



**ЕЛЕКТРОТЕХНИЧКИ ФАКУЛТЕТ
УНИВЕРЗИТЕТ У БАЊОЈ ЛУЦИ**

ДИГИТАЛНА ОБРАДА СЛИКЕ

Студенти: Ана Кркљић 80/08

Маја Поповић 21/08

Игор Томић 14/08

Предметни наставник: др Зденка Бабић
Предметни асистент: мр Владимир Рисојевић

Бања Лука
Јул 2012. године

КЛАСИФИКАЦИЈА САТЕЛИТСКИХ СНИМАКА

НЕНАДГЛЕДАНА КЛАСИФИКАЦИЈА

1. ДЕФИНИСАЊЕ ПРОБЛЕМА

Задатак нашег пројекта јесте да користећи сателитске слике територије која обухвата дио Босне и Херцеговине и Хрватске снимљене у више видљивих и инфрацрвених електромагнетних опсега, прикажемо неколико покривача земљишта на том простору. Пиксели слике се групишу према сличности својих спектралних карактеристика (боја, нијанса боје, ниво сивила и слично) аутоматски (ненадгледана класификација), према неком алгоритму, а додјелу типова покривача земљишта (информационих класа) тим групама (спектралним класама) обавља корисник.

2. РЕАЛИЗАЦИЈА ПРОЈЕКТА

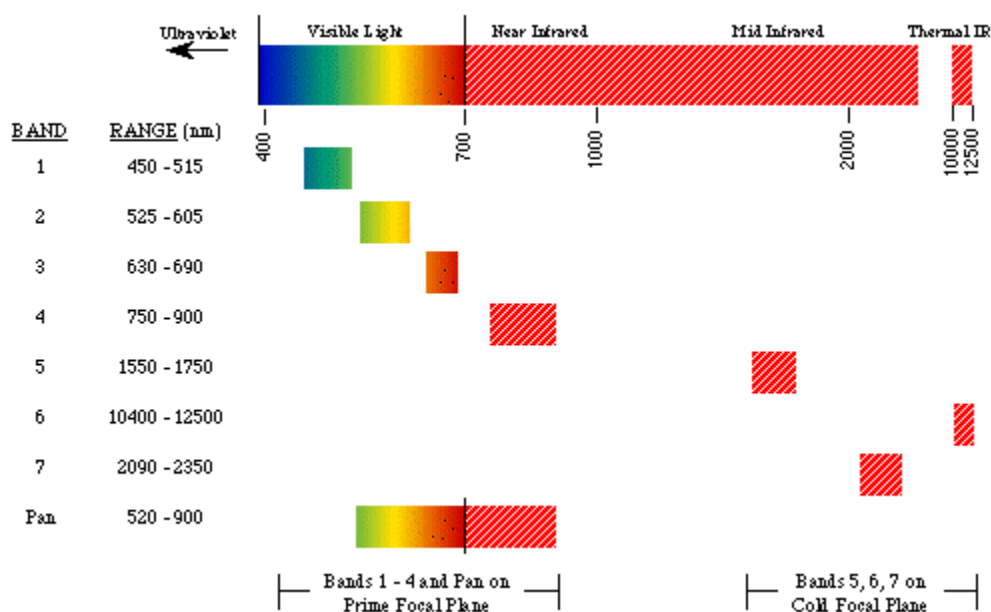
ОБЈЕКТИ ОБРАДЕ

На располагању су снимци сателита Landsat 7 снимљени сензорима ЕТМ+. Радиометријске карактеристике снимака дате су у Табели 1. На Слици 1 су визуелно приказани опсези таласних дужина наведени у табели.

Satellite	Spectral Resolution (μm)	Band	Spatial Resolution
Landsat 7	ETM+		
	Band 1: 0.450 – 0.515	Blue	30
	Band 2: 0.525 – 0.605	Green	30
	Band 3: 0.630 – 0.690	Red	30
	Band 4: 0.760 – 0.900	Near IR	30
	Band 5: 1.550 – 1.750	Mid IR	30
	Band 6†: 10.40 – 12.5	Thermal	60
	Band 7: 2.080 – 2.35	Mid IR	30
	Band 8: 0.52 – 0.92	Pan	15

† Band 6 on Landsat 7 is divided into two bands, high and low gain.

Табела 1. Радиометријске карактеристике



Сл.1. Електромагнетни опсези слика

У овом пројекту смо користили слике из видљивог опсега (Blue, Red и Green), инфрацрвеног опсега MidIR (Band 5) и опсега Pan (панхроматска слика) који обухвата зелени, црвени и дио инфрацрвеног опсега. Прве четири слике смо комбиновали у једну мултиспектралну слику на следећи начин (у Matlab-у):

```

RGBIC(:, :, 1) = Red;
RGBIC(:, :, 2) = Green;
RGBIC(:, :, 3) = Blue;
RGBIC(:, :, 4) = MidIR , а панхроматску слику смо обрађивали као интензитетску.
  
```

Све ове слике приказују дио територије Босне и Херцеговине и Хрватске. Због сложености алгоритама за обраду, користили смо само дијелове слике која обухвата околину неког града.

КЛАСИФИКАЦИЈА САТЕЛИТСКИХ СЛИКА

Класификацијом слике се сваком пикселу те слике придружује категорија из предефинисаног скупа. Резултат класификације је тематска мапа оригиналне слике. У нашем случају се класификација врши на основу спектралних информација које су представљене интензитетом пиксела у различитим спектралним опсезима. Скупови пиксела који имају блиске вриједности представљају спектралне класе. За корисника су од интереса информационе класе. У нашем случају, то су типови покривача земљишта. Одлучили смо се за приказ пет типова покривача: водене површине, травнате површине, шуме, голети и површине измјењене људским дјеловањем (насеља, путеви, њиве, копови...).

Наш задатак јесте ненадгледана класификација слика, односно да се неким алгоритмом аутоматски формирају спектралне класе, а затим интерактивно додјељујемо тим класама информационе класе. За ненадгледану класификацију смо користили метод K-means.

K-MEANS МЕТОД

Кластеризација је распоређивање објеката у групе сличних објеката. Те групе се називају кластери. Кластеризација се врши на основу интензитета пиксела у спектралним опсезима.

Први корак K-means метода јесте одређивање броја кластера. Чест је случај да пиксели једне спектралне класе припадају различитим информационим класама те је потребно подијелити ту спектралну класу на више класа. Дакле, потребно је усвојити већи број кластера. Ако се деси да пиксели двије спектралне класе припадају једној информационом класи, потребно је објема класама придружити исту информациону класу. Видимо да је за успостављање односа између спектралних и информационих класа потребна људска интервенција.

Поред усвајања броја кластера, потребно је одредити и почетне центроиде, центре кластера. За вријеме процеса кластеризовања, рачуна се удаљеност сваког пиксела од свих центроида према предефинисаној метрици (нпр. Еуклидово растојање). Пиксел се придружује кластеру са најближим центроидом. Када се сви пиксели слике распореде у кластере, врши се помјерање центроида тако да је збир удаљености од свих пиксела његовог кластера минималан. Затим се поступак понавља са новим центроидима све док разлика удаљености центроида у двије итерације не буде мања од неког предефинисаног броја. Што је тај број мањи, број итерација је већи. Број итерација се може ограничити, али у нашем пројекту нисмо користили овај начин. Када се испуни наведени услов, свим пикселима оригиналне слике се придружују вриједности центроида кластера којима припадају. На овај начин добијамо слику са онолико региона колико имамо кластера.

Класификацију слике смо вршили помоћу функције `klasteri`:

`funkcija [novasl C] = klasteri(GBIC,k)` врши класификацију слике GBIC у k региона

Argumenti функције су:

- GBIC, слика која има двије просторне димензије и четри спектралне компоненте (R,G,B и IC, мада могу бити и било које друге)
- k је број кластера, природан број

Izlazi из функције су:

- novasl, слика са k региона, има двије просторне и четри спектралне компоненте, вриједности слике одговарају вриједностима центроида
- C, вектор центроида димензије $k \times 4$

Пошто слика која је вишедимензионална пролази кроз више итерација и у свакој итерацији се врши неколико операција над сваким пикселем, како би се смањило вријеме потребно за извршење програма, улазна слика се претвара у тип `single` и остатак програма ради са овим типом података:

```
slika=im2single(GBIC);% функција ради са single
```

Програм је и даље сложен па бирамо улазне слике мањих просторних димензија.

Како је на појединим регионима слике велика варијанса вриједности пиксела (посебно у градским подручјима), уједначавамо њихове вриједности помоћу униформног филтра како на излазној слици не бисмо добили на малом региону пуно различитих пиксела:

```
slika=imfilter(slika,fspecial('average',2),'symmetric');
```

Почетни центроиди се одређују на основу највећих вриједности интензитета пиксела у спектралним опсезима дате слике и на основу броја кластера. Када бисмо имали три спектралне класе (нпр. R,G и B) и вриједности представили у тродимензионалном простору, највеће вриједности интензитета по класама би формирале тјемена над којима се може конструисати квадар, а почетни центроиди би били равномјерно распоређени по дијагонали квадра. Дио кода који рачуна почетне центроиде је:

```
Mr=max(max(slika(:, :, 1)))/(k+1);
Mg=max(max(slika(:, :, 2)))/(k+1);
Mb=max(max(slika(:, :, 3)))/(k+1);
Mic=max(max(slika(:, :, 4)))/(k+1);
s=size(slika);
Cnovo=zeros(k, 4, 'single');
for r=1:k % petlja koja prolazi kroz klase
    Cnovo(r, :)= [Mr*r, Mg*r, Mb*r, Mic*r]; %pocetni centroidi klasa
end
```

У главном дијелу кода се рачуна Еуклидово растојање пиксела слике од свих центроида:

```
ro(1)=sqrt((C(1,1)-slika(i, j, 1))^2+(C(1,2)-slika(i, j, 2))^2+(C(1,3)-slika(i, j, 3))^2+(C(1,4)-slika(i, j, 4))^2);
```

Одређује се најближи центроид и пиксел се смјешта у кластер коме тај центроид припада:

```
[dmin indeks]=min(ro); % dmin=najmanja udaljenost piksela od svih centroida
                        % indeks=klasa od koje je najmanja udaljenost
klase(indeks, br(indeks), :)=slika(i, j, :); % u odgovarajucu klasu smjesta
rgb_piksel
novasl(i, j, :)=C(indeks, :);
br(indeks)=br(indeks)+1; %broji elemente jedne klase
```

Када се сви пиксели распореде у кластере, рачунају се нови центроиди:

```
for r=1:k
    for j=1:4
        Cnovo(r, j)=sum(klase(r, :, j))/(br(r)-1+1e-5); % komponente novog
        centroida su aritmeticka sredina komponanti clanova klase
    end
end
```

Провјерава се да ли се нови центроид помјерио у односу на центроид у претходној итерацији за мање од 0.008 по свим спектралним координатама, и ако је овај услов испуњен за све центроиде, излази се из петље:

```
uslov=im2single(zeros(k,1));
while (sum(sum(uslov))<k*4)

    ...

uslov=(abs(Cnovo-C)<0.008);
end % kraj while-a
```

Формирана је слика са k региона. Вриједности пиксела одговарају вриједностима центроида кластера којима припадају. На овој слици се могу појавити усамљени пиксели који немају информациону вриједност. То је пиксел једне боје који је потпуно окружен пикселима друге боје (других боја). Уклањамо их функцијом `cisti`

```
%=====POSTOBRADA-uklanjanje usamljenih piksela=====
novasl=cisti(novasl);
%=====
```

Функција провјерава да ли се пиксел разликује од свих осам сусједних пиксела и ако се разликује, додјељује му вриједност сусједног лијевог пиксела:

```
function [sl] = cisti(sl)
s=size(sl);
for i=2:s(1)-1
    for j=2:s(2)-1
        brojac=0;
        for m=-1:1
            for n= -1:1
                if (sl(i,j,:)==sl(i+m,j+n,:))
                    brojac=brojac+1;
                end
            end
        end
        if brojac==1
            sl(i,j,:)=sl(i,j-1,:);
        end
    end
end
end
```

Овим се излази из функције `klasteri`.

Излазни аргументи из функције `klasteri` (`novasl` и `C`) се заједно са улазном сликом (RGBIC) прослеђују функцији `boje` која нема излазне параметре већ даје двије слике. На једној слици су приказане оригинална слика (слика коју класификујемо, RGBIC) и класификована слика гдје су различитим бојама приказане информационе класе (типови покривача земљишта), а друга слика представља легенду (типови покривача земљишта).

Ова функција може да ради са највише 7 кластера. Предефинисано је седам боја:

```
CRVENA=[1 0 0 0];
ZELENA=[0 0.2 0 0];
ZUTA=[1 1 0 0];
SVJZELENA=[0 0.7 0 0];
ROZA=[0.7 0.3 0.4 0];
CRNA=[0 0 0 0];
PLAVA=[0 0 1 0];
BIJELA=[1 1 1 0];
```

Пошто смо одлучили да приказујемо највише пет типова покривача земљишта, формирамо легенду са пет елемената, нпр:

```
p1(:, :, 1)=ones(10,20)*PLAVA(1); p1(:, :, 2)=ones(10,20)*PLAVA(2);
p1(:, :, 3)=ones(10,20)*PLAVA(3);
```

```
...
figure, subplot(515), imshow(pl), xlabel('vodene površine')
```

Затим се свим пикселима који су унутар једне класе даје једна од предефинисаних боја, нпр:

```
if ((sl(i,j,1)==C(1,1)) && (sl(i,j,2)==C(1,2)) && (sl(i,j,3)==C(1,3)) &&
    (sl(i,j,4)==C(1,4)))
    sl(i,j,:)=ZELENA;
end
```

Када први пут позивамо функцију *boje*, свакој класи додјељујемо посебну боју. На излазној слици обично боја не одговара својој природној информационој класи, на примјер, море је представљено црном бојом. Зато у едитору функције мијењамо боје које смо додијелили центроидима и поново позивамо функцију. Ако више спектралних класа припада једној информационој, једноставно тим класама додјељујемо исту боју.

Пошто се наша верзија **kmeans** алгоритма, код **klasteri.m**, извршава веома споро, те отежава тестирање и модификацију, направили смо код **obrni.m** који користи уграђену МАТЛАБ-ову функцију **kmeans**. Алгоритам кода **obrni.m** је следећи:

- Улазни параметри су слика и број спектралних класа.
- Учитану слику најприје претвори у слику типа **single**, због уштеде меморијског простора. Такође, при свакој алокацији празних низова и матрица, наводи се **'single'** јер је **double** подразумевани тип за функције **zeros**, **ones**...

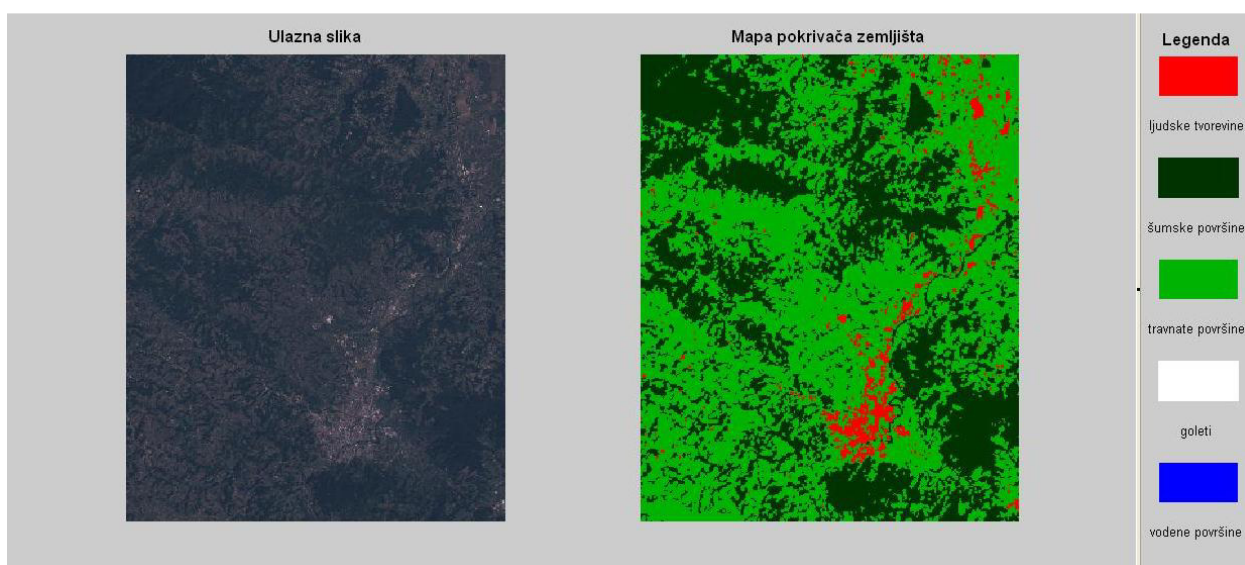
```
prazna=zeros(s, 'single');
```
- Слику затим филтрира униформним кернелом 2x2 да је уједначи
- Пресложи пикселе онако како то захтијева **kmeans**. Користи функцију **reshape**.
- Од добијеног вектора лабела те матрице центроида направи поново структуру од 4 матрице (као што је слика RGBIC).
- Уради постобраду у виду уклањања усамљених пиксела, што је реализовано у функцији **cisti.m**
- На крају функције налази се звучни сигнал, који обавјештава корисника да је процесирање готово. Ово се показало корисно у случају већих слика односно дуготрајнијег процесирања, јер корисник тада не мора проверавати знак **busy** у доњем лијевом углу МАТЛАБ прозора.

3.ТЕСТИРАЊЕ ПРЕДЛОЖЕНОГ РЈЕШЕЊА

Наводимо неколико примјера резултата који се добију употребом кода **obrni.m** те кода **boje.m**.

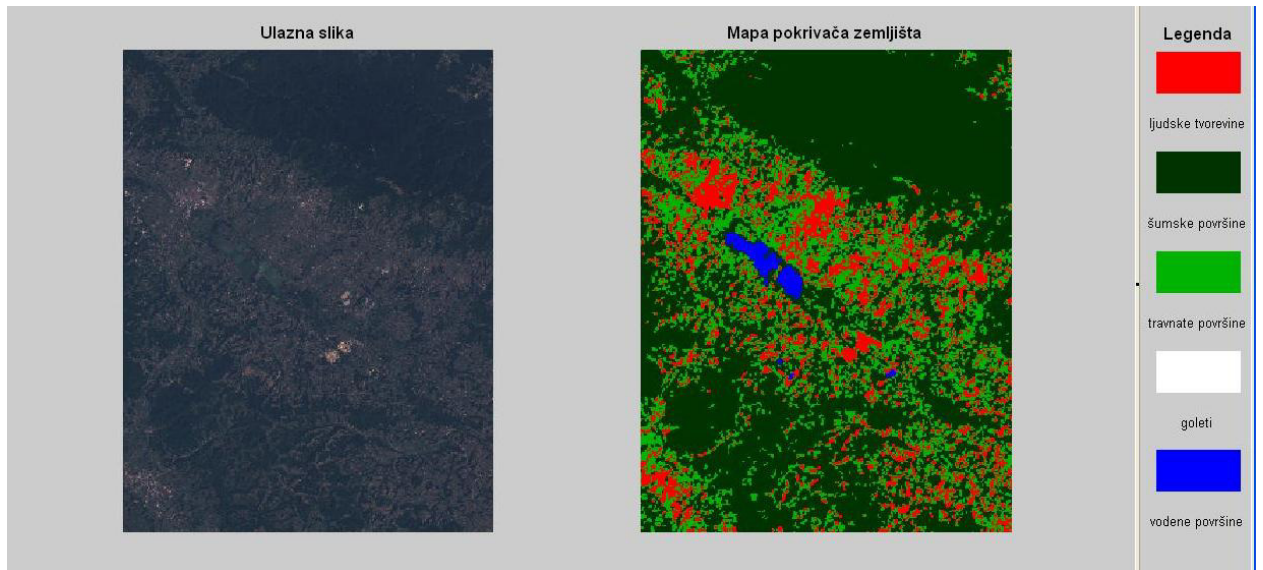
Зог ограничења у погледу ресурса нисмо сегментирали цијелу добијену слику која приказује дио БиХ и Хрватске, него неколико мањих области:

- Бањалука са околином,
- дио Козаре, Приједор, Сански мост
- Карловац
- Задар са околином



Слика 1. Бањалука са околином

Приказана је сегментирана слика са 3 информационе класе, те легенда која показује врсте покривача земљишта које је могуће издвојити. Овај резултат је добијен позивањем функције **obrni.m** са 5 спектралних класа након чега су, посматрањем добијене мапе те употребом интерфејса **boje.m** неке спектралне класе спојене. Уочавају се области покривене шумом, травнате површине те људске творевине, које обухватају урбану зону те друге облике површина обрађених „људском руком“. Сматрамо да се са ове удаљености не види Врбас, јер би то сувише танка линија. Такође, његово корито је заклоњено дрвећем те се може уочити линија тамно зелене боје, која представља шумско растиње.



Слика 2. дио Козаре, Приједор, Сански мост

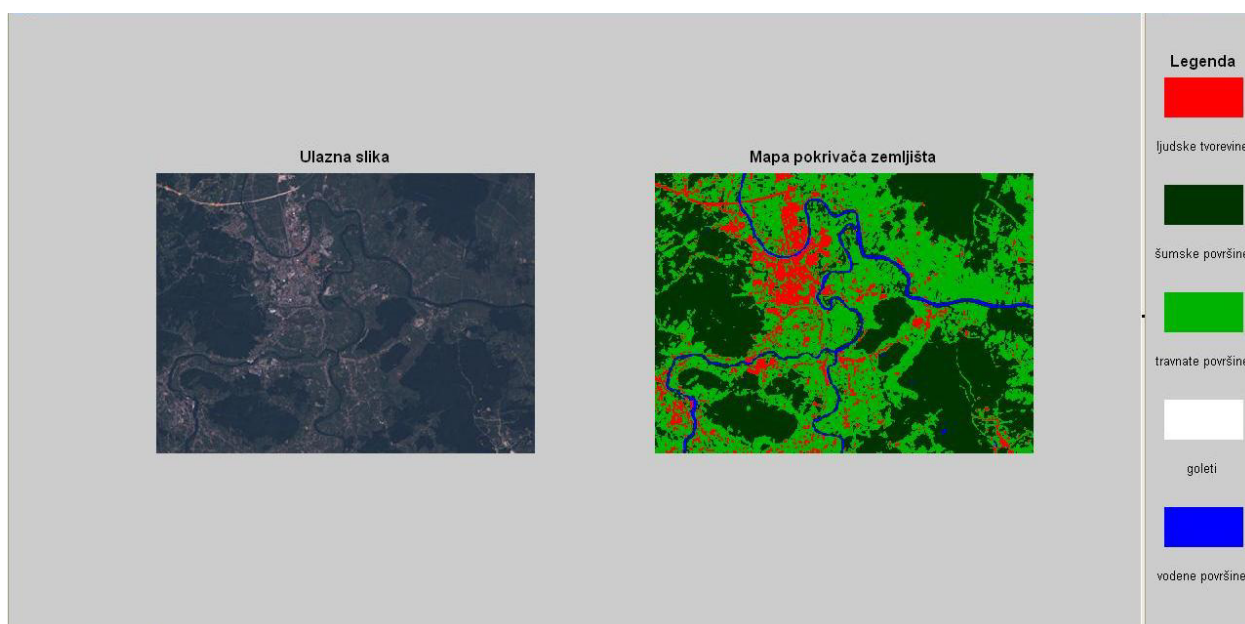
Овдје је издвојена и класа водене површине. Ријеч је о рибњаку у близини Приједора. Такође је вршена сегментација са више спектралних класа, те обједињавање неких од њих. Ово је било неопходно, јер би у случају мање спектарлних класа рибњак био сврстан у исту класу као шуме Козаре. У класи људских творевина, осим Приједора, Санског Моста те других мањих насеља уочавамо и рудник жељеза Омарска.



Слика 3. Рудник жељеза „Омарска1“

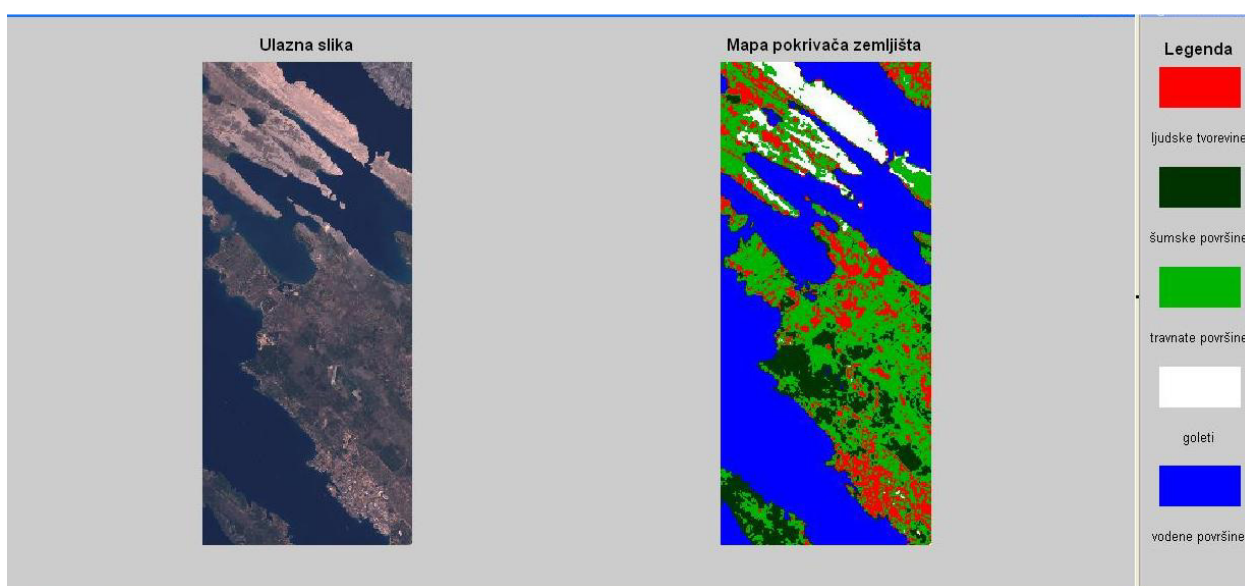


Слика 4. Рибњак „Саничани“ код Приједора



Слика 5. Карловац

Ово је примјер сегментације која препознаје ријечне токове.



Слика 6. Задар са околином

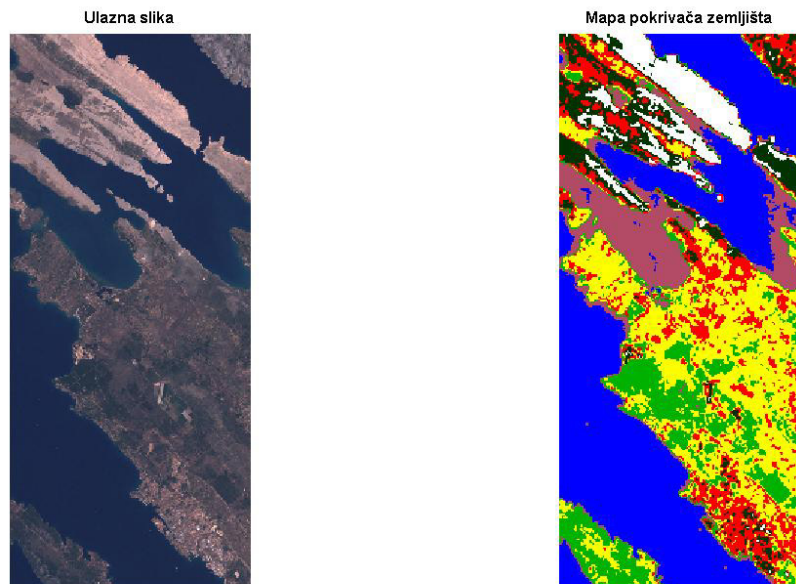
Поред људских творевина, шумских, травнатих те водених површина овдје уочавамо и голети на острву Паг. Следећа слика приказује терен о ком је ријеч.



Слика 7. Голет на острву Паг

Поступак добијања описаних резултата, корак по корак ,на примјеру околине Задра:

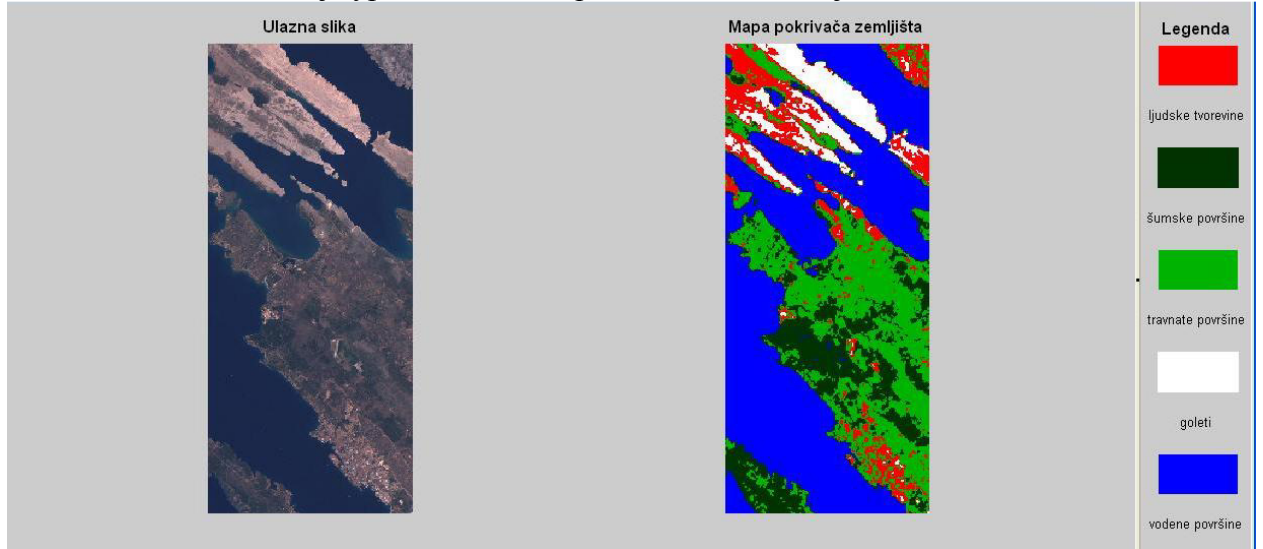
Слика коју видимо лијево се прво сегментира помоћу **obrni.m** при чему се претпостави 7 спектралних класа. У коду воје.m за почетак се понуђене боје се насумично додијеле класама,при чему се употрејеби 7 различитих боја.



Слика 8. Задар са околином,сегментације са 7 класа

Корисник на добијеној мапи уочи да се јављају двије различите класе за дио слике који представља море, па у коду воје.m замијени РОЗА са ПЛАВА. Тиме споји ове двије класе.Слично томе, жуту боју замијени свјетлозеленом, те обрне свјетлозелену и зелену (то је ова најтамнија боја на слици). Тако се добије резултат са слике 4.

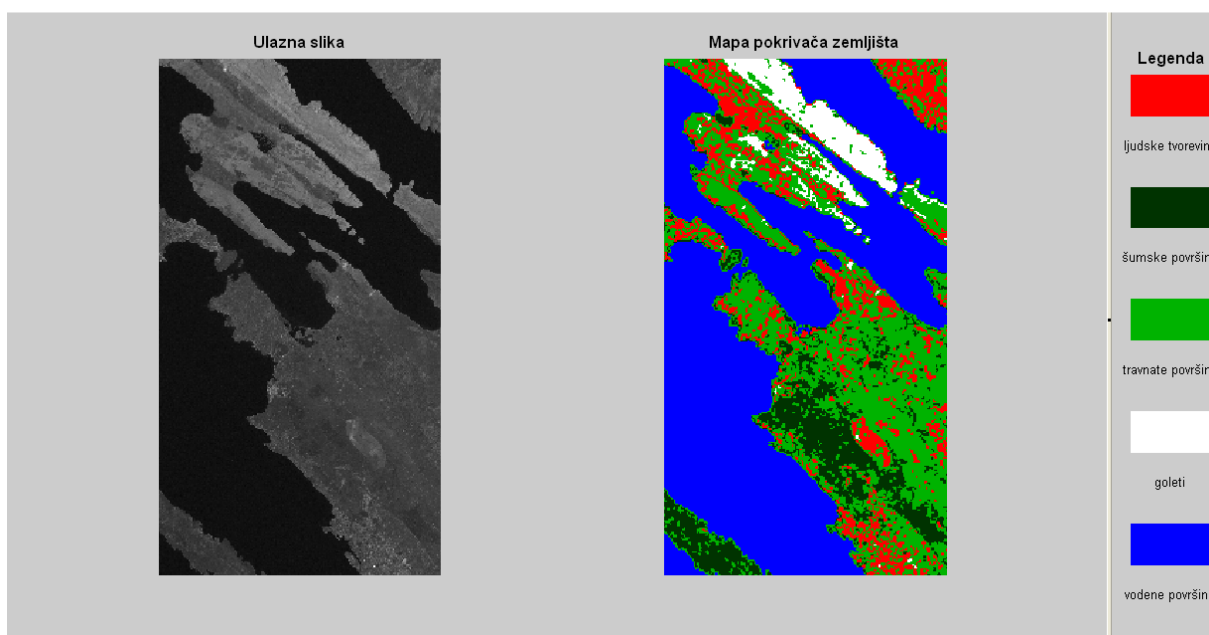
Уколико се сегментација уради са 5 спектралних класа, добије се следеће:



Слика 9. Задар са околином, сегментације са 5 класа

Уочимо да на овој слици град Задар изгледа прилично „распарчан“, док се на острвима јавља пуно црвене боје, дакле та област изгледа доста урбанија него што заправо јесте. Такође, у граду се уочавају дијелови који припадају класи голети, што не би имало смисла.

Због ове, условно речено, погрешне сегментације, која се јавља на примјеру Задра те рибњака код Приједора, одлучили смо се на горе описани приступ, сегментацијом са више спектралних класа, него што је број очекиваних информационих класа.



Слика 10. Сегментација панхроматске слике

На овој слици приказана је класификација панхроматске слике. Користе се функције `panbrni.m`, `panboje.m` и `cisti.m`. Класификација је урађена са седам класа, а затим смо вршили спајање спектралних класа како бисмо добили одговарајуће информационе. Видимо да резултати нису добри као код вишеспектралне слике: површине измјењене људским дјеловањем се појављују тамо гдје у стварности не постоје, а градска зона Задра је распарчана. Дакле, како бисмо добили бољу класификацију, потребно је користити више спектралних опсега.