



EÖTVÖS LORÁND TUDOMÁNYEGYETEM

INFORMATIKAI KAR

INFORMÁCIÓS RENDSZEREK TANSZÉK

TANSZÉK

Időszakos belvizek detektálása műholdfelvételeken

Témavezető:

dr. Vincellér Zoltán

mesteroktató

Szerző:

Kilácskó Dávid

programtervező informatikus BSc

Budapest, 2022

Tartalomjegyzék

1. Bevezetés	3
2. Felhasználói dokumentáció	4
2.1. Szoftver használatához szükséges ismeretek	4
2.1.1. Műholdfelvételek	4
2.1.2. Sentinel-2 műhold	5
2.1.3. Vízmáskok létrehozása	6
2.1.4. GeoTIFF	8
2.2. Gépígyény és telepítési útmutató	8
2.2.1. Gépígyény	8
2.2.2. Telepítési útmutató	8
2.3. Használati útmutató	9
3. Fejlesztői dokumentáció	17
3.1. Fejlesztési környezet	17
3.2. Szoftverarchitektúra	17
3.3. Tervezés	18
3.3.1. Funkcionális követelmények	18
3.3.2. Kiválasztott módszerek	19
3.3.3. A szoftver felhasználói esetei	21
3.3.4. A szoftver felépítése	23
3.4. Tesztelés	25
3.4.1. Felhasználói felület tesztelése	25
4. Összegzés	30
Köszönetnyilvánítás	32
A. Szimulációs eredmények	33

A.1. Felszíni vizek változásdetektálása	33
Irodalomjegyzék	38
Ábrajegyzék	40
Táblázatjegyzék	42
Forráskódjegyzék	43

1. fejezet

Bevezetés

A távérzékelés egyik legsokrétűbb területe a műholdfelvételek multitemporális elemzése. Szakdolgozatomban ennek a témakörnek egyik felhasználásához készítettem alkalmazást. Olyan szoftver létrehozása a cél, ami képes azokat a területeket detektálni, amelyek belvizes időszakban telítődnek, de aszályos vagy átlagos időszakban kiszáradnak. Például ilyenek lehetnek kiszáradt folyóvölgyek, folyókanyarulatok, és árterületek. Ezeknek sok területen van jelentősége, a mezőgazdaságot, az ipart, a várostervezést és a közlekedést is meghatározzák. Valamint ezen területeken az árvizek veszélye is nagy.

A téma kiválasztásához a motivációt a Lechner Tudásközpont munkatársai fogalmazták meg. Szakmai gyakorlatom során, arra kértek fel minket, hogy keressünk módszert az időszakos belvizek detektálásának automatizálására. Ezt eddig szakértők vizuális interpretációja alapján állították elő. Célunk ehhez hasonló minőségű megoldás előállítása volt anélkül, hogy a folyamat során a felhasználói tudásra hagyatkozzunk. Úgy döntöttem, hogy szakdolgozatomat ebben a témában fogom elkészíteni. A probléma megoldására létrehoztam egy grafikus felületet, ami képes a detektálási feladatot elvégezni, valamint eredményeit megjeleníteni.

A megoldáshoz több módszer is vizsgálva lett, összehasonlításukról és eredményeikről egy TDK dolgozat is született. Ezek közül a legjobbak kerültek bele ebbe a programba. A végső a következő lépésekkel működik. Elsőként mivel a víz detektálásánál nagy problémát okoznak a felhők és felhőárnyékok, ezekre hoz létre egy maszkot az fmask algoritmus felhasználásával. Ezután a felvételek indexei, és küszöbölő algoritmusok segítségével vízmaszkot készít a felvételekre, majd ezeken végez változás vizsgálatot.

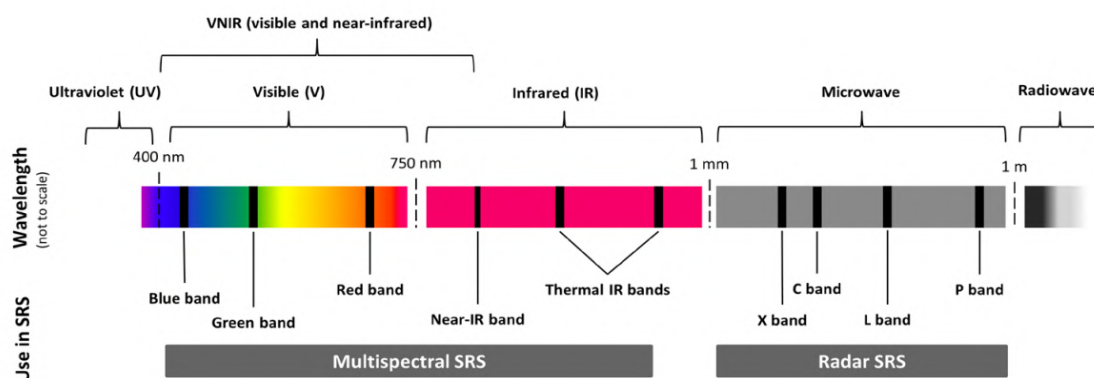
2. fejezet

Felhasználói dokumentáció

2.1. Szoftver használatához szükséges ismeretek

2.1.1. Műholdfelvételek

Multispektrális felvételnek azokat a műholdképeket nevezzük, amelyek az elektromágneses spektrum különböző tartományyaiból foglalnak magukba adatokat. Ezek hullámhosszszávonként csatornába vannak elkülönítve képeken, például: az RGB, a közeli- és termális infravörös sávok. A legtöbb földmegfigyelő műhold így rögzíti az adatait. Különbség abban van közöttük, hogy hány csatornát, és milyen hullámhosszok értékeit tartalmazzák. A program a Sentinel-2-es műholdak képeit tudja feldolgozni. Az elektromágneses sugárzás spektrumát a 2.1-es ábrán lehet megtekinteni.



2.1. ábra. Az elektromágneses sugárzás spektruma (nem méretarányos) és felhasználása a műholdas távérzékelésben.

Forrás: <https://www.mdpi.com/2075-4450/12/3/233/htm>

2.1.2. Sentinel-2 műhold

A Sentinel-2 az Európai Űrügynökség Kopernikusz programja¹ keretében működtetett földmegfigyelő műholdpár. Mindkét műhold nagyfelbontású, multispektrális kamerával van felszerelve. Adataikat 13 sávban rögzítik a spektrum látható, közeli- és közepes infravörös tartományaiból, ezeknek a térbeli felbontása 10 és 60 méter között van. A sávokról bővebben a 2.1 táblázatban vannak információk. A Sentinel-2 felvételei sok területen felhasználtak, többek között mezőgazdaságban a termésbecsléshez és káresemények felméréséhez alkalmazzák. Emellett katasztrófavédelmi helyzetekben is jó hasznát veszik az általuk gyűjtött információknak.

Sáv sorszáma	Sáv neve	Felbontás (m)
B01	Coastal aerosols	60
B02	Blue	10
B03	Green	10
B04	Red	10
B05	Red Edge	20
B06	Red Edge	20
B07	Red Edge	20
B08	Near Infrared	10
B8A	Narrow Near Infrared	20
B09	Water Vapour	60
B10	Shortwave Infrared - Cirrus	60
B11	Shortwave Infrared	20
B12	Shortwave Infrared	20

2.1. táblázat. Sentinel-2 sávjainak sorszáma, elnevezése és felbontása

A szoftveren belül a változásdetektálás elvégzéséhez szükséges legalább kettő műholdkép, ami ugyanarról a területről, de különböző időpontban készült. Ezen felül a képeknek Level 1-C feldolgozottságúnak kell lennie (azaz a pixelértékek a légkör tetején mért reflektanciában számítottak), hogy minden lépés működjön rajtuk. A Sentinel-2 felvételek díjmentesen elérhetőek a <https://scihub.copernicus.eu/dhus/#/home> weboldalon.

¹<https://scihub.copernicus.eu/>

Az itt található file-ok a következőképpen vannak elnevezve:

MMM_MSIXXX_YYYYMMDDHHMMSS_Nxxyy_ROOO_Txxxxx_YYYYMMDDHHMMSS

Ennek részei a következőket jelentik:

- MMM: Azt jelzi, hogy melyik műhold készítette a felvételt. (S2A/S2B)
- MSIXXX: A felvétel feldolgozottsági szintje, MSIL1C a Level-1C-t, az MSIL2A pedig a Level-2A-t jelenti
- YYYYMMDDHHMMSS: A kép készítésének kezdeti időpontja.
- Nxxyy: A PDSG feldolgozási alapszám.
- ROOO: A relatív keringési pálya száma.
- Txxxxx: A kép Sentinel-2 helyszíni csempéjének azonosítója.
- YYYYMMDDHHMMSS: A felvétel termékazonosítója, ez különbözteti meg egymástól az ugyanakkor készült képeket. A felvétel készítésének kezdeti időpontjától kicsivel eltérő időpont.

2.1.3. Vízaszkok létrehozása

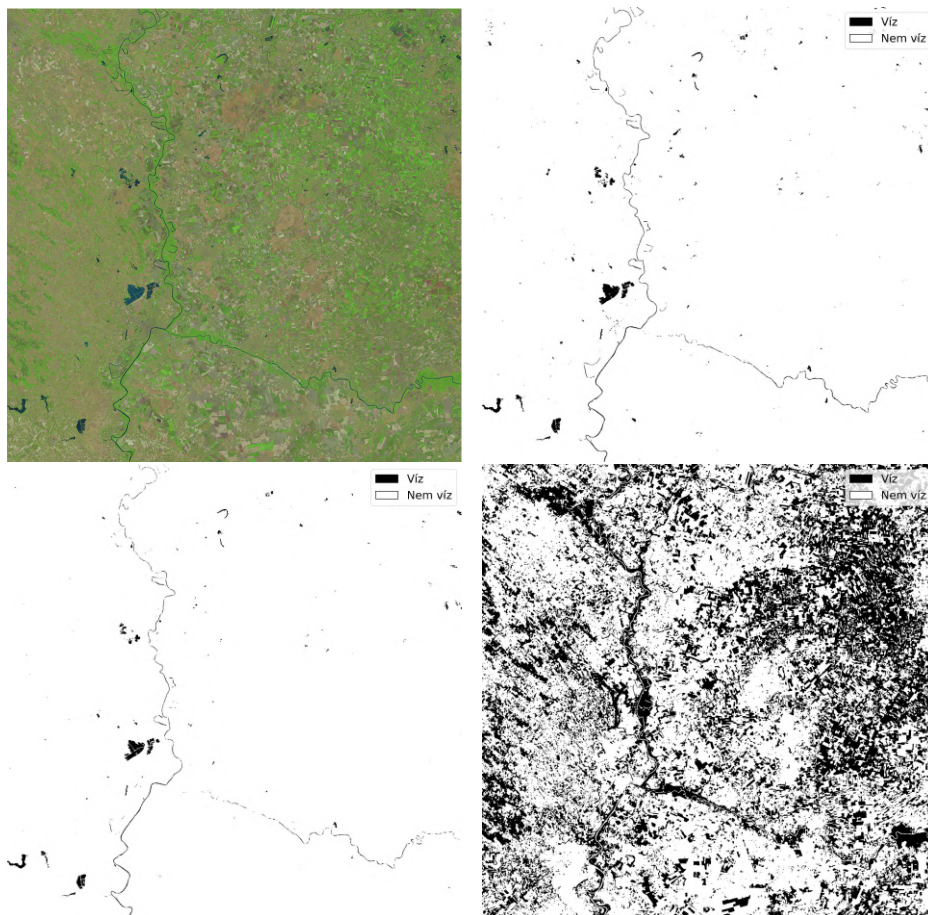
A folyamat egyik legfontosabb lépése a vízaszk elkészítése. Ez azt takarja, hogy a műholdfelvételhez elkészítünk egy új képet, amin csak azokat a területeket jelöljük, ahol valamilyen felszíni vizet detektáltunk. Később a változásvizsgálatot már csak ezeken a maszkokon fogjuk elvégezni. Így kiszűrve olyan differenciákat a képeken, amiket olyan területen találtunk, ahol egyik időpontban sem volt víz.

A vízaszk előállítása két fő lépésből áll. Első egy vízindex kiszámítása a képen. Ezt az index fajtájától függően a műholdfelvétel két sávjának normalizált különbségének meghatározásával kapjuk meg. A programon belül két ilyen index közül lehet választani mindkettő az NDWI [1] egy módosított verziója. Ezek az NDWI_g és az mNDWI[2]. Az előbbi a Sentinel-2 SWIR(B11) és zöld(B04), az utóbbi a vörös(B03) és SWIR(B11) sávjaiból állítható elő.

A második lépés az indexek küszöbölése, azaz egy olyan érték meghatározása, aminél kisebb értékeket szárazföldnek, nála nagyobbakat pedig víznek osztályozunk be. Ezt a számot úgynevezett *Thresholding*(küszöbölő) algoritmussal számoljuk ki a képekre. Ezeknek előnye, hogy automatikusan tudjuk meghatározni velük a határértéket, anélkül hogy szakértői segítségre lenne szükségünk. A szoftverben három ilyen módszer közül választhatunk: a Minimum, a Yen és az Otsu.

A Minimum algoritmus az NDWIg indexszel párosítva működik legjobban, az mNDWI-vel viszont van, hogy nem ad értelmes eredményt. A Yen pedig éppen az mNDWI-vel működik remekül, és az NDWIg-vel tér vissza rossz kimenetellel. Az Otsu mindegyikre működik, de mNDWI-nél hajlamos túlságosan alacsony értéket meghatározni ezzel rontva a végső pontosságot. Így elkészült vízmaszkok összehasonlítása a 2.2 képen látható.

Ezen kívül fontos még felhőmaszk alkalmazása a képekre, egyrészt mivel a felhők elrejtik az alattuk lévő területeket, másrészt a felhőárnyékok indexszeléssel nagyon nehezen különböztethetőek meg a vizektől. Így az ilyen területeket eltávolítjuk a maszkokból. Ehhez az fmask [3] algoritmus lett kiválasztva. Ez egy eredetileg Landsat felvételekre kidolgozott algoritmus, ami képes felhőket és árnyékaikat detektálni. Az alkalmazott implementáció az algoritmus egy továbbfejlesztett verziója [4], ami már működik a Sentinel-2-es képekre is.



2.2. ábra. Elkészült vízmaszkok összehasonlítása

Bal felül a színes műholdkép, majd a Yen, Minimum, Otsu algoritmusok által elkészített vízmaszkok.

2.1.4. GeoTIFF

A GeoTIFF egy térinformatikában gyakran használt fájlformátum, aminek előnye hogy a képi adatok mellett tud földrajzi információkat is tárolni. Ez lehetővé teszi a folyamatunk eredményeinek mentésekor hozzárendeljük a feldolgozott Sentinel-2 felvétel térbeli elhelyezését. Így más térképészeti szoftverben is kényelmesen kezelhetők lesznek.

2.2. Gépigény és telepítési útmutató

2.2.1. Gépigény

Név	Minimum ajánlott
Operációs rendszer	Windows 10 64bit
CPU	Intel I5
RAM	8 GB
Tárhely	5 GB

2.2. táblázat. A szoftver gépigénye

Maga a szoftver ugyan nem igényel jelentős tárhelyet, mind a feldolgozandó műholdfelvételek, mind az elkészített eredmények is nagy mennyiségű helyet tudnak foglalni.

2.2.2. Telepítési útmutató

A szoftver futtatásához kell 3.10.-es Python verzió, emellett a működéshez szükség van számos csomagra is. Ehhez ajánlott az Anaconda²-ban, vagy hasonló programon belül egy virtuális környezet létrehozása. Továbbiakban az Anaconda-ban való telepítést fogom bemutatni.

Első lépés az Anaconda Prompt megnyitása, ezen belül a virtuális környezet létrehozása, és annak aktiválása a következő parancsokkal 2.1 (myenv-et tetszőleges névvel lehet helyettesíteni):

```
1 conda create -n myenv python=3.10
2 conda activate myenv
```

2.1. forráskód. Anaconda virtuális környezet létrehozása, aktiválása

²<https://www.anaconda.com/>

Ezután telepíteni kell a futáshoz szükséges csomagokat. Ezek egy részét pip-pel (*Package Installer for Python*) 2.2 telepítjük.

```
1 pip install numpy
2 pip install matplotlib
3 pip install scikit-image
4 pip install opencv-python
```

2.2. forráskód. Csomagok telepítése pip-pel

Többi csomaghoz létrehozuk a conda-forge 2.3 csatornát, amin keresztül fogjuk letölteni őket.

```
1 conda config --add channels conda-forge
2 conda config --set channel_priority strict
```

2.3. forráskód. Conda Forge létrehozása

```
1 conda install -c conda-forge gdal
2 conda install -c conda-forge python-fmask
```

2.4. forráskód. Conda Forge csomagok telepítése

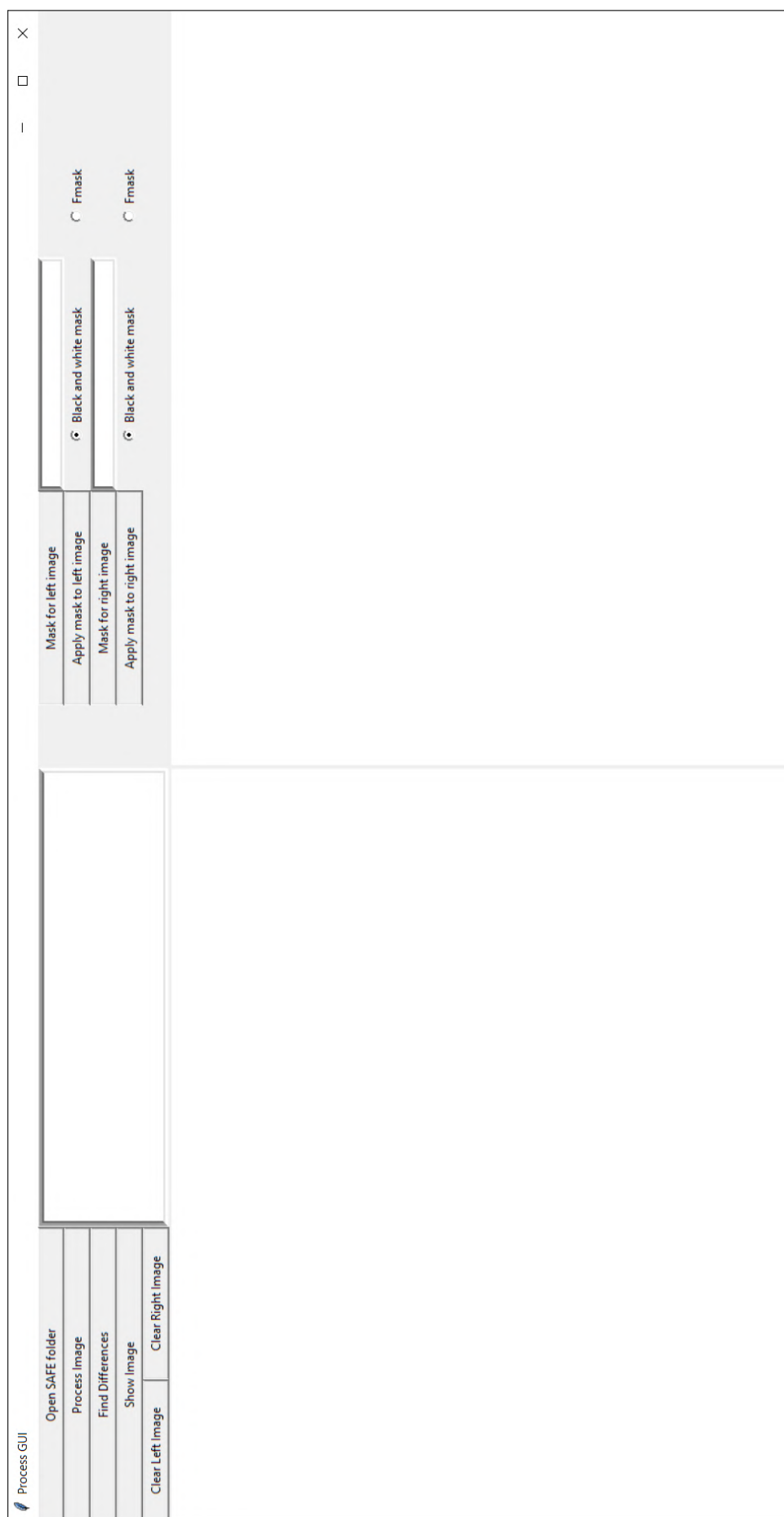
Ezen csomagok letöltése után készen állunk futtatni a szoftvert. Ehhez megint az Anaconda Prompt-ra lesz szükség. Miután ezt megnyitottuk aktiválnunk kell a létrehozott környezetünket. Ha a környezet nem abból a mappából fut, ahol a program található akkor indítás előtt oda kell navigálnunk. Ezt követően a következő 2.5 paranccsal indíthatjuk el az alkalmazást.

```
1 python __main__.py
```

2.5. forráskód. Az alkalmazás futtatásához szükséges parancs

2.3. Használati útmutató

A program elindítása után a következő kezdőképernyővel fogunk találkozni 2.3. Ezen található a főmenü, ahol betölthetjük a feldolgozni kívánt Sentinel-2 felvételeket, ezeket megjeleníthetjük, majd elindíthatjuk feldolgozásukat. Emellett látható a képek maszkolására szolgáló menü, ahol a képekre helyezhetjük az elkészült eredményeinket, ezzel hangsúlyozva a releváns területeket. Alul pedig két a képek megjelenítésére szolgáló keret látható.



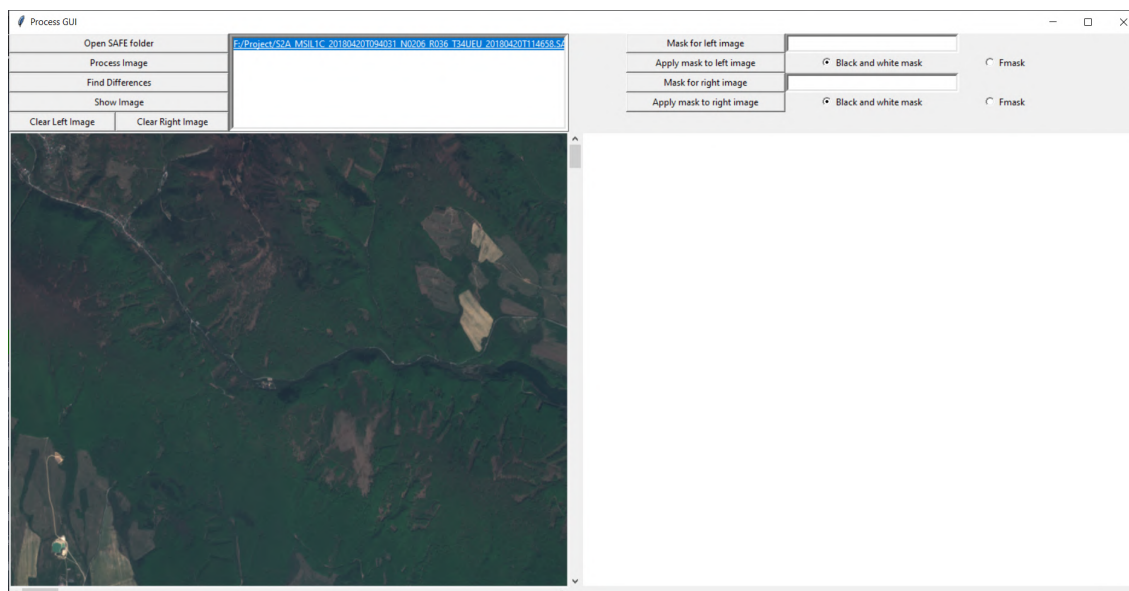
2.3. ábra. A GUI kezdőképernyője

A szoftver használata során az első lépés mindig egy Sentinel-2 felvétel SAFE mappájának megnyitása kell, hogy legyen. Ez a kezdőképernyő bal felső menüjének **2.4 Open SAFE Folder** gombjára kattintva tehető meg. Itt csak Level-1C feldolgozottságú, még meg nem nyitott Sentinel-2 SAFE mappákat fogad el megnyitásra a program.



2.4. ábra. A főablak menüje kezdőállapotban, megnyitott mappák nélkül

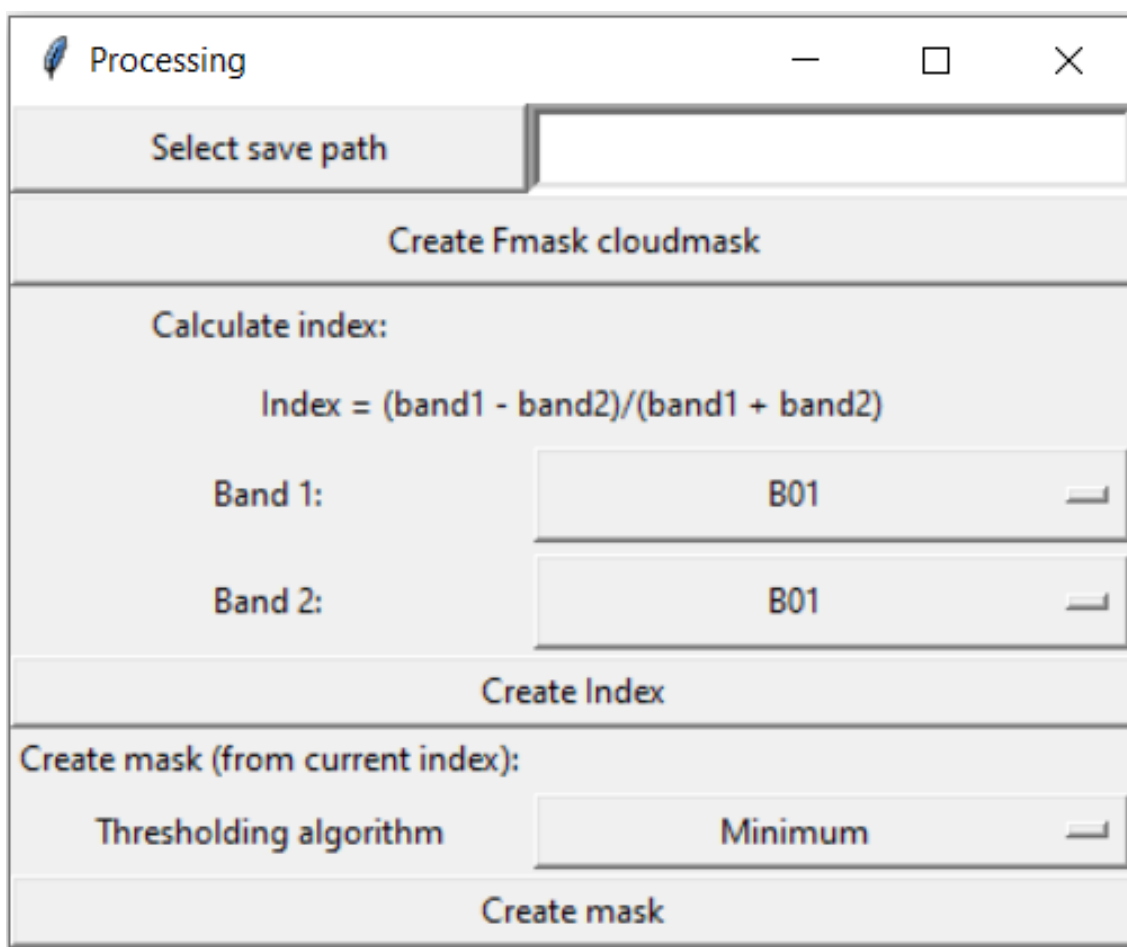
Miután legalább egy Sentinel-2 mappát megnyitottunk, megnyílik a lehetőségünk a színes műholdkép megjelenítésére 2.5. Ezt a **Show Image** gombbal tehetjük meg. Erre kattintva a program elkezd betölteni a kiválasztott képet, egyik éppen szabad keretbe. Ha a bal keret szabad, akkor ott fogja megjeleníteni a képet, ha a bal nem, de a jobb szabad akkor abba helyezi. Olyan esetben, amikor nincsen szabad keret a program felszólít, hogy megjelenítés előtt ki kell üríteni egyiket. Ez a **Clear Left Image** és a **Clear Right Image** gombokkal tehető meg.



2.5. ábra. A GUI egy képpel a bal keretben

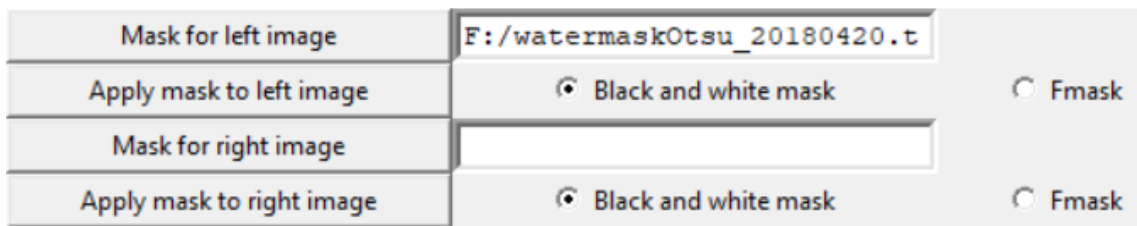
Miután egy keretben megjelenítettünk egy képet odakerülnek a szélére a görgők, amikkel mozgatható lesz. Abban az esetben, amikor mindkét keretben látható kép az egér mozgásával szinkronban tudjuk őket mozgatni. A képek pozícióját jobb egérgomb lenyomásával tudjuk összehangolni, ha erre szükség van.

Lehetőség van még a kiválasztott képre elvégezni a folyamat egyes lépéseit, azaz elkészíteni a felhőmaszkot, indexet kiszámítani, vagy küszöböléssel maszkot készíteni egy indexből. Ez a **Process image** gomb lenyomásával tehető meg. Itt nincs korlátozva a folyamat vízmaszkokra, bármely két sáv kiválasztható és belőlük index számítható. Ez a 2.6 ábrán láthatófelületen végezhető el.



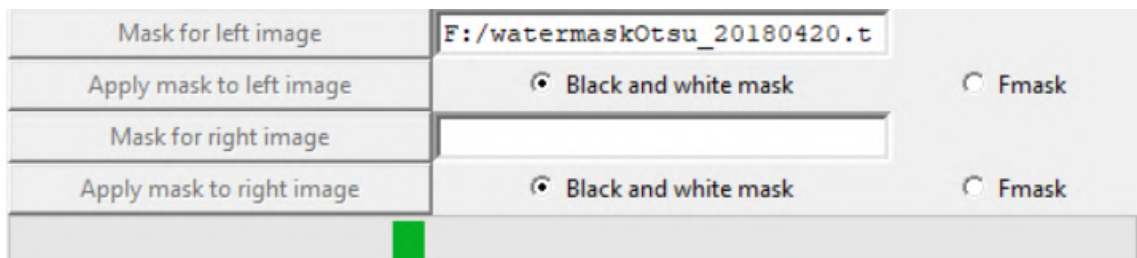
2.6. ábra. A folyamatok külön-külön erről a felületről végezhetőek el.

Megjelenített képre lehetőségünk van maszkot 2.7 helyezni. A bal képre szánt maszk útvonalát a **Mask for left image** gombbal lehet betölteni. A folyamat során készült vízmaszk, különbségmaszk vagy hasonló fekete fehér színekből álló maszk ráhelyezéséhez a **Black and White mask**, az fmask-kal készültekhez, pedig az **Fmask** opció kiválasztása szükséges. Majd az **Apply mask to left image** lenyomásával fogja a szoftver a megjelenített képet maszkolni, és azt megjeleníteni 2.9 a felületen, A jobb keretben lévő kép maszkolása ugyanígy működik a **Mask for right image** és az **Apply mask to right image** gombok használatával.

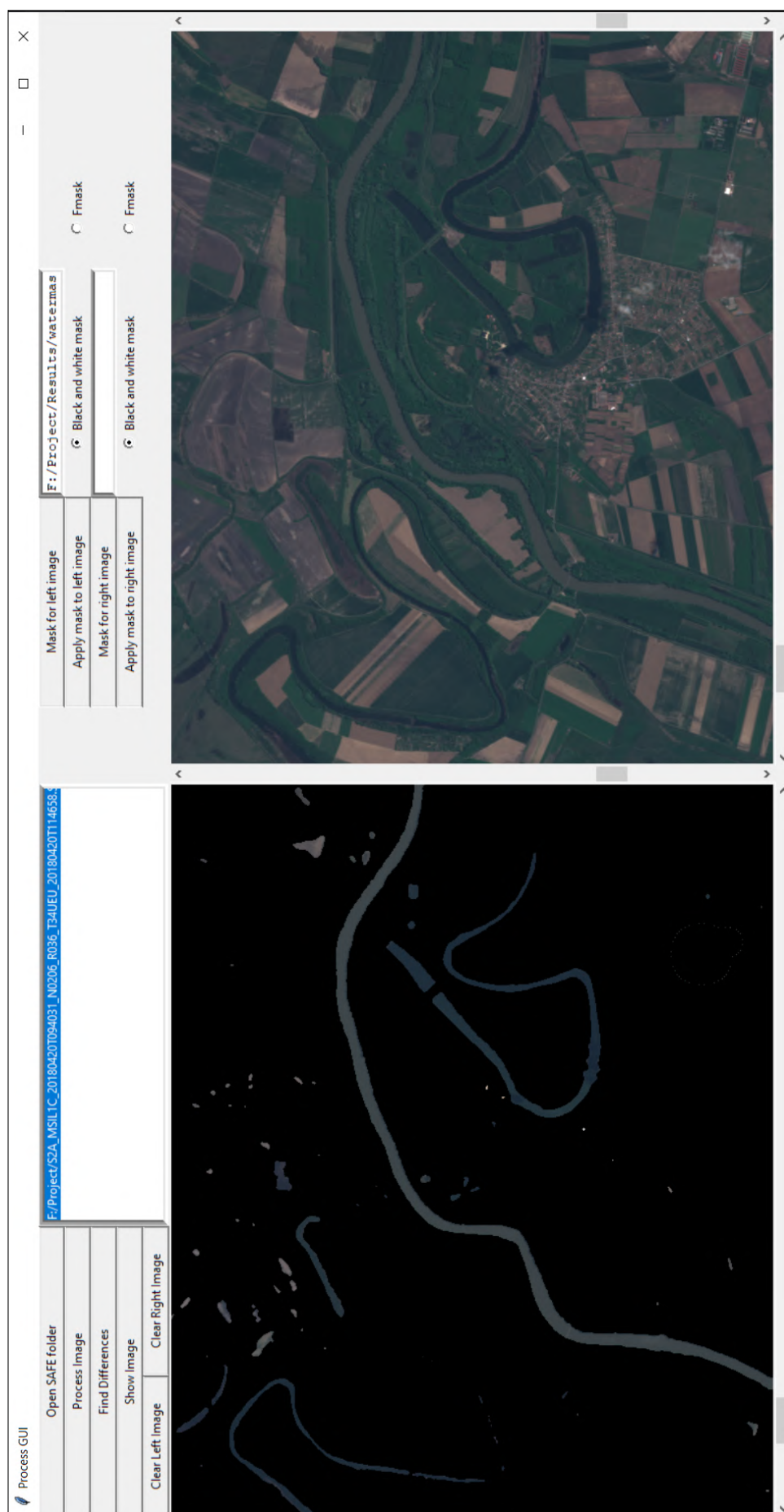


2.7. ábra. A GUI maszkolás menüje

Mind a kép megjelenítése, mind a maszkolás közben a program egy folyamatjelző sávot 2.8 fog megjeleníteni. Eközben a többi funkció le lesz zárva, és csak az éppen futó befejezése után indítható új.



2.8. ábra. A maszkolás menü folyamat futása közben



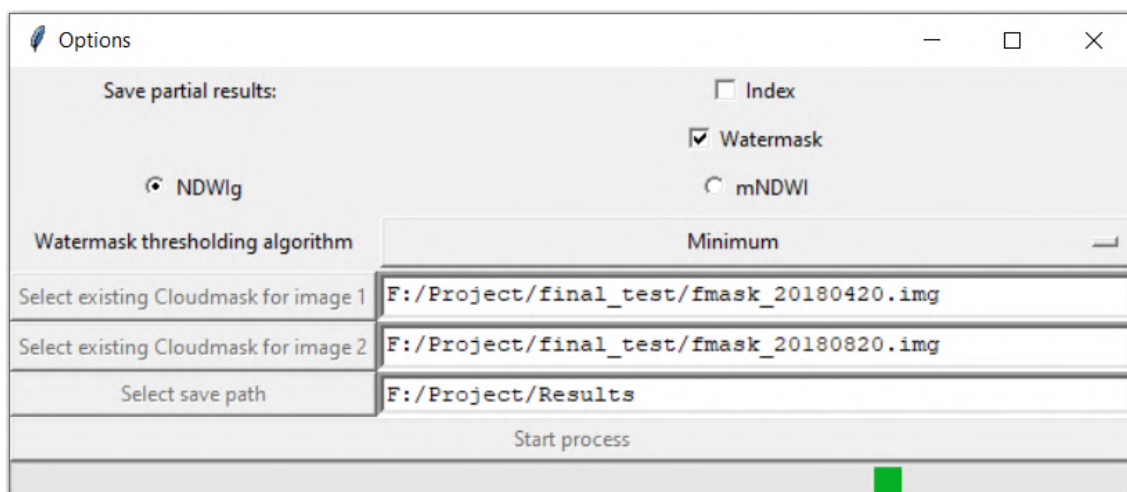
2.9. ábra. A GUI egy maszkolt és egy tiszta Sentinel-2 képpel megjelenítve

Miután betöltöttünk két különböző időpontbeli Sentinel-2 mappát ugyanarról a csempéről, ezek kijelölésével már használni tudjuk a **Find Differences** gombot. Ez meg fog nyitni egy beállításokat tartalmazó 2.10 ablakot, ahonnan futtatni fogjuk majd tudni a folyamatot.



2.10. ábra. A folyamat beállításait tartalmazó ablak

A beállításoknál eldönthetjük, hogy a folyamat részeredményeit, az index képet és a vízmaszkot szeretnénk-e elmenteni. Megjegyzem, hogy az elkészített felhőmaszkot mindenképpen elmenti a szoftver. Ki kell választani, hogy NDWIg vagy mNDWI index alapján szeretnénk elkészíteni a vízmaszkokat, valamint eldöntendő még az is, hogy melyik küszöbölő algoritmust használja a folyamat. A futás gyorsítható már elkészített felhőmaszkok elérési útvonalának megadásával. Ezek a **Select existing Cloudmask for image 1** és **Select existing Cloudmask for image 2** gombokkal tölthetők be. Fontos megjegyezni, hogy a képek sorrendje nem időrendi, hanem abban a listabeli sorrendjük alapján dől el, ahonnan kiválasztva lettek. Végül azt a mappát lehet megadni, ahova szeretnénk, hogy a szoftver mentse a folyamat során kapott eredményeit, ha itt nem adunk meg semmit akkor abba a mappába fogja ezeket menteni, ahonnan futtatjuk a programot. Miután minden beállítást megadtunk a **Start process** gomb lenyomásával indíthatjuk el 2.11 a folyamatot. Miután a folyamat teljesen lefutott a beállítások ablak be fog záródni, és visszakerülünk a kezdőképernyőre.



2.11. ábra. A folyamat futás közben

Megjegyzés. A folyamat futási ideje jelentősen függ a megadott beállításoktól. Ha korábban elkészített felhőmaszkot használunk és nem mentünk el részeredményeket, mindössze 1,5 percbe telik, hogy végezzen. Viszont, ha elmentjük a részeredményeket, vagy új felhőmaszkot készítünk a futási idő körülbelül 6 percre nő, ha mindkettőt megtesszük már csak nagyjából 10 perc alatt fog végezni a folyamat.

Futtatás során minden file georeferált formátumban kerül elmentésre, így ezek az alkalmazáson kívül is használhatóak lesznek. Megtekinthetők térképészeti programokban (pl.: QGIS³), és ráilleszthetők minden ugyan ezt a területet fedő Sentinel-2 felvételre.

³<https://qgis.org/en/site/>

3. fejezet

Fejlesztői dokumentáció

3.1. Fejlesztési környezet

A szoftver Python nyelven készült, ennek 3.10-es verzióját használva. Képek feldolgozására az OpenCV¹, a scikit-image² és a Pillow³ csomagok kerültek felhasználásra. A térinformatikai adatok kezelése a GDAL⁴, a felhőmaszkok létrehozása pedig a python-fmask⁵ csomaggal történt. A grafikus felület a tkinter⁶ segítségével készült. A kód PyCharm-ban⁷ íródott.

3.2. Szoftverarchitektúra

A szoftver MVC - azaz Modell, Nézet, Vezérlő - architektúrával lett létrehozva. Itt a modell a logikát, nézet csak a megjelenítést tartalmazza. A felhasználó, pedig a vezérlőt használja, ami irányítja a modellt és frissíti a nézetet. Az architektúra vázlata a 3.1 képen látható.

¹https://docs.opencv.org/4.x/d6/d00/tutorial_py_root.html

²<https://scikit-image.org/>

³<https://pillow.readthedocs.io/en/stable/>

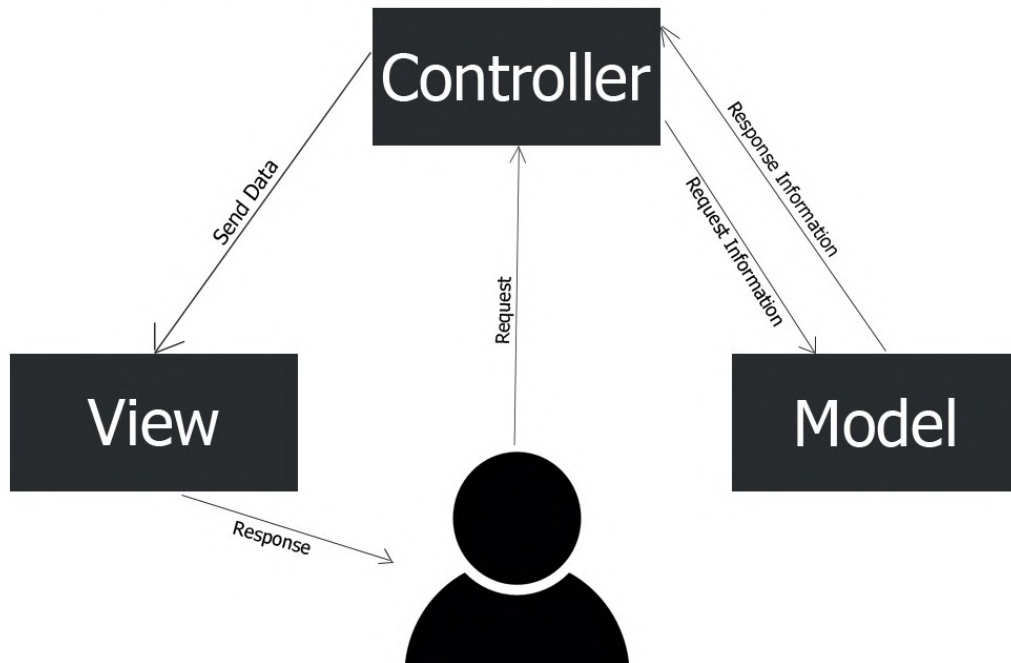
⁴<https://gdal.org/>

⁵<https://www.pythonfmask.org/en/latest/>

⁶<https://docs.python.org/3/library/tkinter.html>

⁷<https://www.jetbrains.com/pycharm/>

Model-View-Controller



3.1. ábra. Az MVC architektúra ábrázolása

Forrás: https://medium.com/@joespinelli_6190/mvc-model-view-controller-ef878e2fd6f5

3.3. Tervezés

3.3.1. Funkcionális követelmények

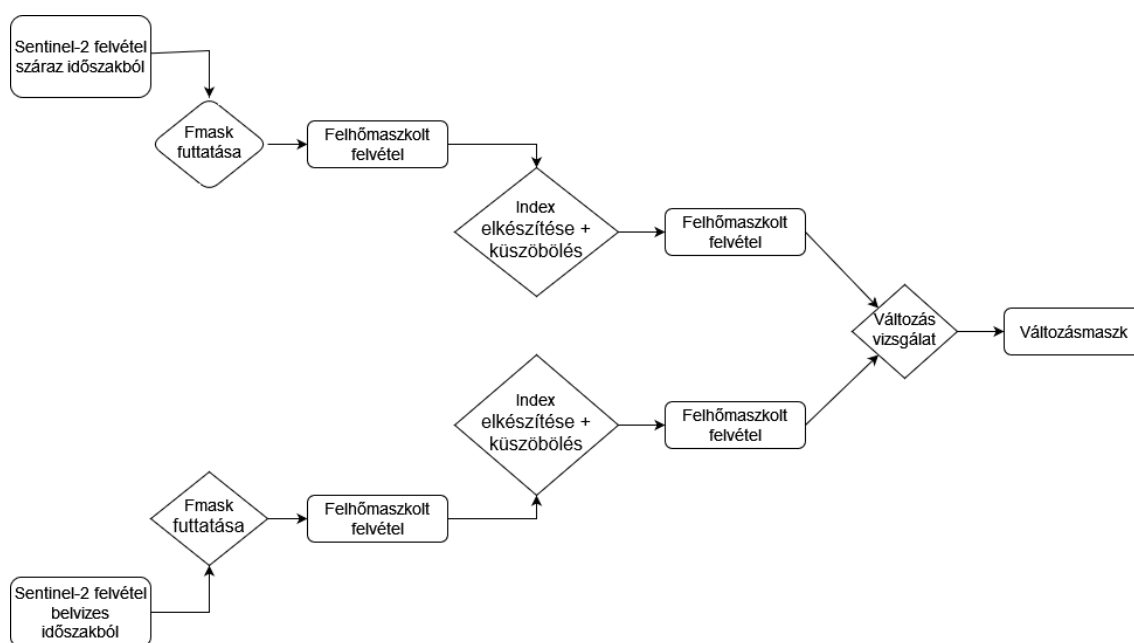
A programnak képesnek kell lenni a kapcsolódó kutatás során legjobb eredményeket mutató módszereket végrehajtania. Ennek lépései a következők:

- Felhő és felhőárnyékmásk elkészítése
- Indexek kiszámítása
- Indexek küszöbölése
- Különbségvizsgálat vízmaszkokon
- Eredmények GeoTiff formátumban való elmentése

Cél, hogy bármilyen a követelményeknek megfelelő Sentinel-2 felvételen működőképes legyen. Ezeken felül az ellenőrzés megkönnyítéséhez, a felvételek megjelenítése és maszkolása szükséges. Valamint egyes lépések külön futtatására is ad lehetőséget.

3.3.2. Kiválasztott módszerek

A belvíz detektálás folyamatát a szoftver három részben végzi el. Első lépésként előfeldolgozást végez el a műholdfelvételeken, hogy kiszűrje a felhőket és felhőárnyékokat. Második lépésben vízmaszkot készít a felhőmaszkolt képekből. Ehhez kiszámítja a felvételek bizonyos indexét, majd ezt küszöböléssel bináris maszkká alakítja át. Végül a harmadik lépés változásdetektálás az elkészült vízmaszkokon. Ezeknek a részfeladatoknak egy folyamatábrája a 3.2 ábrán látható.



3.2. ábra. A folyamat flowchart-ja

Felhőmaszkolás

A szoftver ebben a lépésben az fmask[4] algoritmust futtatja le mindkét Sentinel-2 képre. Az eredmény így két olyan maszk lesz, ami meghatározza a felhő, felhőárnyék, a víz és a hó pixeleket. A feladathoz ez a vízmeghatározás nem elég pontos, és havas területek nem relevánsak. Ezért az elkészült fmask-ból ezeket kitöröli, és csak felhő és felhőárnyék pixeleket takarja ki a feldolgozandó műholdfelvételekből. Mindegyik képre a folyamat elején ráhelyezi a saját felhőmaszkját, majd a változásvizsgálat elején ezeket összeeszeljük, és mindkét vízmaszkra ráhelyezzük, a felhőkből adódó különbségek detektálásának elkerüléséhez.

Vízmaszk készítés

A programban a kutatás során legjobbnak talált módszerek vannak implementálva, így a futtatás során az NDWI_g és mNDWI indexek számíthatóak ki, a 3.1 táblázatban látható módon. Ehhez megnyitja a megadott Sentinel-2 mappákból a szükséges sávokat, majd kiszámolja a normalizált különbségüket. Így kapott eredményeket, pedig három különböző módszerrel képes küszöbölni, azaz bináris képpé alakítani.

$$\text{NDWI}_g = \frac{\text{SWIR1}(B11) - \text{Red}(B04)}{\text{SWIR1}(B11) + \text{Red}(B04)} \quad \Bigg| \quad \text{mNDWI} = \frac{\text{Green}(B03) - \text{SWIR1}(B11)}{\text{Green}(B03) + \text{SWIR1}(B11)}$$

3.1. táblázat. A felhasznált indexek kiszámításának képletei.

Ezek közül az első az Otsu algoritmus [5], ez egy automatikus küszöbérték-választási módszer. A képet két osztályra bontja, és azt a küszöbértéket keresi, ami minimalizálja az osztályon belüli varianciát, amit a két osztály varianciáinak súlyozott összegeként határoz meg. Az algoritmus előnye, hogy csak a szürkeségi szint hisztogramjára van szükség a kép küszöbértékének meghatározásához.

Második a Minimum algoritmus [6]., ez az Otsuhoz hasonlóan csak hisztogramot igényel. Ezt iteratív módon simítja egy 3 méretű futó átlag segítségével, amíg csak két helyi maximum nem lesz. Ezután a kettő közötti minimumot választja meg a küszöbértéknek. Az olyan képekre, amiknek hisztogramja rendkívül egyenlőtlen csúcsokkal vagy széles és lapos völgygel rendelkezik, nem alkalmazható ez a módszer.

Utolsó pedig a Yen algoritmus [7], ez egy kritériumot javasol a többszintű küszöbérték meghatározására, ami két tényezőt vesz figyelembe. Az első a küszöbölt és az eredeti kép közötti eltérés, a második pedig a küszöbölt kép ábrázolásához szükséges bitek száma. A kétszintű küszöbértékelés új maximális korrelációs kritériuma alapján meghatározza az eltérést, majd a többszintű küszöbértékeléshez egy olyan költségfüggvényt javasol, ami mindkét tényezőt figyelembe veszi. A költségfüggvény minimalizálásával automatikusan meghatározható a szürkeségi szintek osztályozási száma és a küszöbértékek.

Változásdetektálás

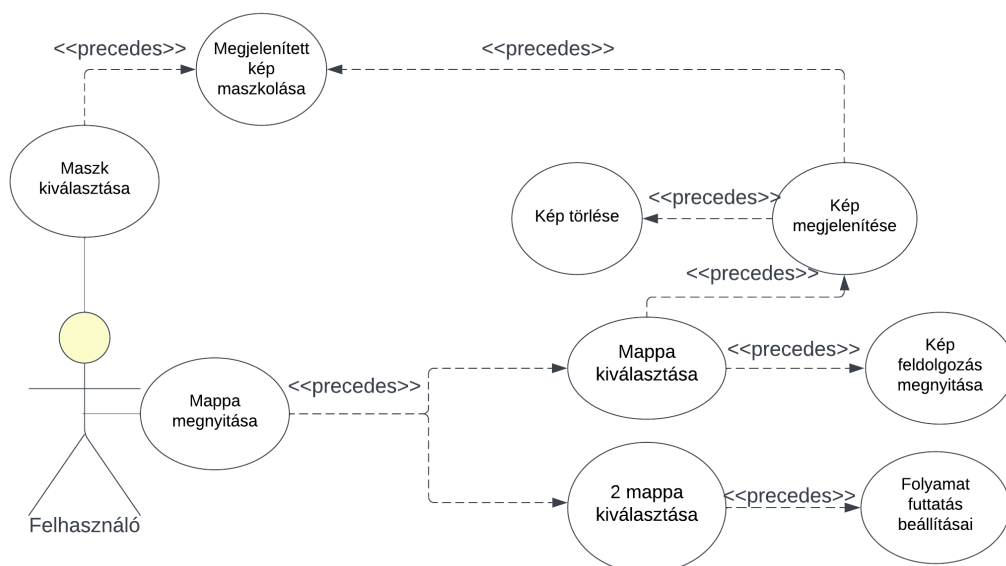
Folyamatunk utolsó lépése a változásdetektálás, ahol az elkészült vízmaszkokat hasonlítja össze egymással. Ennek a lépésnek a célja egy olyan maszk elkészítése, ahol

csak a két vízmaszk különbségei jelennek meg. Így kiszűrve az állandó vízfelszíneket, például tavakat és folyókat, de meghagyva az időszakosan feltöltődő területeket.

Ezt a lépést a strukturális hasonlóságot összehasonlító (SSIM) algoritmussal[8] végzi el. Ehhez a scikit-image csomagban elkészült implemetációt használjuk fel (structural_similarity metódus).

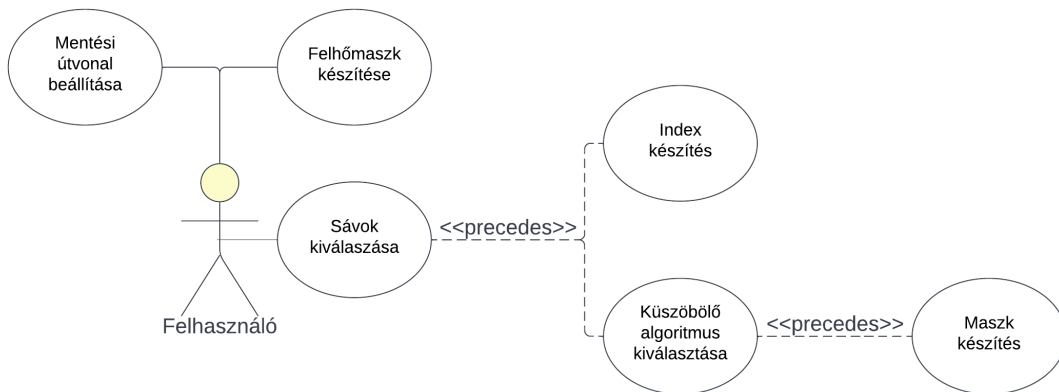
3.3.3. A szoftver felhasználói esetei

A program három különböző képernyőt tartalmaz. Első a főablak, ami futtatáskor megjelenik. Itt nyithatóak meg a Sentinel-2 mappák, ezen jelennek meg a felvételek, és a maszkolásuk is itt végezhető el. Megnyitás után az egyetlen működő gomb, a mappák megnyitására szolgáló **Open SAFE Folder**. Miután be lett töltve egy mappa, használhatóvá válik több funkció. Megjeleníthető lesz a mappa kijelölésével a színes Sentinel-2 kép, valamint ez maszkolható lesz. Ezen kívül fel is dolgozható a megnyitott felvétel a **Process Image** gomb nyomásával. Ez egy új ablakot nyit meg, ahonnan a folyamatok indíthatóak lesznek. Két azonos területen készült felvétel megnyitása után megnyitható a folyamat indítását kezelő ablak a **Find Differences** gombbal. A fő ablak megjelenését a *GuiView* osztály tartalmazza, használati eseteit a 3.3 képen találhatjuk.



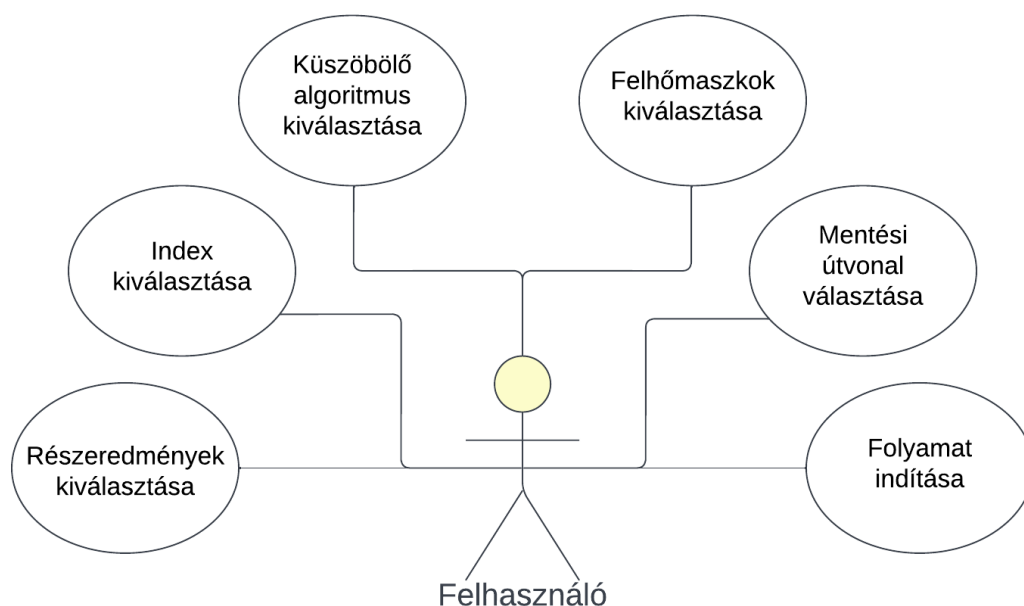
3.3. ábra. A GuiView osztály használati esetei

Az egyes képek feldolgozásáért felelős ablakon futtatható az fmask algoritmus, kiszámítható a kép bármely indexe, valamint maszk is készíthető ezekből. Első opció mentési útvonal megadása az itteni eredményeknek, ha ez nem történik meg akkor a program futtatási helyére fognak kerülni az adatok. A **Create Fmask cloudmask** gomb a felhőmaszkolás folyamatát indítja el. Következő opció az index számítás, ez a felületen látható képlet alapján lesz kiszámítva a megadott sávokkal. Ugyanitt lehet megadni, hogy a **Create Mask** gomb lenyomása után melyik index alapján készüljön a maszk. A használandó küszöbölő algoritmus is választható. Ennek a felületnek a megjelenését a ProcessingView osztály tartalmazza, használati esetei a 3.4 ábrán láthatóak.



3.4. ábra. A ProcessingView osztály használati esetei

Teljes folyamat egy harmadik ablakból indítható el. Ennek megjelenítéséhez a főablakon ki kell választani két, azonos csempéről, de különböző időpontban készült Sentinel-2 felvételt. Az új felületen kiválasztható, hogy részeredményeket (index kép, vízmaszk) el akarjuk-e menteni. Majd beállítható a használni kívánt index, és hogy melyik küszöbölő algoritmust segítségével készüljön el a vízmaszk. Az fmask futási idejének hossza miatt lehetőség van korábban létrehozott felhőmaszkok felhasználására. Végül beállítható, hogy hova mentse az eredményt a szoftver. A felület megjelenéséért felelős kód az OptionsView osztályban található, használati esetei a 3.5 ábrán láthatóak.



3.5. ábra. Az OptionsView osztály használati esetei

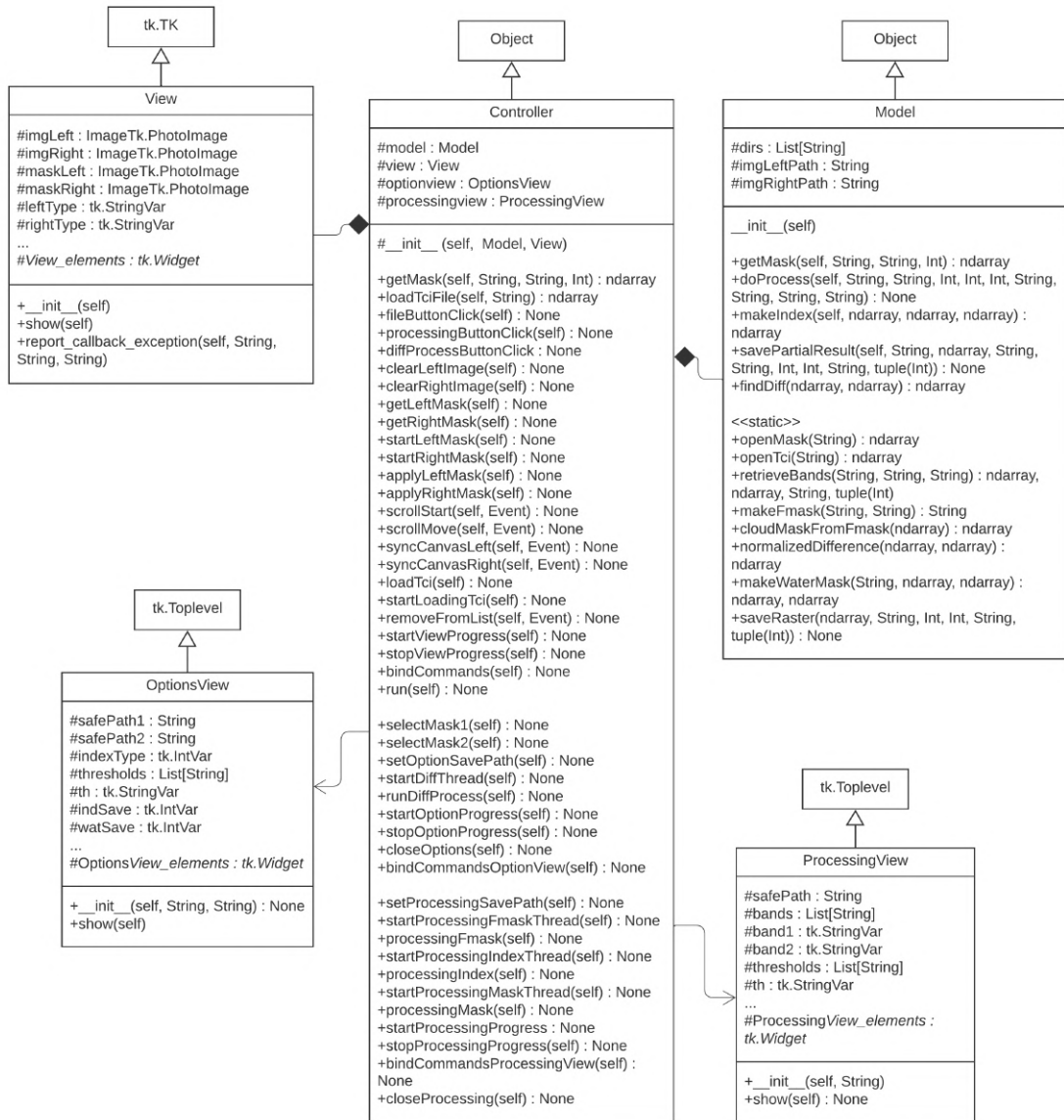
3.3.4. A szoftver felépítése

A program öt osztályból áll: a modell(GuiModel), a vezérlő(GuiController) és a három ablak nézetei(GuiView, ProcessingView, OptionsView). Ezek UML diagramja a 3.6 ábrán látható.

A modell tartalmazza a szoftver logikáját, ez felelős a file-ok kezeléséért, valamint az időszakos belvizeket detektáló folyamat futtatásáért. A *dirs* adattagja tárolja a főablakban megnyitott mappákat, az *imgLeftPath* és *imgRightPath* pedig az éppen megjelenített képek útvonalát tárolják.

A detektáló folyamat a **doProcess** metódus meghívásával indítható el. Ennek argumentumai az összehasonlítandó SAFE mappák elérési útvonalai, valamint az OptionsView-ban megadott beállítások. Futása során meghívja a részfolyamatok elvégzésére szolgáló metódusokat. A használandó sávokat a **retrieveBands** nyitja meg, és alakítja át kezelhető formátumba. A felhőmaszk elkészítését a **makeFmask**, ennek megfelelő formátumba alakítását a **cloudMaskFromFmask** végzi. A **makeIndex** ráhelyezi a felhőmaszkot a sávokra és a **normalizedDifference** meghívásával számítja ki a továbbiakban használt indexet. Vízmaszk készítése a **makeWaterMask** metódussal készül, ennek argumentumban adjuk meg a használni kívánt indexeket és a kiválasztott küszöbölő algoritmus nevét. A változások detek-

tálását a **findDiff** metódus végzi el. Az eredményeket a **saveRaster** menti el. A részfeladatokat ugyanezek a metódusok végzik el külön-külön meghívva.



3.6. ábra. A szoftver UML diagramja

A vezérlő biztosítja a szoftver és a felhasználó közötti kommunikációt. Itt találhatók a nézetekben létrehozott gombok által meghívott metódusok, és ez hívja meg a modell metódusait is. Saját adattagjai nincsenek, de itt kapnak értéket a többi osztálybeliek. A kód nézetenként három részre van bontva: első a *GuiView* elemihez rendelt metódusokat és ezek segédjeit tartalmazó rész, második az *OptionsView*-ért felelős, utolsó pedig a *ProcessingView*-hoz tartozó rész. Mindegyik nézethez tartozik ezeken kívül egy haladásjelző sáv, aminek a kezelése külön van véve, ennek indításakor mindig kikapcsolódnak az adott ablak gombjai, és csak az elindított folyamat

elvégzése után válnak újra használhatóvá. Ezért a működésért az adott nézethez tartozó **startProgress** és **stopProgress** metódusok felelnek. A hosszabb folyamatok futtatásához mindig külön szálat indítunk, hogy elvégzésük közben ne váljon használhatatlanná a program. A metódusok nézetekhez rendelését a **bindCommands** kezdetű függvények végzik el. Itt található a program futtatásáért felelős **run** metódus is.

A nézetek csak a konstruktorukat, valamint a megjelenítésükért felelős **show** metódust tartalmazzák. A konstruktorukban példányosítva vannak a felület részei, és ezek elhelyezésre kerülnek egy rácsban.

3.4. Tesztelés

3.4.1. Felhasználói felület tesztelése

A szoftver tesztelése manuálisan lett elvégezve, ennek részleteiről a 3.2, 3.3, 3.4 táblázatokban olvashatunk.

Funkció	Teszteset	Kimenetel
Open SAFE folder	Még meg nem nyitott, Level-1C feldolgozottságú, SAFE mappa kiválasztása	A kiválasztott mappa megjelenik a listában, és a modell eltárolja az útvonalat.
	Korábban megnyitott mappa kiválasztása	Jelzés, hogy a mappa már korábban ki lett választva, a mappa nem jelenik meg, a modellben nem tárolódik el
	Nem megfelelő feldolgozottságú a Sentinel-2 felvétel	Jelzés, hogy csak Level-1C feldolgozottságú felvétel dolgozható fel, a mappa nem jelenik meg, a modellben nem tárolódik el
	Nem Sentinel-2 felvétel SAFE mappája kiválasztva	Jelzés, hogy csak SAFE mappa választható ki, a mappa nem jelenik meg, a modellben nem tárolódik el

Funkció	Teszteset	Kimenetel
Process Image	Egy Sentinel-2 felvétel kiválasztva	Megnyílik a kép feldolgozására szolgáló ablak
	Nulla, vagy egynél több felvétel kiválasztva	Jelzés, hogy egy Sentinel-2 felvételnek kell kiválasztva lennie
	Egy Sentinel-2 felvétel kiválasztva, az ablak már meg van nyitva	Jelzés, hogy a feldolgozó ablak már meg van nyitva
Find Differences	Kettő, azonos csempéről készült Sentinel-2 felvétel kiválasztva	Megnyílik a folyamat beállításait tartalmazó ablak
	Kettőtől eltérő számú felvétel kiválasztva	Jelzés, hogy két felvételt kell kiválasztani
	Két különböző területről készült felvétel kiválasztva	Jelzés, hogy azonos területről készült felvételek szükségesek
	Két felvétel kiválasztva, az ablak már meg van nyitva	Jelzés, hogy az ablak már meg lett nyitva
Show Image	Egy felvétel kiválasztva, van üres hely a képnek	Elindul a kép megjelenítési folyamat, és ennek végeztével balról jobbra haladva az első üres keretben megjelenik a kép, modellben tárolódik a megjelenített kép útvonala
	Egytől eltérő számú felvétel kiválasztva	Jelzés, hogy egy felvételnek kell kiválasztva lennie
	A kiválasztott felvétel mapájából hiányzik a TCI sáv	Jelzés, hogy nem található a megjelenítéshez szükséges sáv
	Egy felvétel kiválasztva, de nincs üres hely	Jelzés, hogy új kép megjelenítése előtt el kell távolítani egy képet

Funkció	Teszteset	Kimenetel
Clear Left Image	Van megjelenítve kép a bal oldali helyen	A megjelenített kép eltűnik, modellbeli adattag ki lesz törölve
	Nincs megjelenítve kép	Nem történik semmi
Clear Right Image	Van megjelenítve kép a jobb oldali helyen	A megjelenített kép eltűnik, modellbeli adattag ki lesz törölve
	Nincs megjelenítve kép	Nem történik semmi
Apply mask to left image	Helyes maszk lett kiválasztva, helyes típus lett megadva, van bal oldalon megjelenítve kép	Megkezdődik a megjelenített kép maszkolása, majd bal képkeretben megjelenik a maszkolt kép
	Helyes maszk lett kiválasztva, helyes típus lett megadva, nincs bal oldalon megjelenítve kép	Jelzés, hogy nincs megjelenítve kép bal oldalon
	Helyes maszk lett kiválasztva, helytelen típus lett megadva	Jelzés, hogy nem kiválasztott típusú a maszk
	Nem létező maszk lett kiválasztva	Jelzés, hogy kiválasztott maszk nem létezik

3.2. táblázat. A GuiView funkcióinak tesztelése

Funkció	Teszteset	Kimenetel
Create Fmask	Létező mentési útvonal kiválasztva	Elindul az fmask futtatása
	Hibás mentési útvonal megadása	Jelzés, hogy a megadott mentési útvonal nem létezik

Funkció	Teszteset	Kimenetel
Create Index	Jó sávok, és mentési útvonal megadva	Elindul az index készítés folyamata
	Hibás mentési útvonal megadása	Jelzés, hogy a megadott mentési útvonal nem létezik
	Azonos sávok kiválasztva	Jelzés, hogy két különböző sávot kell megadni
	Jó sávok kiválasztva, de valamelyik nem létezik a Sentinel-2 map-pában	Jelzés, hogy egyik kiválasztott sáv nem található
Create Mask	Jó sávok, és mentési útvonal megadva	Elindul az index készítés folyamata
	Hibás mentési útvonal megadása	Jelzés, hogy a megadott mentési útvonal nem létezik
	Azonos sávok kiválasztva	Jelzés, hogy két különböző sávot kell megadni
	Jó sávok kiválasztva, de valamelyik nem létezik a Sentinel-2 map-pában	Jelzés, hogy egyik kiválasztott sáv nem található

3.3. táblázat. A ProcessingView funkcióinak tesztelése

Funkció	Teszteset	Kimenetel
Start Process	Részeredmények elmentésének megjelölése	Elindul a folyamat futtatása, és ennek során elmentésre kerülnek a kijelölt részeredmények
	Létező felhőmaszk kiválasztva	Elindul a folyamat futtatása, de az fmask futtatásra helyette megnyitja a megadott maszkokat
	Nem létező felhőmaszk kiválasztva	Elindul a folyamat, de generál új felhőmaszkokat
	Nincs felhőmaszk kiválasztva	Elindul a folyamat, és generál új felhőmaszkokat

Funkció	Teszteset	Kimenetel
Start Process	Hibás mentési útvonal megadása	Jelzés, hogy a megadott mentési útvonal nem létezik
	Hiányzó sávok a vizsgált felvételek mappáiban	Jelzés, hogy szükséges sáv nem található

3.4. táblázat. Az OptionsView funkcióinak tesztelése

A tesztek elvégzéséhez használt file-ok az inf-es levelezőrendszerből elérhető Onedrive-on elérhetőek a következő linken: https://ikelte-my.sharepoint.com/:f:/g/personal/auk0tg_inf_elte_hu/EmzhpU3SkzNLszGWJAN1uSoBC1oPji4vUojq82rV3M0pdA?e=ULfzlk.

A részfolyamatok futása közben változhatnak ki más hibák is, de mivel ezek Python csomagok metódusaival lettek lefuttatva, őket nem teszteltem. Ha valamilyen előfordul a hibaüzenetet hasonló módon továbbítjuk a felhasználó felé, mint az eddigieket.

4. fejezet

Összegzés

Az időszakos belvizek detektálása több területen is hasznos. Ha ezeket a helyeket meg tudjuk találni, akkor megkönnyíthetünk például a mezőgazdasági és várostervezési munkákat. Ennek a folyamatnak automatizálásával segíthetjük az ezzel foglalkozó szakemberek munkáját. Alkalmazásomban erre hoztam létre egy megoldást.

Az elkészített megoldás nagyrészt megegyezik a témabejelentőben leírttal. A felhőmaszkoló algoritmust az s2cloudless-ről az fmask-ra cseréltem, mivel az utóbbi képes a felhőárnyékok detektálására is. A vízmaszkok elkészítéséhez az NDWIg és mNDWI indexek közül választhatunk. A küszöbölést az Otsu, Minimum és Yen algoritmusokkal végezhetjük el. Majd ezeken végzünk változásvizsgálatot. A detektáló folyamat futtatásán kívül Sentinel-2 felvételek, és hozzájuk készült maszkok az alkalmazásunk felületén megjeleníthetők. Ezen kívül az eredményeink GeoTIFF formátumban elmenthetők.

Összegzésként elmondhatjuk, hogy a létrehozott alkalmazás képes elvégezni a meghatározott lépéseket akár egyesével, akár a folyamat részeként. A grafikus felület képes a felvételek megjelenítésére, és az eredmények vizsgálatára. A szoftver nem igényel hozzáértést a futtatáshoz, így egyaránt alkalmazható szakemberek és amatőrök által.

Továbbfejlesztési lehetőségek

A legnagyobb fejlesztési irány a feldolgozható adatok bővítése lenne. A kutatás során vizsgálva lettek PlanetScope műhold felvételei is, így ezek inkorporálása lenne a legnyilvánvalóbb. A folyamat kis módosításokkal alkalmazható lenne Landsat műhold képeire, valamint akármilyen felvételre, ami tartalmazza a felhasznált sávokat.

Fontos szempont még, hogy a képek felbontása nagyban befolyásolja a kapott eredmények minőségét. Minél kisebb területet fednek le a felvétel pixelei, annál nagyobb pontosságú eredményt tudunk elérni.

A futási idő továbbá javítható lenne a folyamat párhuzamosításával. Főként a részeredmények mentésére lenne ez alkalmazható, ami a folyamat leghosszabb időt igénylő lépése.

Köszönetnyilvánítás

A kutatási munka és szakdolgozat az ELTE Informatikai Kar és az InforNess Training Kft. ösztöndíjas pénzügyi támogatásával, továbbá a Lechner Tudásközpont szakmai támogatásával valósult meg. A PlanetScope műholdfelvételek a Planet Labs Inc. *Education and Research Program*-ja keretében kerültek a kutatáshoz biztosításra [9].

A. függelék

Szimulációs eredmények

A.1. Felszíni vizek változásdetektálása

Következőkben bemutatom a folyamat tudományos eredményeit. Itt csak a végeredmények validációja lesz bemutatva. Részeredmények validációja a témában született TDK dolgozatban található.

A változásdetektálás tesztelése a Sentinel-2 34UEU és 34TET csempéin lett elvégezve a 2018. április 20-a és 2018. augusztus 20-ai felvételekkel. Ezeken vizsgálva lettek az NDWIG és az mNDWI indexszel készült vízmaszkok Yen, Otsu és Minimum küszöböléssel létrehozott verziói. Ezek validálására felhasználtuk a 34UEU csempe és a Lechner Tudásközpont által biztosított adatok átfedését.

A folyamat során gondot okozott a felhőmaszkolás és küszöbölő algoritmusok egymást befolyásolása. A felhőmaszkolás után futtatott Yen küszöbölés több felvételnél is hibát okozott, az algoritmus nullával való osztásba futott, így itt nem született eredmény. Ez elsősorban az NDWIG-ből származó vízmaszkok létrehozásakor fordult elő, a mNDWI-ből számoltaknál nem okozott gondot. A Minimum küszöbölés viszont pont az utóbbi esetben hibázott, ugyanis a felhőmaszkolt mNDWI indexeknél a határértéket nem tudta jól kiszámolni, és üres vízmaszkokat adott meg eredményként.

A legjobb eredményeket NDWIG küszöbnél a Minimum algoritmus adta, az így készült vízmaszkokon talált különbségek 92,6%-ban egyeztek meg a Lechner Tudásközpont belvíz térképeinek különbségeivel. Ezeknél nagyobb hibát az okozott, hogy a mi vízmaszkjainkban eléggé eltért az állandó vizek (pl.: Tisza) értéke, hogy a változásdetektálásban ezek belekerültek a maszkba. Továbbá néhány szántóföld

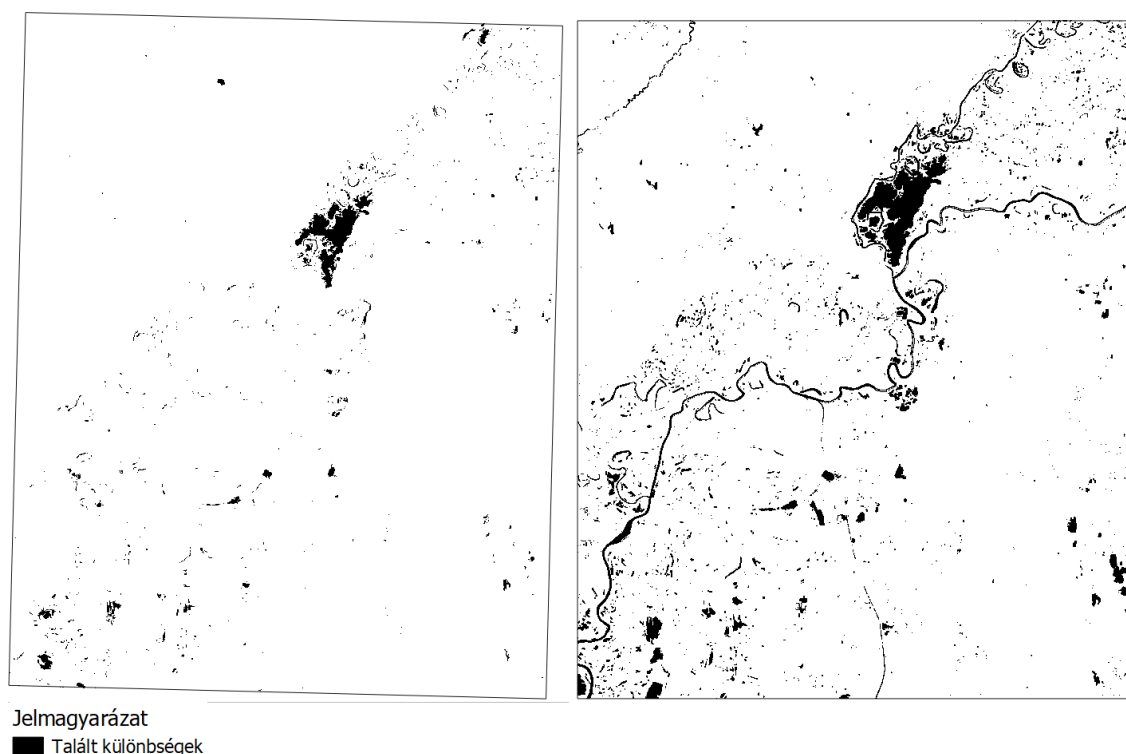
adott fals pozitív találatot, ahol a belvizes időszakban még a termények látszódtak így magasabb volt az NDWI_g értékük. A száraz időszakban viszont már learatták őket, és már nem kerültek bele az ekkor készült maszkba, ezzel változást okozva a végső eredményünkben. Az otsu küszöbölt vízmaszkoknál ezek a hibák még relevánsabbak voltak, és így csak 86,9%-os volt az egyezés.

Az mNDWI-s vízmaszkokra legjobb eredménnyel a Yen küszöbölés szolgált. Itt a megegyezés a szakértői adatokkal 95,3% volt. Ez a módszer jobban kiszűrte mind az állandó vizeket, mind a szántóföldeket. A havas és vizes területek meghatározása pedig a kiválasztott időszak miatt nem okozott gondot. Az Otsu küszöbölt vízmaszkok különbségei itt nagyon eltértek a validáló adatoktól, csak 8,9%-os egyezés volt közöttük.

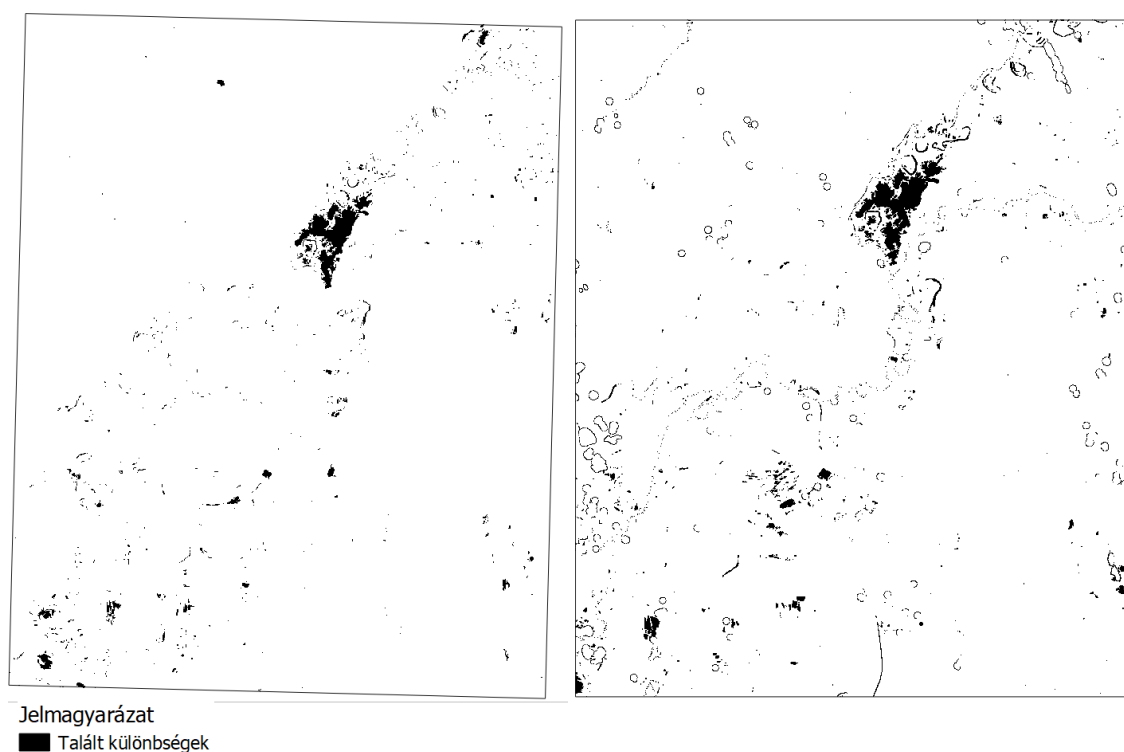
Eredmények összefoglalása a A.1 táblázatban látható.

NDWI _g	Minimum	92,6%
	Otsu	86,9%
mNDWI	Yen	95,3%
	Otsu	8,9%

A.1. táblázat. Eredmények összefoglalása



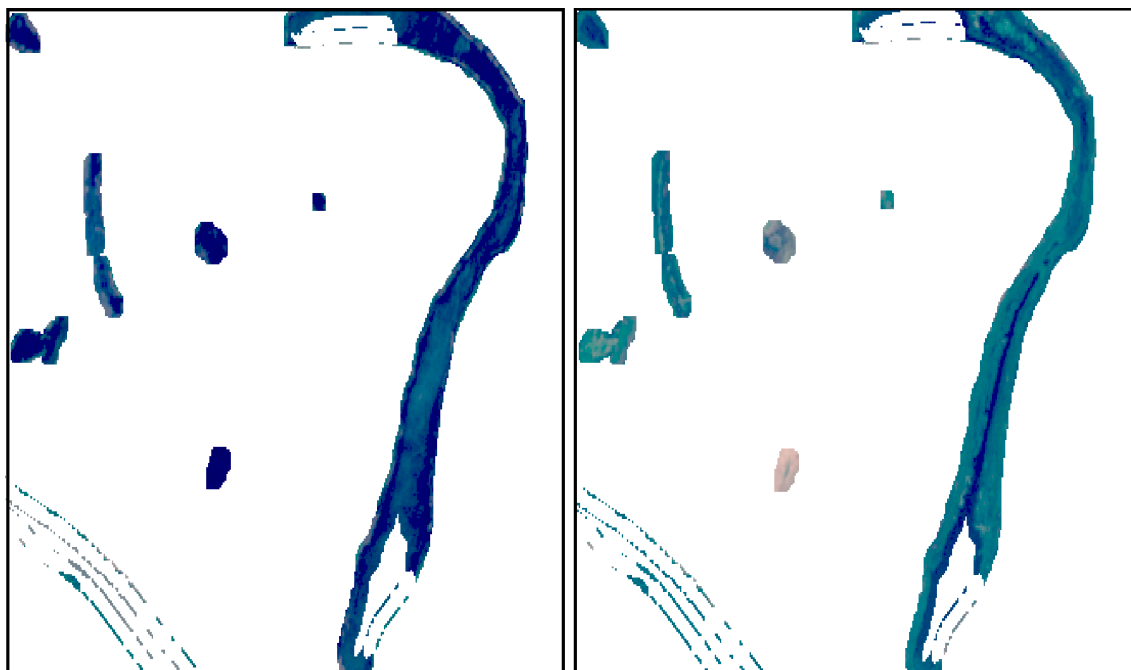
A.1. ábra. A Lechner Tudásközpont vízmaszkjainak és az NDWI_g Minimum küszöböléssel készült vízmaszkjainak különbségei



A.2. ábra. A Lechner Tudásközpont vízmaszkjainak és az mNDWI Yen küszöböléssel készült vízmaszkjainak különbségei

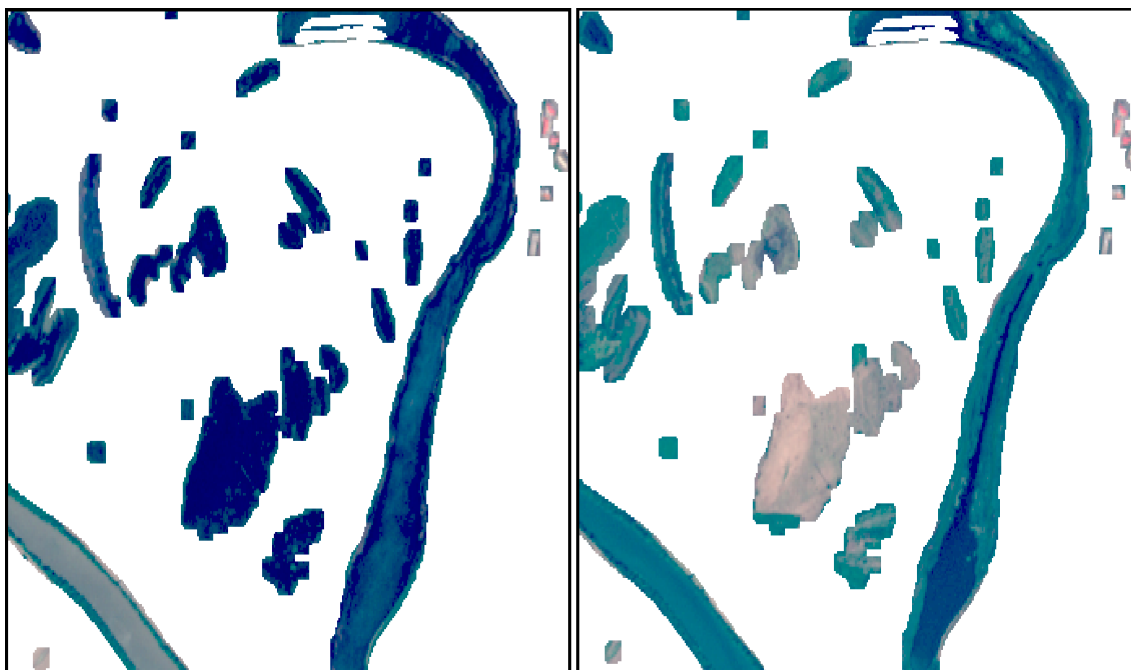
Következő képeken látható a különböző végeredmények alkalmazása színes Sentinel-2 felvétel területének egy részére. Először az mNDWI és Yen küszöb párosítás, második az NDWIg és Minimum, utolsó pedig az NDWIg és Otsu. Ezt a kivágott területet úgy választottam ki, hogy lehetőleg jobban illusztrálja az egyes módszerek erősségeit és gyengéit.

Következő képen A.3 látható, hogy alacsony küszöbérték miatt nem került bele ezen a területen hibás adat, ki lett szűrve bal lent a Tisza, jobb felső részen nem kerültek bele lakóházak. Szántóföldek sem kerültek bele egyáltalán. Középső részen viszont adatvesztést eredményezett, kimaradt több vízzel feltelt terület is.



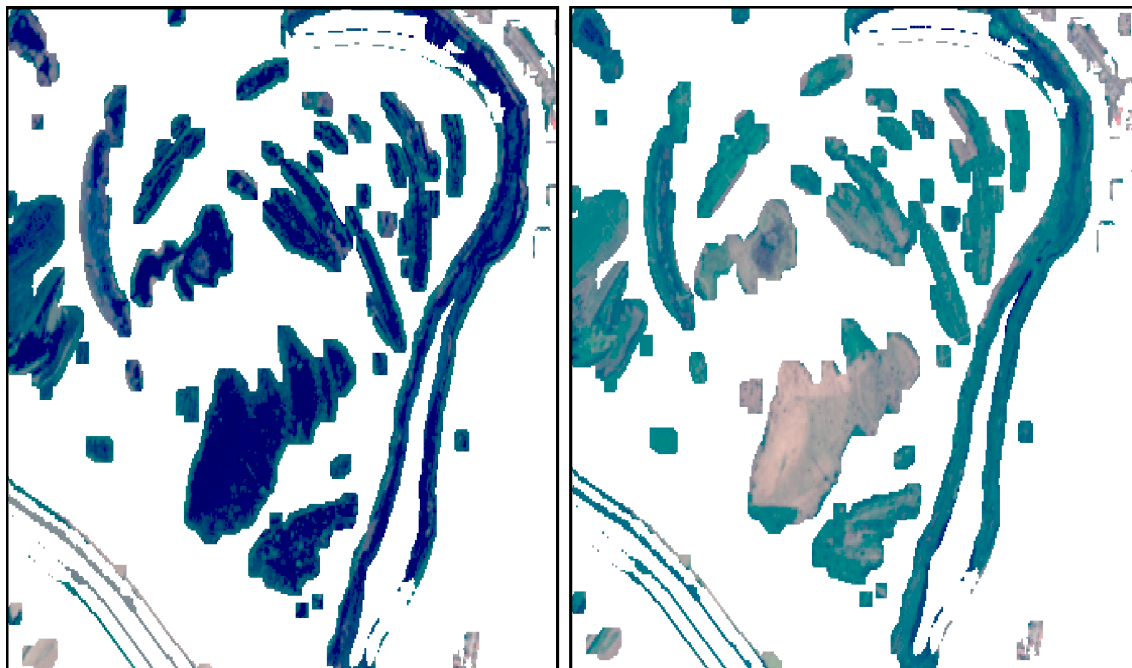
A.3. ábra. 2018.04.20-ai (bal) és 2018.08.20-ai (jobb) 34UEU csempe különbség maszkja mNDWI index Yen küszöböléssel.

Itt A.4 már magasabb a küszöbérték, és ez azzal a hibával jár, hogy a jobb felső területen belekerült az eredménybe néhány lakóház. Középső területen, viszont már látható több vízzel telítődő terület. Másik hibaforrás, hogy a kép bal alján láthatóvá vált a Tisza.



A.4. ábra. 2018.04.20-ai (bal) és 2018.08.20-ai (jobb) 34UEU csempe különbség maszkja NDWIg index Minimum küszöböléssel.

Utolsó képen A.5 látható módszer eredményezte a legnagyobb küszöbértéket, itt látható a legtöbb belvíz, viszont ezek körül már több nem ide tartozó terület is megjelenik. A Tiszát, mint állandó folyóvizet itt sikerült kiszűrni. Az alacsony küszöb miatt itt megjelenik néhány szántóföld (bal alsó részen), valamint jobb fent egy egész utca belekerült eredményünkbe.



A.5. ábra. 2018.04.20-ai (bal) és 2018.08.20-ai (jobb) 34UEU csempe különbség maszkja NDWIG index Otsu küszöböléssel.

Irodalomjegyzék

- [1] S. K. McFEETERS. „The use of the Normalized Difference Water Index (NDWI) in the delineation of open water features”. *International Journal of Remote Sensing* 17.7 (1996), 1425–1432. old.
- [2] Hanqiu Xu. „Modification of normalised difference water index (NDWI) to enhance open water features in remotely sensed imagery”. *International Journal of Remote Sensing* 27.14 (2006), 3025–3033. old.
- [3] Zhe Zhu és Curtis E. Woodcock. „Object-based cloud and cloud shadow detection in Landsat imagery”. *Remote Sensing of Environment* 118 (2012), 83–94. old.
- [4] Zhe Zhu, Shixiong Wang és Curtis Woodcock. „Improvement and expansion of the Fmask algorithm: Cloud, cloud shadow, and snow detection for Landsats 4-7, 8, and Sentinel 2 images”. *Remote Sensing of Environment* 159 (2015), 269–277. old.
- [5] Otsu és Nobuyuki. „A Threshold Selection Method from Gray-Level Histograms”. *IEEE Transactions on Systems, Man, and Cybernetics* 9.1 (1979), 62–66. old.
- [6] Judith M. S. Prewitt és Mortimer L. Mendelsohn. „The analysis of cell images”. *Annals of the New York Academy of Sciences* 128.3 (1966), 1035–1053. old.
- [7] Jui-Cheng Yen, Fu-Juay Chang és Shyang Chang. „A new criterion for automatic multilevel thresholding”. *IEEE Transactions on Image Processing* 4.3 (1995), 370–378. old.
- [8] Zhou Wang és tsai. „Image Quality Assessment: From Error Visibility to Structural Similarity”. *IEEE Transactions on Image Processing* 13.4 (2004), 600–612. ISSN: 1057-7149.

- [9] Planet Team. *Planet Application Program Interface: In Space for Life on Earth*. <https://api.planet.com/>. San Francisco, CA. 2017.

Ábrák jegyzéke

2.1. Az elektromágneses sugárzás spektruma (nem méretarányos) és felhasználása a műholdas távérzékelésben.	4
2.2. Elkészült vízmaszkok összehasonlítása	7
2.3. A GUI kezdőképernyője	10
2.4. A főablak menüje kezdőállapotban, megnyitott mappák nélkül	11
2.5. A GUI egy képpel a bal keretben	11
2.6. A folyamatok külön-külön erről a felületről végezhetőek el.	12
2.7. A GUI maszkolás menüje	13
2.8. A maszkolás menü folyamat futása közben	13
2.9. A GUI egy maszkolt és egy tiszta Sentinel-2 képpel megjelenítve . . .	14
2.10. A folyamat beállításait tartalmazó ablak	15
2.11. A folyamat futás közben	16
3.1. Az MVC architektúra ábrázolása	18
3.2. A folyamat flowchart-ja	19
3.3. A GuiView osztály használati esetei	21
3.4. A ProcessingView osztály használati esetei	22
3.5. Az OptionsView osztály használati esetei	23
3.6. A szoftver UML diagramja	24
A.1. A Lechner Tudásközpont vízmaszkjainak és az NDWIg Minimum küszöböléssel készült vízmaszkjainak különbségei	34
A.2. A Lechner Tudásközpont vízmaszkjainak és az mNDWI Yen küszöböléssel készült vízmaszkjainak különbségei	35
A.3. 2018.04.20-ai (bal) és 2018.08.20-ai (jobb) 34UEU csempe különbség maszkja mNDWI index Yen küszöböléssel.	36
A.4. 2018.04.20-ai (bal) és 2018.08.20-ai (jobb) 34UEU csempe különbség maszkja NDWIg index Minimum küszöböléssel.	36

A.5. 2018.04.20-ai (bal) és 2018.08.20-ai (jobb) 34UEU csempe különbség maszkja NDWlg index Otsu küszöböléssel.	37
--	----

Táblázatok jegyzéke

2.1. Sentinel-2 műhold sávjai	5
2.2. A szoftver gépigénye	8
3.1. A felhasznált indexek kiszámításának képletei.	20
3.2. A GuiView funkcióinak tesztelése	27
3.3. A ProcessingView funkcióinak tesztelése	28
3.4. Az OptionsView funkcióinak tesztelése	29
A.1. Eredmények összefoglalása	34

Forráskódjegyzék

2.1. Anaconda virtuális környezet létrehozása, aktiválása	8
2.2. Csomagok telepítése pip-pel	9
2.3. Conda Forge létrehozása	9
2.4. Conda Forge csomagok telepítése	9
2.5. Az alkalmazás futtatásához szükséges parancs	9