

# Galgóczi Áron

## NBII6E

Gépi látás - (GKNB\_INTM038)

2020.12.20.

# Tartalomjegyzék

|  |   |
|--|---|
| 1 Feladat.....                           | 3 |
| 2 Elméleti háttér.....                   | 3 |
| 3 Megvalósítás terve és kivitelezés..... | 3 |
| 4 Tesztelés.....                         | 4 |
| 5 Felhasználói leírás.....               | 4 |
| 6 Irodalomjegyzék.....                   | 5 |

# Dobókocka számláló alkalmazás

## 1 Feladat

Bármilyen fényviszony és háttér előtt képes legyen megszámolni maximum 4 db dobókocka összértékét, illetve a dobókockák darabszámát is jelezze ki.

## 2 Elméleti háttér

Az interpoláció során a képeket nagyítjuk ( $x > 1$ ) vagy kicsinyítjük ( $1 < x < 0$ ) ahol  $x$  az interpoláció mértéke. A folyamat során a kép nem ismert értékeit közelítjük az ismert értékek felhasználásával. A medián szűrő folyamata során a vizsgált kép egy egységnyi méretű részlete alapján képzett számsorozat mediánja lesz az új intenzitás érték. Só bors jellegű zajok eltüntetése, de jelen feladatban a felesleges éllek eltávolítását végzi. A morfológia a képek összehasonlítása egy csúsztatott strukturáló elemmel. Ennek hatására bináris képeken alakzatokat lehet kiemelni egy általunk definiált négyzetes kernel segítségével. Valamint szürke árnyaltos képen képes kihangsúlyozni a kontúrokat egy mediánnal szűrt képen. A kör detektálásra két külön megközelítés van. Az egyik az ismert méretű köröket keres, a másik ismeretlen méretűeket. A lényeg, hogy a bináris kép pontjain egy vagy változó sugarú köröket helyez el. Ahol ezeknek a köröknek az íve metszi egymást ott potenciális kör középpont található, melynek a sugara előre definiált vagy a metsző körök sugarának átlaga. Él detektálás során megjelöljük azokat a helyeket, ahol az intenzitás nagy mértékben változik így kapunk egy bináris képet.

## 3 Megvalósítás terve és kivitelezés

Terv szerint a kockák pontjai körök így azokon lehet kör keresést végrehajtani, valamint egy kellően homályos képen egy kocka szintén kör alakúnak látszik.

Megvalósítást 3 különböző fájlban, egy tartalmazza detektálást és a maradék kettő egy tesztelői és egy felhasználói alkalmazást, melyek az első meghívásért felel.

1. Vizsgalat.py itt van definiálva a vizsgal(dir,bool) függvény. A függvény meghívásakor egy elérési utat tartalmazó szöveget vár és egy logikai változót vár, mely felelős engedélyezni a kép megjelenítését. Az elején probléma adódik a képek felbontásával, mert egy 3000 pixel széles képen egy kocka pl. 80 pixel széles, de egy kisebb képen ez jelentősen csökken, így mindenképp szükség van átalakításra így a képeket  $2000 \div \text{széllesség}$  arányszámmal interpoláltam így egy nagyobb kép esetén is 50 pixel maximum lehet egy pötty. Ezután 8-bites szürke árnyaltos képpé alakítom majd, ami át esik egy medián szűrőn és morfológiai záráson, és ezután következik a körök detektálása, a cv2 HoughCircles függvényével, majd az eredmények képre rajzolása, ahol egyben számolja is a dobás értékeit. »korok« nevű változó tartalmazza a dobás értéket. Kocka detektálás ezt egy 500 pixel szélességű képen hajtom végre. Szintén egy 8-bites szürke árnyaltos képen, melyet követ egy morfológia zárás és egy medián szűrés. A kis méretű kép előnye, hogy jobban elmosódottabb lesz a kép és folyik egybe, mintha csak egy egyszerűbb medián szűrés alkalmaznánk nagy kernel méret mellett. Az így kapott képen a kockák körszerűbbek. »kocka« nevű változó tartalmazza a kockák számát. A következő rész az eredmény írja ki a képre a »cv2.putText()« függvény segítségével és jeleníti meg a »megjelenit« változó bool értékének megfelelően, itt van lehetőség a kilépést biztosító »q« betű lenyomására. A futás végén egy tömbben vissza adja a kép nevét, a pontok és kockák számát és visszaad egy nullát, ha nem engedélyezik a kép megjelenítést, különben a cv2.waitKey(0)- kor leütött billentyű kódját mely ha a »q« betű ASCII kódját tartalmazza a program megszakítja a futását.

2. kockadobas.py a felhasználónak készült program. A program első sorában lehetőség van egy mappa név megadására, hogy mely mappából olvassunk sok képet. Ezután importálja a vizsgal függvény a vizsgalat fájlból, a munpy-t, cv2-t és az os- t. A futás során a program kér egy útvonalat, melynek egy képre kell mutatnia és át adja a »megjelenit« -nek, ha üresen hagyjuk akkor végig iterál az első sorban megadott mappán és egyesével megjeleníti a benne lévő képeket. A »megjelenit« nevű függvény a kapot elérési útvonalat ellenőrzi a cv2.haevImageReader eljárás segítségével, hogy képről van-e szó, hiba esetén kiírja az útvonalat és egy üzenetet: „Ez nem kép!”. Különben az »eredmeny« változóban tárolja a »vizsgal« függvényről kapottakat és (-1) -gyel tér vissza kilépés esetén.

3. teszt.py: A teszteléshez készült program. A vizsgalat.py-ból a vizsgal függvényt, a numpy-t, a cv2-t és az os-t importálja. A program az elején megkérdezi a tesztelőt, hogy „Vannak-e új képek? I/H” az adatbázisban, I/i az igazat jelenti, bármely más a hamisat jelenti, ezt az »uj« nevű változóban tárolja el. Ezután az „Eredmeny\_mert.txt” fájl nyitással kezd annak függvényében hogy vannak-e »uj« képek mert akkor ezen felül megnyitja „Eredmenyek\_valos.txt” -t is, és felírja a következő oszlopokat, akár mindkét fájlban is: Kép neve, Dobás értéke és Kockák száma. Ezután az előre definiált mappa tartalmát olvassa és ellenőrzi a cv2.haevImageReader eljárás segítségével, hogy képről van-e szó, hiba esetén kiírja az útvonalat és egy üzenetet: „Ez nem kép!”. Különben az »eredmeny« változóban tárolja a »vizsgal(kep,megjelenjen)« (ha a »megjelenjen«-t 1-re átírjuk a kódban megjelennek a képek egyesével) függvényről kapottakat, majd ezeket írja fájlba, ha van »uj« kép akkor bekéri a képen látható pontok és kockák számát és ezeket is a megfelelő fájlba írja. A végén ellenőrzi a tesztelő kilépési szándékát. Az eredmény fájlok tartalmazzák a kép nevét, a pontok számát és a kockák számát is.

#### 4 Tesztelés

A tesztelés során az eredményeket a „Eredmeny\_mert.txt”, „Eredmenyek\_valos.txt” szöveges dokumentumokból táblázat kezelő program segítségével lehet kinyerni. Ehhez először a teszt.py futtatása szükséges a „Vannak-e új kepek?” kérdésre I vagy i válaszolva, bármely olyan grafikus felülettel rendelkező operációs rendszeren, melyen telepítve van a python, a numpy és cv2 modulok. Ha a kép adatbázison nem módosítunk (*FONTOS: a program nem tudja eldönteni, hogy tényleg módosítottunk-e vagy sem! Ez teljesen a tesztelő felelőssége*) lehetőségünk van a képek nem megjelenítésével gyorsítani a tesztelést, ehhez a »megjelenjen« változó a jelzett sorban 0-ra kell állítani, ezt csak az »uj« változó képes ezt felülírni.

A program a kocka dobás értékeket a 56% -os pontossággal ismeri fel és a képek 24% -ban 1 hibát ejt. A kockák felismerése 43,9% pontosságú és 19% -ban egy hibát ejt.

#### 5 Felhasználói leírás

A program futtatására a Thonny nevű programot ajánlom, de megteszi bármely Python értelmező melyet grafikus felületről indítunk.

Én a Thonnyval való használatot fogom bemutatni. A Thonny első indítása után a Tools menüpontban az Open system shell lehetőséget választva, adjuk ki a következő parancsot: pip install opencv-contrib-python. Néha a pip helyet pip3 kell írni. Bármilyen további előtt töltsük le <https://github.com/Kovis97/Gepi-latas> oldalról a kockadobas.py és vizsgalat.py nevű fájlokat. Mozgassuk a kívánt mappába. Ezután a Files menüpontban az open lehetőségen keresztül nyissuk meg a kockadobas.py programot. Az első sorban módosíthatjuk a mappa elérési útvonalat, ha több képet szeretnénk egymás után vizsgálni. A futtatáshoz nyomjuk F5-öt. A program megkérdezi, hogy egy képet szeretnénk-e feldolgozni, ebben az esetben írjuk be a kép

útvonalát (Pl.: C:\képek\kép.jpg), amennyiben rosszul adtuk meg, vagy nem képre mutat a program erre figyelmeztetni fog és megáll („Ez nem kép!”). Ha több képet szeretnénk vizsgálni akkor hagyjuk üresen, ha rosszul adtuk meg, vagy nem csak képek vannak benne a program ekkor is „Ez nem kép!” üzenettel reagál. Egy kép megjelenése után egy billentyű lenyomására leáll vagy a következő képre ugrik a program, ekkor „q” betűre letudjuk állítani. A dobott kockák száma és azok összértéke a képre kerül kiírásra.

## 6 Irodalomjegyzék

- [http://www.inf.u-szeged.hu/~tanacs/pyocv/krk\\_detektlsa.html](http://www.inf.u-szeged.hu/~tanacs/pyocv/krk_detektlsa.html)
- [https://docs.opencv.org/master/d4/da8/group\\_imgcodecs.html](https://docs.opencv.org/master/d4/da8/group_imgcodecs.html)
- <https://hup.hu/comment/654902#comment-654902>
- <https://stackoverflow.com/questions/2349991/how-to-import-other-python-files>
- <https://infopy.eet.bme.hu/fajlkezeles/>