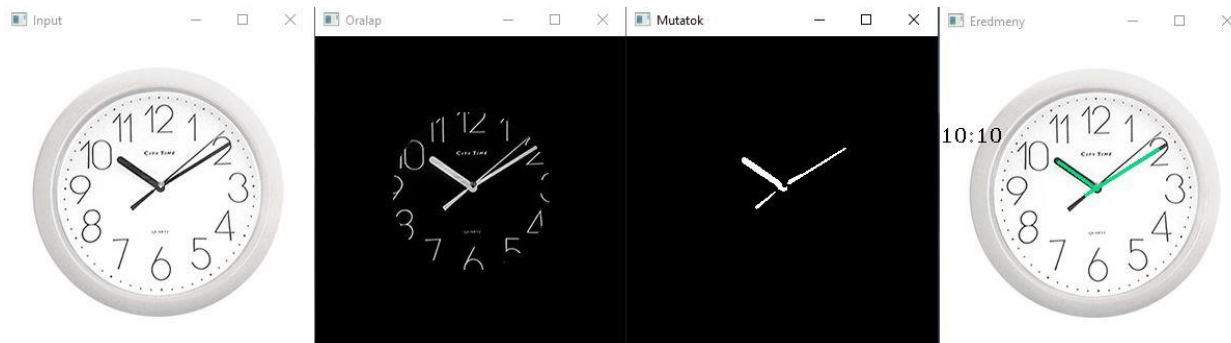


# Mennyi az idő?

(Digitális képfeldolgozás gyakorlat)

A program képes "jó" minőségű képeken megállapítani a hagyományos analóg órák mutatóinak állása alapján az időt, majd ezt az időt digitális órai formátumra konvertálni. A megközelítésem az óramutatók relatív bezárt szögeit veszi alapul, így olyan analóg órát is tudunk olvasni, amely nincs számozva.

Kitétel: Az óralapnak nem szabad túl árnyékosnak lennie, illetve arányaiban a leghosszabb és kicsit vastagabb mutatónak a percmutatónak kell lennie (és nem a másodpercmutatónak). A képnek rendelkeznie kell egyszínű háttérrel. Az órának nem szabad ferdén állnia.



A működés lépései:

1. A paraméterben kapott, a kitételnek megfelelő képet (.jpg vagy .png preferrált) beolvassuk a clockPic nevű változóba.
2. Ha megfelelő a kép az óráról, meghatározzuk a középpontot és a sugarat, majd a kapott információk alapján maszkolással és Hough transzformációval detektáljuk az óralapot és a mutatókat. Globális küszöböléssel-vágással (threshold) és éldetektálással (skimage skeletonize) meg hagyjuk az óramutatót és a percmutatót.
3. Megvizsgáljuk külön, hogy a két mutató milyen szögben áll kiindulva egy 360 fokos körből, majd összehasonlítjuk a két eredményt, ezzel kiadva a két szakasz közötti szögműködését.
4. A szögeket felhasználva algoritmussal (minden szám között az óralapon  $30^\circ$  van, pl 9 óra =  $30^\circ \cdot 9 = 270^\circ$ ) meghatározzuk az időt "digitális óra" formában (a perc oldalon az egyjegyűek elé nem kerül 0).
5. Kiírjuk az eredményt a konzolra és az eredeti képre berajzoljuk a mutatók állását az eredménnyel együtt.
6. Az output négy kép különböző fázisokból:  
Input -> inputkép,  
Oralap -> kép az óralapról,  
Mutatok -> Kép a szegmentált mutatókról,  
Eredmeny -> Eredménykép.

#### Megjegyzések:

- Rossz input esetén vannak változók, melyek undefined értéket vehetnek fel (pycharm warning).
- A futásidőt nagyban befolyásolja a beolvasott fájl mérete.
- Ha az óralapon a percet és órát jelző szakaszok hosszúak, megzavarhatják a program működését.
- A percmutatókat mellőző képeken jobban fut a program.
- Fotókon és digitális rajzokon is egyaránt működik az algoritmus, ha a kör alakú óralap jól szegmentálható.
- A programot lehetne tovább fejleszteni a másodperc mutató mérésének implementálásával, azonban a gyakorlatban problémás a másodpercmutató és a perc mutató megkülönböztetése csupán forma alapján.
- Szakirodalom: <http://www.inf.u-szeged.hu/~tanacs/pyocv>, <https://docs.opencv.org>, egyéb non-primer források