

Kő, Papír, Olló

(Képfeldolgozás alap algoritmusai 2022/23/1 féléves projekt)

A projekt egy olyan alkalmazás lenne, ami a Kő-papír-olló nevű játékon alapszik. Fő feladata az, hogy két játékos által készített kép alapján, amik tartalmaznak egyenként egy-egy kéz jelet (kő, papír vagy olló), el tudja dönteni, hogy melyik játékos nyerte a meccset. A két képet, vagy mobiltelefonos kamerával, vagy valamilyen számítógépes eszközzel kérné be a program két játékostól.

Képfeldolgozás terve

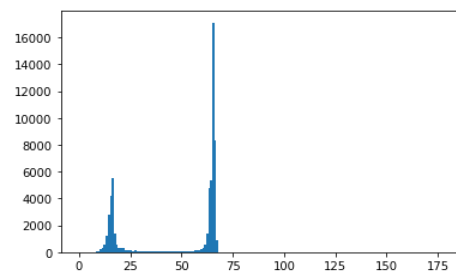
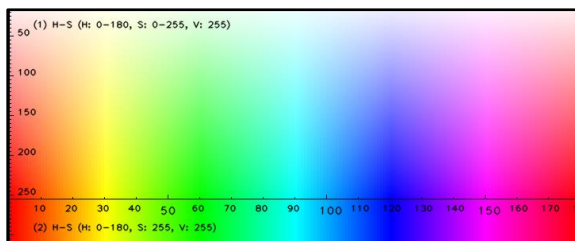
Képvétel

A képek két féle forrásból lennének beszerezve. Az egyik egy adatsorból tanító célból, amik azonos háttérrel és nagysággal rendelkeznek, ez lehetővé teszi, hogy könnyen elkülöníthető lesz a kéz alakja. A másik forrás, pedig a játék közben a játékos eszközén levő kamera képe. Ezek a képek, zajosak vagy nem egyszínű háttérrel rendelkeznek, ami miatt nehezebben lehet elkülöníteni a hátteret.



Tanító képek feldolgozása

A tanító képeken, a háttér jól elkülöníthető a kézfejektől a szín mélységek különbségének segítségével, a képek készítésekor jelen lévő stabil megvilágítás miatt. Ezért érdemes a képeknek Hue Value Saturation szín csatornából a színárnyalat intenzitás értéke alapján felállítani egy hisztogrammot, összeegyeztetni a lehetséges színárnyalat értékekkel és megkeresni a megfelelő küszöbértéket színek alapján.



Miután ki lett választva a küszöbérték T , az eredeti képek szürkeárnyaltos változatából, bináris képet lehetne készíteni, úgy hogy a küszöbölést egy adott (x, y) pixel $h(x, y)$ színárnyalat értéke alapján választjuk feketévé vagy fehérré. Ezen képek esetében, mivel a kéz színárnyalata kisebb, mint a háttéré az $f(x, y)$ függvény a következő:

$$f(x, y) = \begin{cases} 255 & , \text{ha } h(x, y) < T \\ 0 & , \text{különben} \end{cases}$$

De ha a kéz színárnyalata nagyobb, mint a háttéré, abban az esetben a feltételben a reláció megfordul. Ennek a küszöbölésnek a segítségével megtarthatjuk a képek azon

részét, amelyik kézfejet tartalmazza. Viszont zajossá válnak a képek, a kézfejek élei durvakká válnak, ezért további javításokat kell végezni.

Kép feljavítása

Először morfológiai szűrést, azon belül nyitást érdemes végrehajtani a képen, mert hasznos a kisebb objektumok és vékony vonalak képből való eltávolítására, miközben megőrzi a kép nagyobb objektumainak (jelen esetben kézfej) alakját és méretét. A nyitási művelet egy erózió (legyen a jele \ominus), aztán egy dilatació (legyen a jele \bullet), végrehajtása a képen úgy, hogy a két műveletek ugyanazt a kernel $n \times n$ mátrixot, K -t használják.

$$\text{Nyitás(Kép)} = (\text{Kép} \ominus K) \bullet K$$

Eróziónál, a kép pixelein végig csúsztatjuk a kernel mátrixot, ha a kernel, az adott pixelnél pontosan egyezik a képpel a pixel értéke ahol járunk 1 lesz, különben 0. Dilataciónál az adott kernelből, ha legalább 1 elem illeszkedik a képre, akkor az ott pixelnél, ahol járunk 1 lesz, különben 0

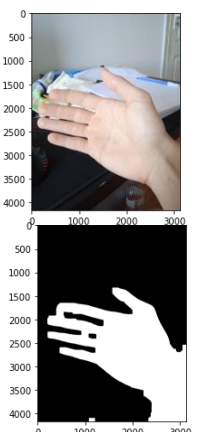
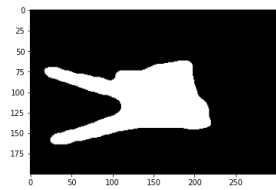
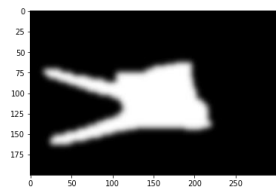
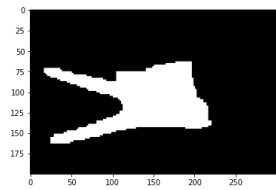
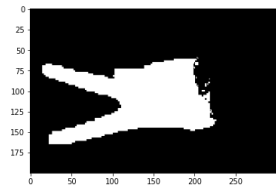
Ezek után valószínűleg a kép élei még mindig durvák lesznek ezért érdemes elhomályosítani a képet, hogy az élek kisimuljanak. A Gauss szűrő kerneljének elemei, a két dimenziós normális eloszlásból származnak, és mivel a kernel szélén az elemek 0-át közelítik, és a középponttól a széléig σ (szórás) állításával változnak az értékek, ezért megfelelő választás lenne, mert jól finom hangolható a kernel nagyságával és a szórás változtatásával.

Ezek után egy utolsó, küszöbölést végrehajthatunk, hogy megszabaduljunk a homályosítás során keletkezett felesleges szürke értékektől.

Az így keletkezett kép $n \times n$ mátrixából egy $n^2 \times 1$ vektort jellemző készül, ami a bemenete lesz az osztályozónak.

Játékból szerzett képek

A játék során az eszköz kamerájával fotózott kép esetében, mivel a háttér nem egységes, több lehet benne a főls objektum vagy zaj, ezért a nyitás lépést más léptékkal, nagyobb kernellel, vagy más típusú kernel struktúrával, lesz érdemes véghezvinni. Továbbá, ha úgy választjuk el a háttértől a kezét, mint tanításnál, akkor ügyelni kell rá, hogy a háttérben ne tartózkodjon olyan színű objektum, aminek a színárnyalata az emberi kezéhez hasonlít (parketta, lambéria). Miután sikerült elválasztani a háttértől a kezét, le kell kicsinyíteni, aztán fel kell majd javítani a fent tárgyalt módszerekkel, a képet a keletkezett torzulások és az egyenetlen élek miatt. Kicsinyítés előtt a kézfejet a kép középpontja felé kell tolni, hogy amikor tanításkor használt nagyságra



alakítjuk a képet, a kezekből ne vágódjon le fontos rész. Ha ez mind megvan, akkor beküldhetjük, a már kész osztályozó modelbe a képet egy jellemző vektor formájában.

Osztályozás

Support Vector Machines osztályozó algoritmusra esett a választás, mert egy olyan vektor lesz a bemenet az osztályozó algoritmusba, ami nagy dimenziószámmal rendelkezik. A kiválasztott algoritmus hatékonyan tudja kezelni az ilyen bemeneteket. Másrészt, az irodalomkutatás során, az ilyen típusú feladatokhoz, talált algoritmusok közül, az SVM klasszifikáló adta legtöbbször a legjobb megoldást. Az algoritmus, minden egyes mintát, elhelyezi egy annyi dimenziós térben, ahány dimenziós a bemenet, így a többdimenziós térbeli elhelyezkedés alapján, osztályozza a bejövő pontokat.

Megvalósítás

A tanításhoz a képeket tartalmazó adatsor, a kaggle.com oldaláról lenne beszerezve, ami 2188 képet tartalmaz. Mindegyik 300 pixel széles, 200 pixel magas, kézjelenként körülbelül 730 kép.

A szoftver a python nyelv segítségével készülne el, mert könnyen találhatóak hozzá, olyan rendelkezésre álló könyvtárak, amelyeket jól lehet használni képfeldolgozásra és klasszikus gépi tanulásra. Megvalósításhoz tervezett könyvtárak:

- OpenCV – képfeldolgozáshoz hatékony metódusokat tud szolgáltatni
- Matplotlib – képek, hisztogrammok megjelenítése
- Numpy – tömb adatstruktúrához, gyors numerikus számolásokhoz
- Scikit-Learn – gépi tanulás, osztályozó algoritmus, betanult model létrehozásához
- Kivy – Gui mobiltelefonos applikációhoz

A játék lebonyolítására három módot látok, az egyik lesz ezek közül megvalósítva a projekt elkészülésének a haladásától függően:

- Egy mobilon történik a játék, két képet kell egymásután készíteni, ami után az applikáció megmondja melyik kép nyert
- Egy webszerver bonyolítja le a játékot, két játékos két mobilról képet csinál, és a webszerver mondja meg ki nyert.

A feladat megoldási folyamata a következő lenne:

1. Először az osztályozó algoritmus tanítása, amit megelőz a képek feldolgozása, a képeken található kézfejek elkülönítése és aztán a jellemzők kinyerése.
2. Miután sikerült a tanítás, kipróbálni az osztályozó algoritmust éles képeken, ha esetleg nem teljesít jól, akkor a paraméterek változtatása, vagy visszatérés az első lépéshez

3. Ha az előző két lépés sikeres volt, akkor egy mobil vagy számítógépes alkalmazás létrehozása a kész model segítségével, a játéknak.

Felmerülő nehézségek

Legtöbb nehézséget abban látom, hogy implementáláskor a szegmentálás és a képjavítás tervezett pontjainál, a paramétereket úgy kell majd megválasztani, hogy a lehető legjobban kivehető legyen a kézfej alakja, hogy optimálisan működjön az osztályozás. A játék közben készült kép szegmentálásánál, javításánál és a nagyságuk normalizálásánál figyelembe kell venni, azt hogy az milyen hatással lesz az osztályozó algoritmus eredményére.

Források:

https://www.researchgate.net/publication/261168898_Hand_posture_recognition_using_K-NN_and_Support_Vector_Machine_classifiers_evaluated_on_our_proposed_HandReader_dataset

https://wiki.eigenvector.com/index.php?title=Image_Preprocessing_Methods

<https://www.kaggle.com/datasets/drgfreeman/rockpaperscissors>

<https://scikit-learn.org/stable/modules/svm.html>

<https://www.educba.com/kivy-vs-pyqt/>

<https://kivy.org/doc/stable/api-kivy.network.urlrequest.html>

<https://www.geeksforgeeks.org/connect-your-android-phone-camera-to-opencv-python/>