobb részre tudjuk felbontani ezt a függvénykönyvtárat.

- Ablakok kezeléséért felelős rész;
- Képmegjelenítő, mentő eljárások;
- Videó funkciók: frame-ek kinyerése, beállítása, módosítása, írása;
- Rendszerfunkciók: képkonvertálás.

CxCore: Alapvető struktúrák: különféle pontok, skalárok, matematika egységek. Kép (tömb) műveletek: másolás, régió beállítások, matematikai függvények. Memória műveletek: blokkok felépítésének beállítása, memóriával kapcsolatos műveletek. Rajz műveletek: szín beállítások, vonalak, matematikai alakzatok. Adattárolással, írással kapcsolatos függvények. Továbbá ez a függvénykönyvtár tartalmazza a hibakezeléssel kapcsolatos függvényeket is.

CvAux: Több kamera képének feldolgozása, Eigen objektumok kezelése, rejtett Markov modell kereső eljárások.

ML: Gépi tanulást elősegítő függvények, statisztikai eljárások, normál Bayes osztályozók, K legközelebbi szomszéd elvű klaszterezők, döntési és véletlen fák, Neurális hálózatok.

3. Kamera képének feldolgozása

A rövid áttekintés után nézzünk egy rövid példát is. A kamera- és képfeldolgozás nagyon jól kidolgozott az OpenCV-ben, mivel ez a fő tevékenységi köre. Maximum két kamerát enged egy időben használni, amelye(ke)t egy ablak segítségével tudunk kiválasztani.

Egy kamera képének beállítása és megjelenítése pár sorból áll. Elsőnek inicializálni kell a függvénykönyvtárakat, majd az **int nselected=cvcamSelectCamera(&out);** függvény segítségével meghatározni, hogy hány darab kamera van csatlakoztatva, ezt tárolja az nselected változóba, továbbá a kamera azonosítót (ID) az out-ban. Ezek után be kell állítani a képre vonatkozó tulajdonságokat a **cvcamSetProperty(...)**; segítségével. Ha mindez megvan, érvényesíteni kell a fentieket: **cvcamInit()**; továbbá elindítani a **cvcamStart()**; függvény segítségével.

Amennyiben két kamera képét szeretnénk megjeleníteni, akkor csak pár sorral kell kibővíteni a fenti kódunkat. Elsősorban a kamera kiválasztásnál tárolt nselected-ben lévő értéket figyeljük:

Ha az **nselected** értéke nagyobb, mint nulla, akkor két kamera van. Ezért be kell állítani az 1. kamerára vonatkozó paramétereket, majd ha ez megvan, akkor a 2. kamerára vonatkozó beállításokat is.

Ha a kamera által vetített kép módosítása is nagyon egyszerű, tegyük fel, hogy négyzetrácsosan akarjuk megjeleníteni a kamerában látottakat. Ekkor egy **callback(IplImage* image);** függvényre van szükségünk, amit **cvcamSetProperty(0,CVCAM_PROP_CALLBACK,callback);** segítségével kell beállítanunk, ahhoz a kamerához, amelyiken akarunk ilyen effektust.

```
void callback(IplImage* image)
{
    IplImage* image1 = image;
    int i,j;
    assert(image);
    for(i=0; i<image1->height; i+=10)
    {
        for(j=(image1->widthStep)*i; j<(image1->widthStep)*(i+1); j+=image1->nChannels)
        {
            image1->imageData[j] = (char)255;
            image1->imageData[j+1] = 0;
            image1->imageData[j+2] = 0;
        }
    }
}
```

2. ábra Callback függvény

Ezek alapján láthatjuk, hogy elég kevés sorból, egy vagy két kamera képét feldolgozó programot lehet írni, ami négyzetrácsosan jeleníti meg az általa „látott” képet.

4. Összefoglaló

Az eddigiekben láthattunk két példát, ami az OpenCV egyszerűségére és kidolgozottságára mutat rá. Pár sorból álló forráskód segítségével megnyithatunk képet, vagy akár kamerát kezelhetünk, de ha bonyolultabb kép vagy videó feldolgozást, esetleg mátrix vagy vektorműveleteket akarunk megvalósítani, az is hasonló egyszerűséggel oldható meg az OpenCV segítségével. Így az összetettebb munkák során nem kell ezeknek az alkalmazásoknak a megírásával bajlódni, mivel ingyenesen elérhetőek, továbbá platform függetlenek, így könnyen használhatóak a már megírt alkalmazások több számítógépen is.

Irodalom

[1] <http://ce.sharif.edu/courses/85-86/2/ce712/resources/root/OpenCV/OpenCV%20Tutorial.pdf>

[2] <http://nik.bmf.hu/vamosy/GepiLatas2007/OpenCV/OPenCVAlapok.ppt>

[3] Dr. Gary Rost Bradski, Adrian Kaehler - *Learning OpenCV Computer Vision with the OpenCV Library*

[4] Naotoshi Seo - <http://note.sonots.com/OpenCV.html>

Tóth Kálmán

Iskola: Kecskeméti Főiskola GAMF Kar,
Cím: 6000, Magyarország, Kecskemét, Izsáki út 10.
Telefon / Fax: +36-70-3426234
E-mail: radzsah@gmail.com