



UNIVERSITATEA NAȚIONALĂ DE  
ȘTIINȚĂ ȘI TEHNOLOGIE  
POLITEHNICA BUCUREȘTI



Școala Doctorală de Electronică, Telecomunicații  
și Tehnologia Informației

Decizie nr. 82 din 19.07.2024

# REZUMAT TEZĂ DE DOCTORAT

**Ing. Alexandru-Toma ANDREI**

---

METODE INTELIGENTE DE IDENTIFICARE A  
DETALIILOR TERESTRE PE BAZA ANALIZEI  
IMAGINILOR AERIENE MULTISPECTRALE

---

## COMISIA DE DOCTORAT

<b>Prof. Dr. Ing. Ion MARGHESCU</b> Univ. Națională de Știință și Tehnologie POLITEHNICA București	Președinte
<b>Prof. Dr. Ing. Ovidiu GRIGORE</b> Univ. Națională de Știință și Tehnologie POLITEHNICA București	Conducător de doctorat
<b>Prof. Dr. Ing. Radu Gabriel BOZOMITU</b> Univ. Tehnică „Gheorghe Asachi” din Iași	Referent
<b>Prof. Dr. Ing. Radu-Viorel RĂDESCU</b> Univ. Națională de Știință și Tehnologie POLITEHNICA București	Referent
<b>Conf. Dr. Ing. Alin DĂNIȘOR</b> Univ. Maritimă din Constanța	Referent

**BUCUREȘTI 2024**

---

# Cuprins

Cuprins .....	ii
Capitolul 1 Introducere .....	1
1.1    Prezentarea domeniului tezei de doctorat .....	1
1.2    Scopul tezei de doctorat .....	1
Capitolul 2 Stadiul actual al metodelor inteligente aplicate imaginilor multispectrale	2
2.1    Stadiul actual al tehnicilor de clustering .....	3
2.2    Stadiul actual al rețelelor de învățare profundă convoluționale .....	3
Capitolul 3 Baza de date .....	4
3.1    Descrierea sistemului fotogrammetric .....	4
3.2    Culegerea și prelucrarea datelor brute .....	5
3.3    Obținerea bazei de date finale .....	5
Capitolul 4 Studiul algoritmilor nesupervizați pentru identificarea detaliilor terestre	6
4.1    K-Means .....	7
4.2    Clusteringul ierarhic aglomerativ .....	8
4.3    Modelul de amestec gaussian .....	10
4.4    Mean Shift .....	12
4.5    Concluzii .....	13
Capitolul 5 CEM – Modul de extragere a culorii .....	14
Capitolul 6 Îmbunătățirea algoritmului GMM prin tehnici de compresie și procesare a semnalelor .....	18
6.2    Evaluarea rezultatelor metodologiei propuse .....	20
6.3    Concluzii .....	21
Capitolul 7 Învățarea supervizată prin arhitectura U-Net .....	22
7.1    Etichetarea automată a datelor .....	22
7.2    Descrierea arhitecturii U-Net .....	23
7.3    Optimizarea hiperparametrilor nonstructurali .....	23
7.4    Simplificarea modelului U-Net .....	24
7.5    Concluzii .....	26
Capitolul 8 Concluzii .....	27
8.2    Contribuții originale .....	27
8.3    Lista lucrărilor originale .....	28

8.3.1	Articole publicate în jurnale științifice internaționale .....	28
8.3.2	Lucrări în volume de conferințe științifice internaționale indexate WOS. .....	29
8.3.3	Lucrări în volume de conferințe științifice internaționale indexate BDI29	
8.3.4	Rapoarte de cercetare științifică.....	29
Bibliografie .....		30

# Capitolul 1

## Introducere

### *1.1 Prezentarea domeniului tezei de doctorat*

Din punct de vedere istoric, defrișarea a început acum mult timp, în epoca Holocenului [6]. De când oamenii au descoperit focul, au demarat un proces lent, dar sigur, de defrișare la nivel mondial. La început, lemnul era folosit pentru nevoi de bază, cum ar fi încălzirea peșterilor sau gătitul după vânătoare. Aceste activități nu constituiau o problemă pentru pădure, care putea să se regenereze după tăierea arborilor, însă oamenii au început să folosească lemnul pentru construirea de case, întregi orașe sau ziduri medievale, procesând lemnul cu ferăstraie mecanice acționate prin curgerea apei și instalate de-a lungul râurilor. La sfârșitul epocii medievale, pădurea a fost afectată de alte două elemente: inventarea motorului cu abur și creșterea necesității agriculturii [7]. Se estimează că aproape jumătate din pădurea inițială a Pământului a fost pierdută [8]. Doar în ultimul secol, 10% din pădurile lumii au fost defrișate pentru a face loc culturilor sau pășunatului, apogeul fiind în anii '80. În acea decadă, totalul zonelor defrișate au fost egale cu dimensiunea Mongoliei de astăzi [9].

În ultimii ani, diferite organizații sau activiști au început să lupte pentru a salva pădurea și a opri defrișarea necontrolată. În prezent, societatea a început să realizeze importanța pădurilor și efectele toxice ale defrișării. Asociații internaționale precum Uniunea Europeană sau organizații neguvernamentale reglementează această activitate, pentru a ajunge la un echilibru optim între tăierea copacilor și regenerarea pădurii. Dar vechile obiceiuri mor greu, iar abuzurile sau sfidarea legilor sunt încă practicate. Pe lângă întreaga industrie cunoscută și legiferată a lemnului, defrișările ilegale sunt o altă problemă majoră. Prin aplicarea metodelor inteligente cercetate și dezvoltate în această lucrare pentru identificarea de păduri, se oferă un ajutor substanțial activiștilor de mediu și combaterii defrișărilor.

### *1.2 Scopul tezei de doctorat*

Această cercetare își propune să găsească o metodă globală cu costuri reduse de implementare și utilizare pentru detectarea defrișărilor. Se dorește ca algoritmul de segmentare automată final să fie o unealtă ușor de utilizat pentru orice activist sau operator pentru a identifica defrișările. Este foarte important să obținem rezultate rapide în problema defrișării, având în vedere capacitățile moderne de tăiere, transport și prelucrare a copacilor. Astfel, obiectivul se rezumă la identificarea

pădurilor din imaginile aeriene multispectrale, iar exemplificările și analizele s-au realizat pentru clasa de păduri.

Pe lângă toate acestea, un impediment al studiului și comun pentru modelele de inteligență artificială (AI) îl reprezintă necesitatea de resurse. Majoritatea implementărilor în condiții reale ale modelelor AI sunt realizate în condiții dificile sau cu capacități hardware foarte reduse, fără a menționa necesitatea de analiză a datelor și predicție rapide. Un alt obiectiv major al lucrării a fost reducerea de costuri. În cazul monitorizării defrișărilor, acesta este un obiectiv obligatoriu. Din păcate, în cazul defrișărilor ilegale, zone vaste de păduri pot dispărea într-o singură noapte, deci și metoda de combatere trebuie să fie una rapidă. De asemenea, această metodă ar trebui să fie prietenoasă cu mediul și să funcționeze cu resurse reduse de energie. O altă problemă a fenomenului de defrișare este reprezentată de modul de aplicare a legii sau altor reglementări. În acest moment, singura metodă de a depista defrișările în timp real este patrularea activiștilor voluntari a zonelor de interes cu mașina și apelul acestora la organele competente în caz de activitate suspectă. Având în vedere toate cele de mai sus, folosirea unui avion de mari dimensiuni pentru înregistrarea de date este indiscutabilă și nepractică. Pentru a obține rezultate optime, supravegherea zonelor de interes trebuie realizată cu platforme aeriene de mici dimensiuni, cum ar fi o dronă, care survolează zonele de interes și raportează în timp util la un centru de comandă sau la autorități competente. Acest lucru ar rezolva și problema reținerii infractorilor sau cel puțin înregistrării de probe împotriva acestora.

## Capitolul 2

# Stadiul actual al metodelor inteligente aplicate imaginilor multispectrale

În ultimii ani, cererea pentru interpretarea imaginilor a crescut, și odată cu aceasta, și tehnicile de învățare automată. În prezent, majoritatea lucrărilor de clasificare a imaginilor se concentrează pe rețele de tip CNN aplicate datelor open-source deja etichetate. Majoritatea tehnicilor de augmentare sau îmbunătățire sunt legate de antrenarea rețelei neuronale și reglarea parametrilor acesteia, în timp ce metodele de clustering au scăzut în popularitate. Cu toate acestea, fiecare dintre cele două tipuri de învățare automată are avantajele și dezavantajele ei. De exemplu, învățarea supervizată poate fi mai precisă, dar are nevoie de date etichetate, ceea ce poate consuma resurse. Pe de altă parte, învățarea nesupervizată nu necesită date etichetate, dar poate fi dificil să găsești similarități sau corelații între pixelii unei imagini aeriene

și are în general o precizie mai mică. Pentru a înțelege mai bine aceste caracteristici a fost necesară analiza stadiului actual și al lucrărilor conexe tezei de doctorat pentru fiecare dintre cele două abordări.

## ***2.1 Stadiul actual al tehnicilor de clustering***

Chiar dacă există și alte metode cum ar fi SVM (Support Vector Machines) sau autoencodere, învățarea nesupervizată este reprezentată în majoritatea cazurilor de tehnicile de clustering, deoarece de obicei se urmărește identificarea de grupuri cu trăsături asemănătoare într-un set de date, cum este și în cazul nostru. Metodele de clustering sunt diverse, de la cele mai simple până la cele mai complexe. Cea mai comună și simplă de implementat tehnică de clustering este algoritmul K-Means. Acesta nu consumă mult timp și are un rezultat general bun pentru că se pliază pe majoritatea tipurilor de date. De cele mai multe ori acest algoritm este implementat prin diferite tehnici de augmentare sau preprocesare pentru particularitățile fiecărui studiu [10-11].

O altă tehnică de clustering a cărei concepte matematice se regăsesc în contribuțiile lui Carl Friedrich Gauss din începutul anilor 1800 este algoritmul GMM. Chiar dacă Gauss este renumit pentru dezvoltarea distribuției normale, care servește ca bază fundamentală pentru clusteringul GMM, prima mențiune explicită apare în lucrarea lui Karl Pearson din 1894 [13]. Metoda a început să câștige recunoaștere largă în domeniul învățării automate în anii 1970. Un factor semnificativ care a contribuit la popularitatea lor a fost apariția algoritmului EM (Expectation Maximization), o metodă robustă pentru potrivirea eficientă a GMM-urilor la date, prezentată de Arthur P. Dempster, Nan M. Laird și Donald B. Rubin [14]. De atunci, diferiți cercetători au depus eforturi constante pentru a aplica și dezvolta metoda în domeniul clasificării imaginilor [15-17].

## ***2.2 Stadiul actual al rețelelor de învățare profundă convoluționale***

Învățarea profundă este constituită din mai multe ramuri, însă cele mai eficiente în sarcini de recunoaștere de imagini sunt rețelele neuronale convoluționale (CNN). Prima noțiune a CNN a fost introdusă la sfârșitul anilor 1980 de către Yann LeCun și colegii săi [21], dar punctul de cotitură a fost în 2011 când a fost prezentat AlexNet [22], model care a reprezentat nașterea rețelelor neuronale profunde. În afara celor opt straturi, AlexNet a folosit tehnici care par obișnuite astăzi, cum ar fi funcția de activare ReLU (Rectified Linear Unit) și regularizarea de tip dropout, tehnici cu un impact major asupra acurateții rețelei. După aceasta, s-a creat o infinitate de posibilități în ceea ce privește arhitectura unei rețele CNN, însă doar unele dintre aceste s-au evidențiat prin inovație. În 2015, Olaf Ronneberger, Philipp Fischer și Thomas Brox au prezentat arhitectura U-Net [25], o rețea neuronală convoluțională

inovatoare și simetrică orientată spre clasificarea pixelilor. Scopul inițial al studiului a fost segmentarea imaginilor biomedicale, dar de-a lungul timpului U-Net a fost adaptată pentru utilizare într-o multitudine de sarcini de segmentare semantică, cum ar fi scrierea manuală, observațiile medicale, automatizarea industrială sau segmentarea imaginilor satelitare, domeniu în care și monitorizarea defrișărilor se încadrează. Un alt motiv pentru alegerea U-Net în locul altor arhitecturi este abilitatea sa de a funcționa cu mai puține imagini și cu o etichetare slabă.

## Capitolul 3

### Baza de date

Față de alte domenii unde imaginile se pot înregistra chiar și cu telefonul mobil sau o cameră profesională, culegerea imaginilor aeriene implică resurse și costuri mult mai ridicate. Echipamentele cum ar fi sistemul fotogrammetric sau stațiile de procesare sunt scumpe, iar realizarea și planificarea zborurilor sunt activități complexe. Din această cauză majoritatea studiilor și cercetărilor folosesc baze de date din surse deschise puse la dispoziție de către instituții guvernamentale, universități sau comunități online. Chiar dacă sunt gratuite, majoritatea bazelor de date cu imagini aeriene sau satelitare au o rezoluție radiometrică destul de slabă și sunt culese și eventual etichetate pentru anumite scopuri prestabilite, astfel plierea lor pe alte obiective putând fi greoaie. Cu toate acestea, această teză de doctorat folosește imagini aeriene proprii, furnizate de Agenția de Informații Geospațiale a Apărării în scopul cercetării.

#### *3.1 Descrierea sistemului fotogrammetric*

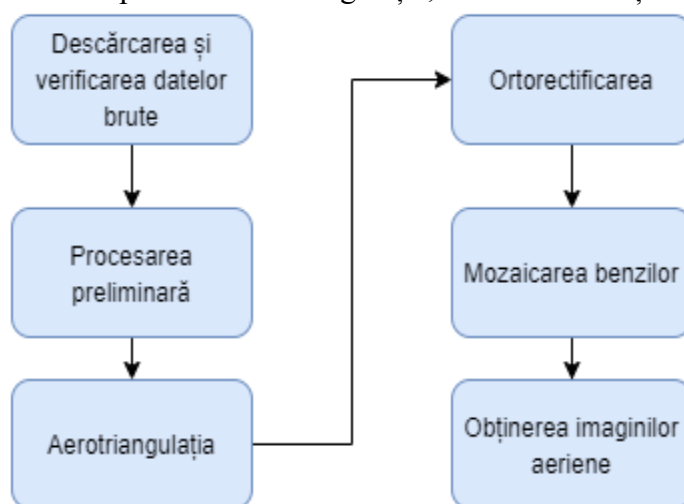
Imaginile aeriene au fost colectate cu ajutorul unui sistem fotogrammetric digital Leica Geosystems ADS80 (Airborne Digital Sensor 80) de tip pushbroom aparținând Agenției de Informații Geospațiale a Apărării. Sistemul a fost instalat pe un avion de mici dimensiuni Antonov An-30, model special proiectat pentru scopuri de aerofotografiere. Pe lângă elementele auxiliare care se regăsesc la bordul avionului, sistemul fotogrammetric este format din senzorul de înregistrare, unitatea de comandă și control, memoria de stocare a înregistrărilor și aplicațiile software utilizate pentru obținerea produsului final. Având în vedere toate acestea, imaginile aeriene realizate au fost descrise de următoarele tipuri de rezoluții și caracteristici:

- rezoluția spațială de 50 cm care reprezintă dimensiunea unui pixel la teren;
- rezoluția spectrală de patru benzi R, G, B și NIR, adică intervalul de lungime de undă în limitele căruia s-a realizat înregistrarea unei imagini;

- rezoluția radiometrică de 8 biți care se transpune în 256 de valori gri posibile pentru înregistrarea răspunsului radiometric de către senzor.

### 3.2 Culegerea și prelucrarea datelor brute

Informațiile culese de sistemul fotogrammetric sunt date brute, imposibil de folosit pentru algoritmi de învățare automată. Acestea au trecut printr-un proces complex de prelucrare pentru a se obține imagini aeriene multispectrale utilizabile. Întreg fluxul de lucru este ilustrat în **Figura 3.3**. După proiectarea și executarea zborului fotogrammetric, datele brute au fost descărcate și verificate vizual pentru identificarea de nori sau păsări care ar putea periclita claritatea din anumite zone. După o procesare preliminară, au urmat etapele de aerotriangulație, ortorectificare și mozaicare.

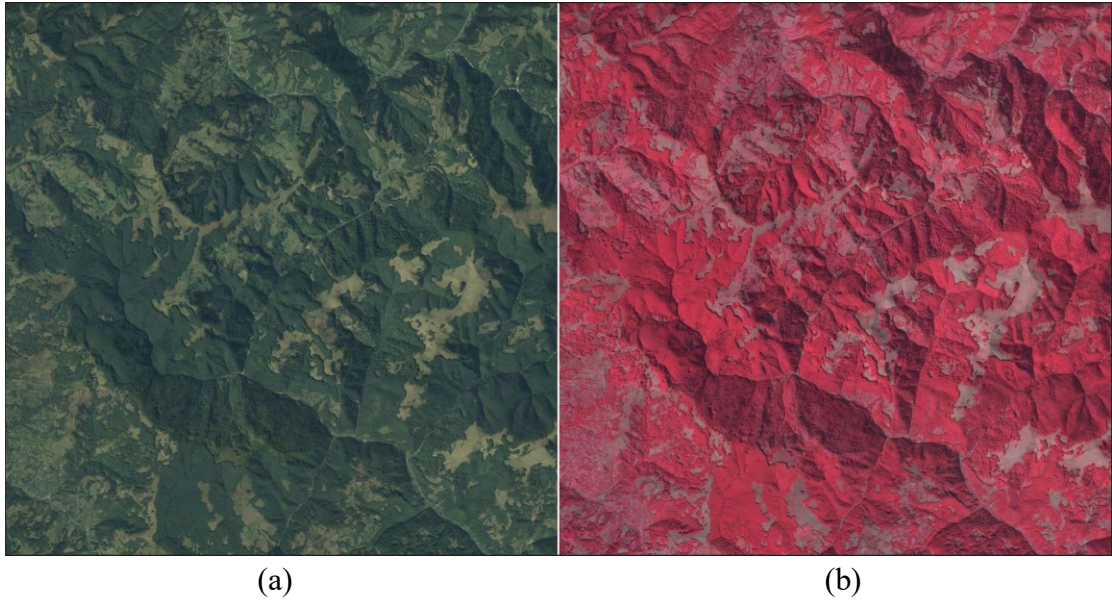


**Figura 3.3** Etapele prelucrării datelor brute pentru obținerea imaginilor aeriene

### 3.3 Obținerea bazei de date finale

Având în vedere interesele instituției, datele prelucrate au fost exportate în beneficiul obținerii a două tipuri de produse: imagini aeriene RGB și CIR (Color InfraRed). De asemenea, aria de zbor a fost secționată în zone de  $10 \times 10 \text{ km}^2$  și salvate sub formă de fișiere TIFF (Tagged Image File Format). Pentru sprijinirea cercetării acestei teze de doctorat, AIGA a pus la dispoziție opt astfel de zone, patru din apropierea orașului Târgu-Mureș și patru din apropierea comunei Roșia Montană, zone bogate în păduri, dar și amestecate cu alte clase de acoperire a terenului, cum ar fi așezări umane, culturi agricole, pășuni sau drumuri. În **Figura 3.5** se poate vizualiza un exemplu de imagine aeriană de  $100 \text{ km}^2$  sub formă RGB și CIR.





**Figura 3.5** Mostre ale imaginilor aeriene; (a) imagine RGB; (b) imagine CIR

Pentru eliminarea acestei redundanțe și păstrarea doar a informațiilor relevante, banda de infraroșu apropiat a fost extrasă din imaginile CIR și concatenată imaginilor RGB, rezultând imagini RGBN. În final, cele trei seturi de date RGBN, RGB și NIR au fost secționare în trei dimensiuni de  $4000 \times 4000$ ,  $1000 \times 1000$  și  $200 \times 200$  pentru a avea diversitate înaintea aplicării algoritmilor de învățare. Acest lucru a generat un set de date generos, care a eliminat necesitatea de augmentare folosită de multe alte studii din domeniu.

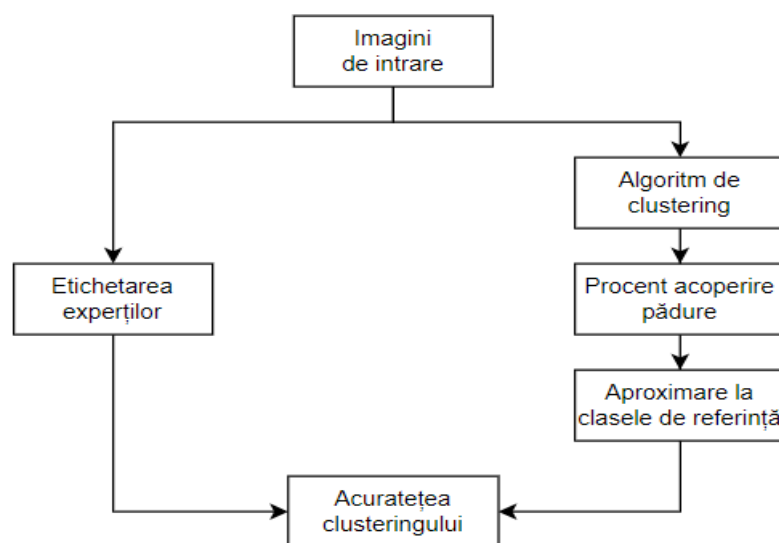
## Capitolul 4

# Studiul algoritmilor nesupervizați pentru identificarea detaliilor terestre

Ideea principală pentru atingerea obiectivelor asumate a fost următoarea: pentru a putea dezvolta o soluție flexibilă care poate ingera imagini aeriene culese în diferite condiții și pentru a ocoli nevoia de etichetare a acestora, se poate folosi o tehnică de clustering al cărui rezultat să servească la antrenarea unei rețele supervizate sau care poate fi îmbunătățită și folosită autonom. În acest mod se scurtează și timpul de predicție prin eliminarea etapei de fotointerpretare, iar algoritmul dezvoltat nu se va supraantrena pe un anumit set de date. Pentru a avea o etichetare cât mai performantă, patru algoritmi de clustering au fost selecționați pentru aplicarea asupra imaginilor aeriene: K-Means, AGNES (Agglomerative Nesting), GMM și Mean shift. La

sfârșitul capitolului s-a concluzionat care dintre aceste patru tehnici a avut cele mai bune rezultate și a fost aleasă ca bază pentru dezvoltările ulterioare.

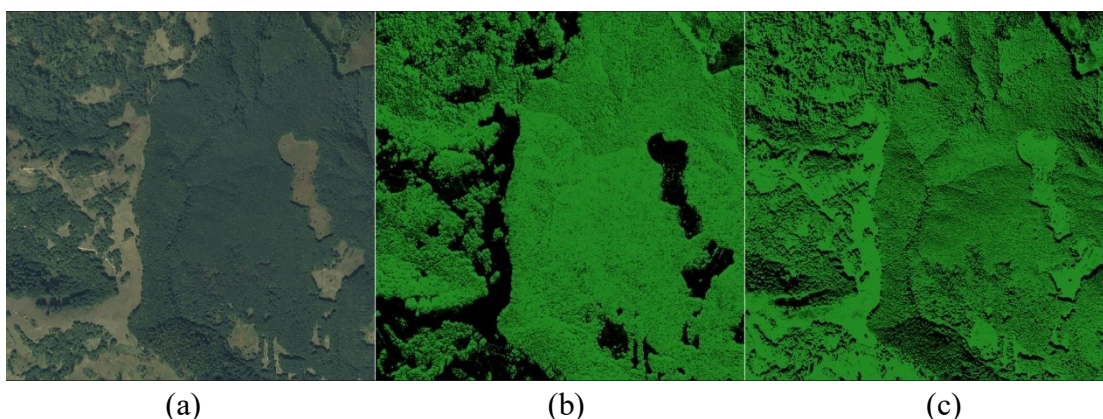
Pentru a realiza o analiză manuală a conținutului imaginilor prelucrate, s-a folosit o echipă de experți GIS specializați în fotointerpretare și fluxul de lucru ilustrat de **Figura 4.1**. Aceștia au stabilit cinci clase reprezentând procentul zonei de pădure posibil existente într-o imagine, în felul următor: imagine acoperită complet de pădure (100%), acoperire mare (75%), medie (50%), mică (25%) și imagine fără pădure (0%). Apoi, fiecare imagine utilizată ca dată de intrare a fost catalogată în una dintre aceste clase. Ulterior, pentru fiecare imagine segmentată s-a calculat cantitatea de pădure identificată de algoritmi de clustering și s-a rotunjit la cea mai apropiată clasă din cele cinci. Având în vedere că numărul total de pixeli este cunoscut, acest lucru a putut fi realizat printr-o regulă de trei simplă, după numărarea tuturor pixelilor care conțin pădure. La final, acuratețea a fost calculată prin compararea celor două etichete, eticheta imaginii de intrare stabilită de experți și eticheta calculată prin numărare.



*Figura 4.1* Diagrama de evaluare a acurateții algoritmilor de clustering

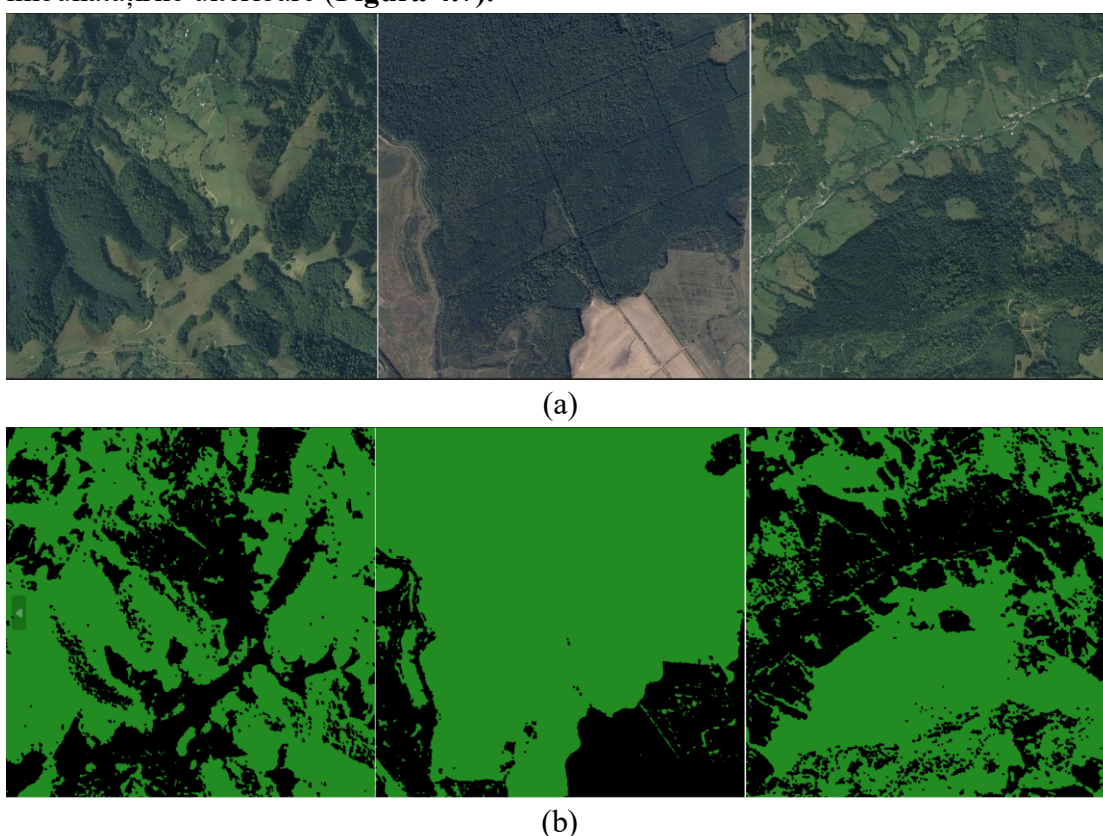
## 4.1 K-Means

Clusteringul K-Means a fost aplicat atât imaginilor RGB cât și RGBN. În mod surprinzător, imaginile RGB s-au comportat foarte bine fără nicio preprocesare, iar introducerea benzii suplimentare de NIR a avut rezultate mai slabe. În **Figura 4.5** este evidențiată această situație. Presupunerea pentru acest lucru este că banda de infraroșu apropiat conține valori ridicate pentru toată vegetația sănătoasă din teren, nu doar pentru păduri, iar algoritmul de clustering nu a putut distinge acest aspect în timpul calculului de distanțe, considerând, de exemplu, și pășunea ca zonă acoperită de pădure.



**Figura 4.5** Rezultatele clusteringului *K-Means*; (a) realitatea din teren; (b) imagine RGB segmentată; (c) imagine RGBN segmentată

Au fost folosite 200 de imagini aeriene RGB de dimensiune  $4000 \times 4000$ . Acestea au fost etichetate de experții GIS în cele cinci clase stabilite și în urma clusteringului s-a calculat o acuratețe de 85.56%. Acest scor a fost considerat satisfăcător pentru început și un bun punct de plecare pentru implementările și îmbunătățirile ulterioare (**Figura 4.7**).



**Figura 4.7** Exemple ale clusteringului *K-Means*; (a) imagini de intrare; (b) imagini segmentate

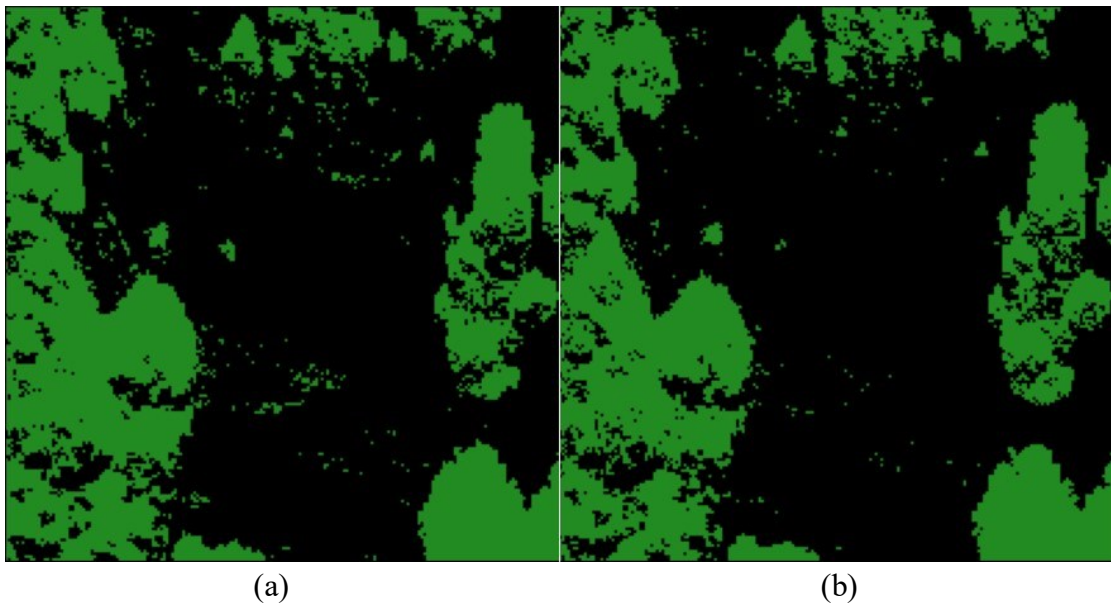
## 4.2 Clusteringul ierarhic aglomerativ

Din motive de resurse ale sistemului de calcul folosit, dimensiunea maximă la care s-a putut aplica algoritmul ierarhic AGNES asupra celor seturi de date NIR, RGB și

RGBN a fost de  $200 \times 200$ . Au fost folosite metricile L1, L2 și cosinus și criteriile de legătură singulară, completă, medie și metoda lui Ward. Singurele scenarii care au avut rezultate satisfăcătoare au fost criteriul de legătură completă folosind metricile L1 și L2 și metoda Ward. Legătura completă a avut rezultate mai bune folosind toate cele patru benzi spectrale, în timp ce metoda lui Ward a avut rezultate mai bune folosind setul de date RGB. Au fost folosite 200 de imagini aeriene de dimensiune  $200 \times 200$ , iar acuratețea finală pentru fiecare scenariu a fost:

- legătură completă, norma L1, RGBN: 58.33%;
- legătură completă, norma L2, RGBN: 73.33%;
- metoda lui Ward, norma L2, RGB: 87.78%.

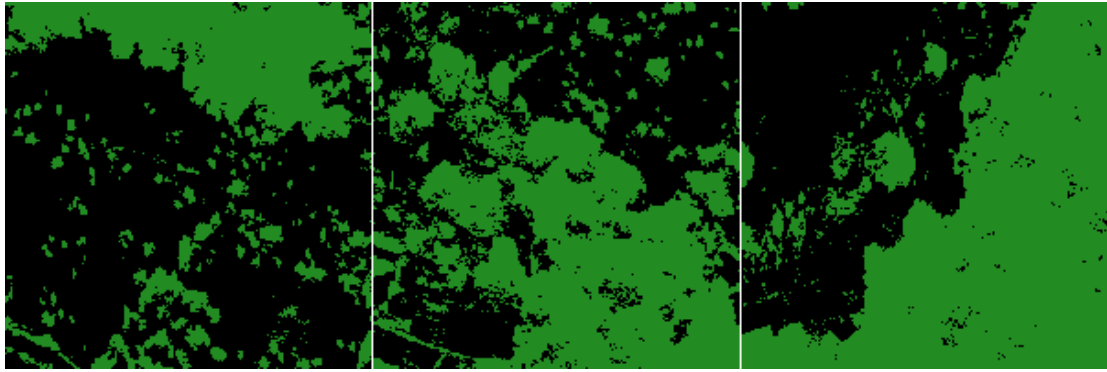
Comparând rezultatele metodei lui Ward pentru setul de date RGB (exemple în **Figura 4.12**) cu rezultatele legăturii complete folosind norma L2 pentru setul de date RGBN, s-a observat o îmbunătățire evidentă. Imaginea este mai puțin zgomotoasă, iar marginile pădurii sunt mult mai bine definite (**Figura 4.11**). În **Anexa 2** pot fi consultate mai multe rezultate în formă mărită ale metodei lui Ward aplicată imaginilor RGB.



**Figura 4.11** Rezultatele comparative ale normei L2; (a) criteriul de legătură completă; (b) metoda lui Ward



(a)

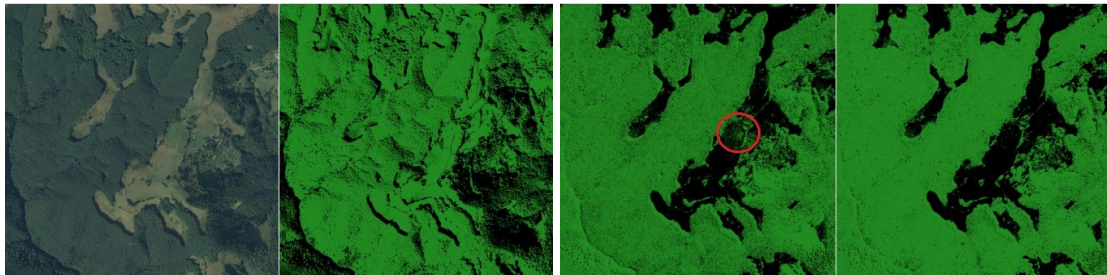


(b)

**Figura 4.12** Mostre ale clusteringului AGNES; (a) imagini de intrare; (b) imagini segmentate

### 4.3 Modelul de amestec gaussian

În studiile anterioare, unde s-au implementat algoritmi K-Means și AGNES, rezultatele setului de date RGB s-au dovedit a fi mai bune decât rezultatele setului RGBN. În cazul GMM, rezultatele au fost diferite, aplicarea GMM benzii NIR a avut un rezultat previzibil, nefiind posibilă construirea unei mixturi cu atât de puține informații. Rezultatele celorlalte două seturi de date, la o primă vedere, au fost similare, dar imaginile RGBN au avut o performanță ușor mai bună.



(a)

(b)

(c)

(d)

**Figura 4.15** Rezultatele GMM pentru imagini cu adâncimi diferite; (a) realitatea din teren; (b) NIR; (c) RGB; (d) RGBN

Un alt factor important al oricărui algoritm de învățare automată, îl reprezintă dimensiunea imaginilor de intrare sau fereastra de analiză pe baza căreia sunt făcute calculațiile. S-a început testarea cu o imagine inițială de  $20000 \times 20000$ , iar ulterior am redus dimensiunea de intrare la zone mai mici, respectiv  $4000 \times 4000$ ,  $1000 \times 1000$  și  $200 \times 200$ . Odată cu analizarea rezultatelor, s-a măsurat și timpul de execuție. Acesta este prezentat în **Tabelul 4.1** folosind secunde ca unitate de măsură. Numărul de intrări reprezintă câte imagini de ordin inferior compun o imagine de ordin superior. Pentru aceeași suprafață și aceleași date de intrare, comparațiile trebuie făcute pe diagonală. Luând în considerare toate acestea, cele mai bune două dimensiuni de intrare au fost alese:  $4000 \times 4000$  și  $1000 \times 1000$ .

**Tabelul 4.1** Timpul de execuție al GMM pentru dimensiuni de intrare diferite

<b>Dimensiune</b>	20000	4000	1000	200
<b>Nr. intrări</b>	-	25	16	25
<b>Timp mediu (s)</b>	17745.2750	151.7057	11.1756	0.6246
<b>Timp total (s)</b>	-	3792.6430	178.8102	15.6150

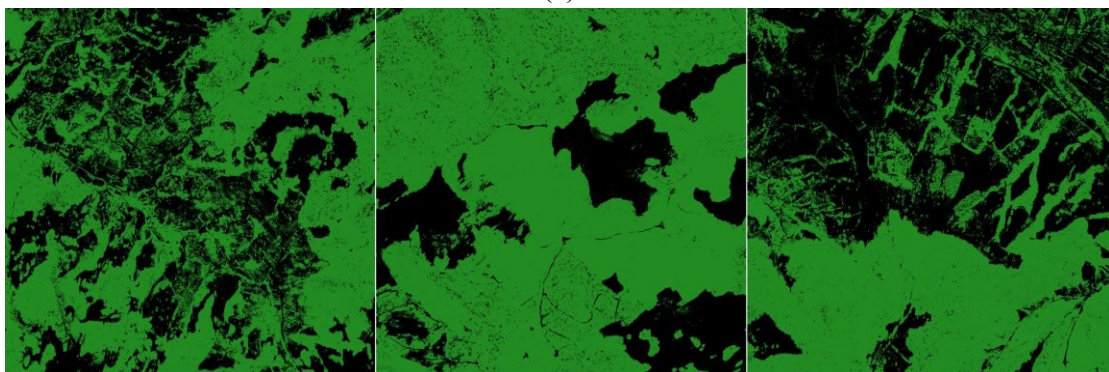
În final, cele 200 de imagini RGBN de intrare și rezultatele clusteringului au fost prezentate echipei de experți GIS pentru analiză. În urma etichetării imaginilor în cele cinci clase, acuratețea finală a GMM pentru cele patru scenarii considerate este prezentată în **Tabelul 4.2**. Cele mai bune rezultate au fost obținute utilizând date de intrare de  $4000 \times 4000$ . În acest caz, ambele metode de inițializare s-au comportat foarte bine, iar diferența dintre rezultatele lor a constat într-un număr mic de pixeli clasificați diferit, ceea ce nu a avut un impact real asupra evaluării experților și acurateței finale.

**Tabelul 4.2** Acuratețea algoritmului GMM

<b>Dimensiune</b> <b>Inițializare</b>	<b>1000 × 1000</b>	<b>4000 × 4000</b>
	<b>Aleatoare</b>	86.11
<b>K-Means</b>	88.89	92.22



(a)



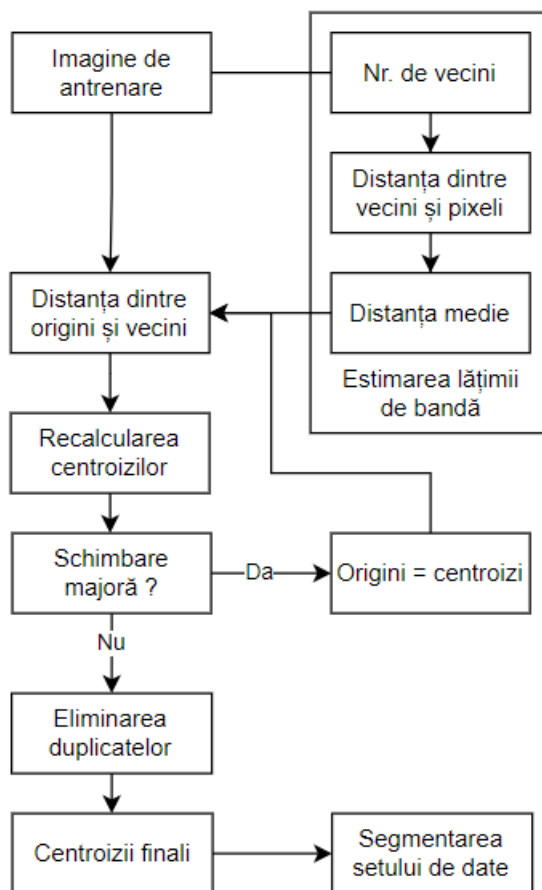
(b)

**Figura 4.18** Exemple ale clusteringului GMM; (a) imagini de intrare  $4000 \times 4000$ ; (b) imagini segmentate

## 4.4 Mean Shift

Dacă în cazul celorlalți trei algoritmi, fiecare imagine a fost segmentată independent, având o etapă de antrenare, unde s-au identificat centrele fiecărui cluster, și o etapă de predicție, unde s-a segmentat întreaga imagine, pentru Mean Shift s-a stabilit utilizarea unei singure imagini din care s-au extras centrele celor două cluster, apoi celelalte imagini au fost segmentate cu ajutorul acestor două centre. Pentru partea de testare și calculare a acurateței, s-au folosit 400 de imagini de  $1000 \times 1000$  din setul de date RGBN.

Pe lângă baza teoretică bine documentată a metodei Mean Shift [67], algoritmul implementat a fost modificat și augmentat cu anumite elemente. Implementarea întregii metode este ilustrată în **Figura 4.20**. În primul rând, imaginea de antrenare este introdusă într-un modul care estimează cel mai important parametru al algoritmului, și anume lățimea de bandă. În al doilea rând, imaginea de antrenare trece printr-un proces iterativ care furnizează o suită de centroizi. În urma eliminării centroizilor duplicați, rămân doar centroizii celor două clase de interes. Apoi, acești centroizi sunt utilizați pentru a prezice zonele de pădure și non-pădure pentru întreg setul de date.



**Figura 4.20** Diagrama algoritmului Mean Shift

Algoritmul necesită doi parametri ajustabili pentru a asigura convergența: factorul de scalare  $Q$  și numărul maxim de iterații  $T$ . După rularea mai multor scenarii, s-a stabilit  $Q = 0.2$ , respectiv o lățime de bandă  $B = 25.38$  ca fiind optimă

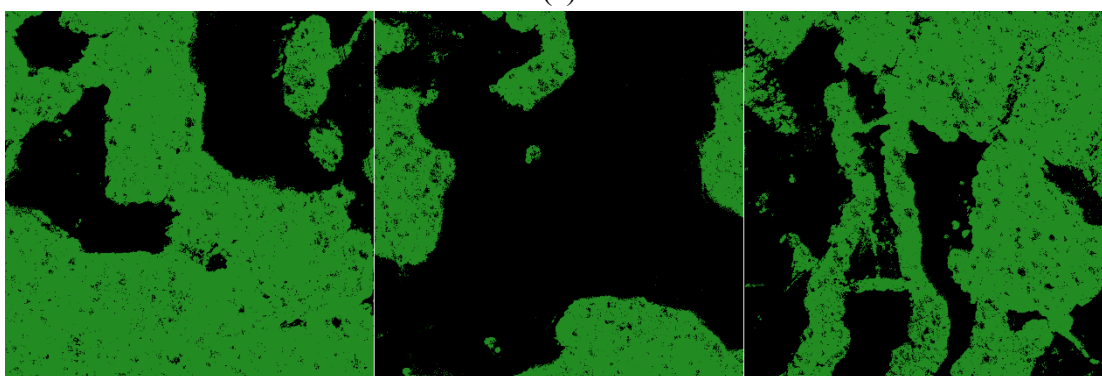
pentru acest set de date. Pentru acest factor de scalare s-au considerat trei praguri  $T$ , 20, 30 și 50. În urma evaluării experților GIS, algoritmul Mean Shift a obținut rezultate bune, apropiate de 90%:

- Mean Shift 20: 89.75%;
- Mean Shift 30: 90.75%;
- Mean Shift 50: 89.75%.

În **Figura 4.24** sunt prezentate unele imagini segmentate pentru cel mai bun scenariu identificat, respectiv clusteringul Mean Shift cu factorul de scalare  $Q = 0.20$  și pragul pentru numărul maxim de iterații  $T = 30$ .



(a)



(b)

*Figura 4.24* Mostre ale clusteringului Mean Shift; (a) imagini de testare; (b) imagini segmentate

## 4.5 Concluzii

Din cauza distinctelor caracteristici ale metodelor, acestea nu au putut fi aplicate uniform, pentru aceleași zone de interes sau dimensiune a datelor de intrare. Acestea au fost studiate independent, astfel încât, pentru a pune în perspectivă evaluarea experților, a fost necesară o comparație mai amănunțită pentru a stabili care tehnică a performat mai bine. Pentru aceasta, s-au măsurat timpul de execuție și memoria alocată a sistemului pentru fiecare etapă computațională, respectiv preprocesarea datelor, antrenarea și predicția.

Pentru a compara în mod corect metodele, s-a stabilit proporționarea măsurătorilor la cea mai mare dimensiune de intrare, respectiv cea a clusteringului



GMM de  $4000 \times 4000 \times 4$ . Concret, folosind dimensiunile de intrare și numărul de benzi ale celorlalte trei metode, s-a calculat câte un factor cu care s-au înmulțit toate valorile măsurate (**Tabelul 4.9**). În ansamblu, cea mai neperformantă metodă a fost AGNES, cu valori mult mai mari față de celelalte. Mean shift, chiar dacă a avut o acuratețe de peste 90%, necesită un timp de 2037.72 s și o memorie de 718.4 Mb pentru a clasifica o imagine de  $4000 \times 4000 \times 4$ . Alegerea celei mai performante tehnici s-a redus astfel la comparația dintre K-Means și GMM. Cu toate că GMM a necesitat un timp de peste două ori mai mare față de K-Means, acesta a performat mult mai bine în celelalte două categorii, a obținut cea mai mică alocare de memorie de doar 492.6 Mb și cea mai bună acuratețe de 92.22%, cu aproape 7 procente mai mult decât K-Means. În acest fel, s-a stabilit ca metoda de clustering care va sta la baza studiului ulterior să fie clusteringul GMM.

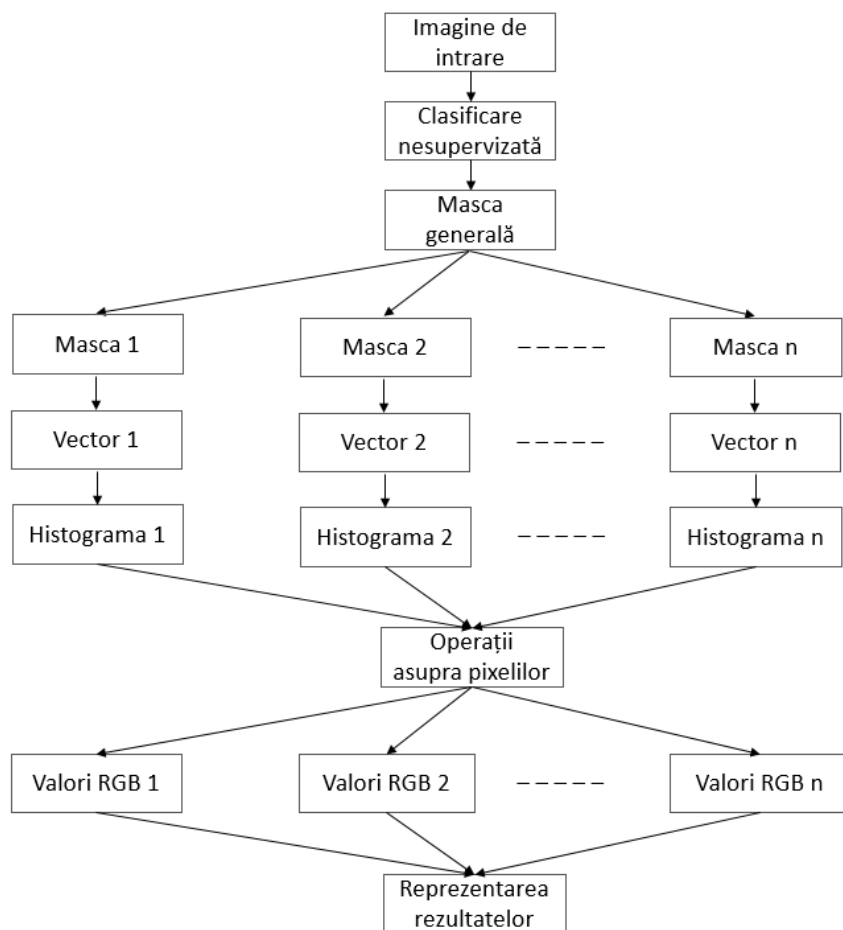
*Tabelul 4.9 Performanța proporționată a tehnicilor de clustering*

Metodă	Timp total (s)	Memorie totală (Mb)	Acuratețe (%)
K-Means	26.91	534.66	85.56
AGNES	34549.84	2447.2	87.78
GMM	61.89	492.6	92.22
Mean shift	2037.72	718.4	90.75

## Capitolul 5

# CEM – Modul de extragere a culorii

Din cauza absenței datelor etichetate, clasificarea nesupervizată nu poate asocia automat rezultatele clusteringului cu clasele reale de apartenență, și implicit reprezentarea lor în mod corect. Acest aspect al tehnicilor de clustering a reprezentat un impediment și pentru manipularea și interpretarea rezultatelor din timpul studiului, prin incapacitatea de a prezenta informații intuitive pentru un utilizator obișnuit și prin încetinirea etapei de evaluare. Procesul de dezvoltare al CEM a trebuit automatizat în așa fel încât să existe o legătură între metodele de clustering și imaginile de intrare. Pentru a servi și alte scopuri decât monitorizarea defrișărilor, s-a stabilit ca CEM să fie independent de metodele implementate sau de parametrii acestora și să fie ușor de aplicat în afara studiului actual. Diagrama algoritmului CEM este prezentată în **Figura 5.2**, generalizată pentru un număr infinit de clase. În aplicații reale, numărul de clase poate varia între 2, dacă este necesar identificarea unui element specific (lac, pădure, așezare umană, etc.) și mult mai multe pentru segmentări avansate ale acoperirii terenului.

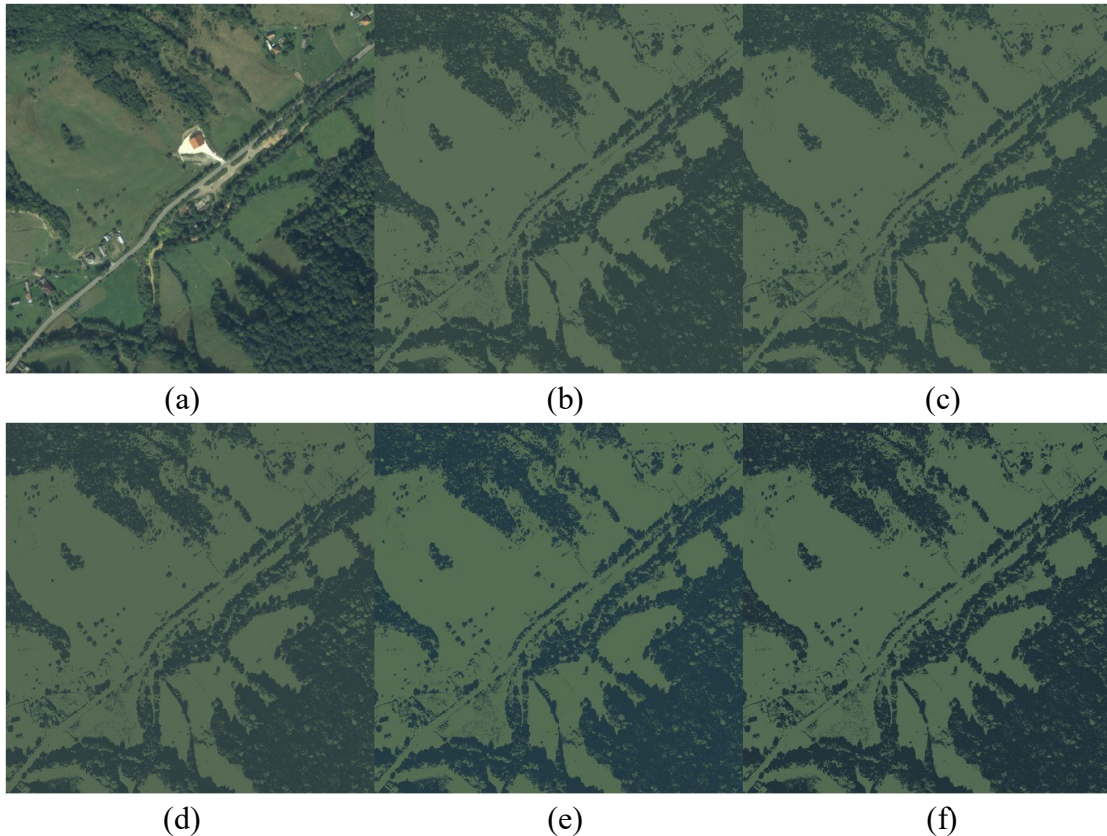


**Figura 5.2** Diagrama CEM

După etapa de antrenare, fiecare algoritm nesupervizat calculează propriul set de variabile care descriu clasele. Cu toate acestea, rezultatele sunt cuantificate folosind un atribut comun, respectiv eticheta atribuită fiecărui pixel. Matricea formată din totalitate etichetelor constituie masca generală a imaginii. În continuare această mască este împărțită în  $n$  măști binare, pentru fiecare clasa identificată în urma segmentării. În următoarea etapă, fiecare dintre aceste măști este suprapusă peste imaginea de intrare și se formează  $n$  vectori de valori care descriu fiecare clasă. Chiar dacă acești  $n$  vectori nu sunt imagini tradiționale, sunt totuși compuși din valori ale pixelilor, astfel încât histogramele lor sunt calculate. În următoarea etapă de operații asupra pixelilor, au fost implementate și comparate trei funcții de medie și două funcții de numărare pentru a concluziona care se potrivește cel mai bine scopului CEM.

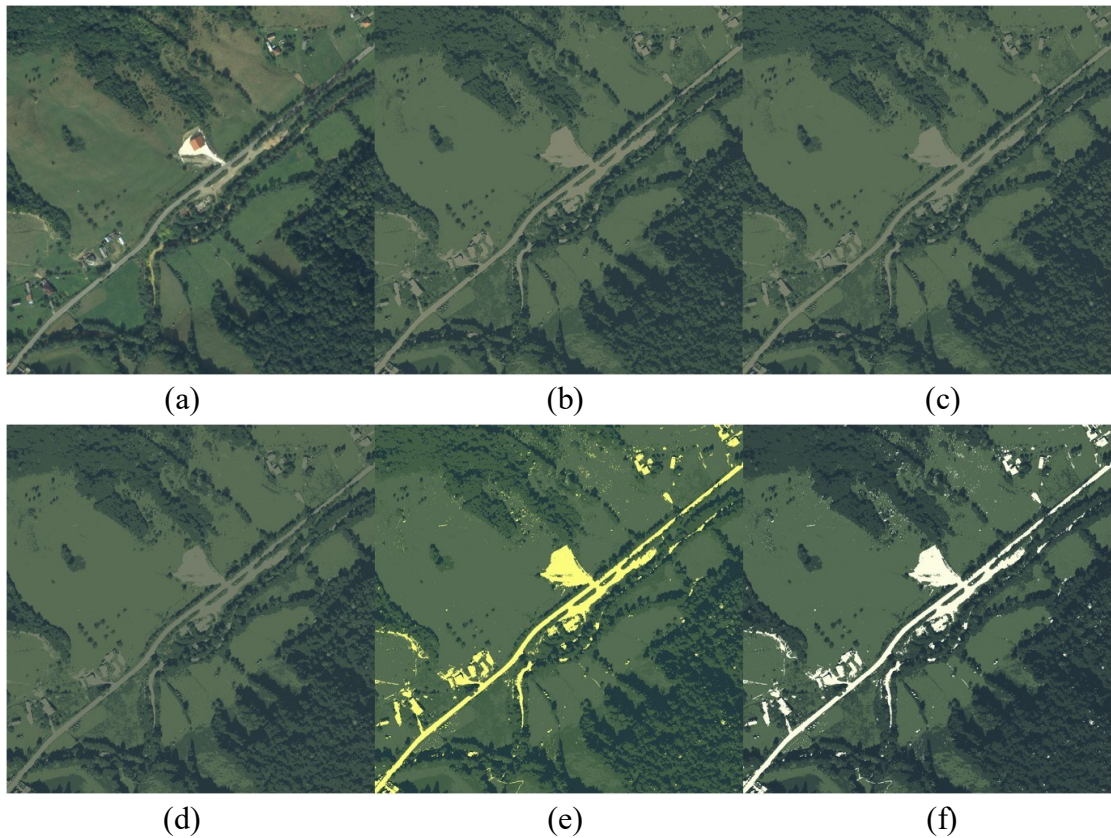
În final, cele trei valori RGB sunt utilizate pentru a reprezenta fiecare dintre clasele identificate. Alegerea celei mai bune funcții a fost stabilită prin comparație vizuală și numerică. În **Figura 5.6** este ilustrată reprezentarea finală pentru două clase. Cele două clase sunt reprezentate similar de cele cinci funcții, funcția mod generală având un contrast ușor mai bun decât celelalte. Față de cele trei medii care au valori mai apropiate între cele două clase, funcțiile mod au extras valori mai excentrice. În exemplul de mai jos, clasa non-pădure este compusă în cea mai mare parte din pajiște și a fost ilustrată cu un verde mai deschis decât pădurea, ceea ce

denotă că drumul și cele câteva elemente umane identificate în această clasă nu au avut un efect major asupra reprezentării finale.



**Figura 5.6** *Reprezentarea clusteringului în două clase; (a) Realitatea din teren; (b) AM; (c) GM; (d) HM; (e) Funcția mod locală; (f) Funcția mod generală;*

Pe de altă parte, reprezentarea finală pentru 4 clase este expusă în **Figura 5.7**, iar **Tabelul 5.2** prezintă valorile calculate pentru fiecare funcție. După cum se poate observa, modificarea numărului de clase de identificat pentru un algoritm de clustering a schimbat semnificativ modul de afișare al acestora pentru funcțiile de numărare. Cele trei medii au avut din nou rezultate și reprezentări similare. Pentru toate cele cinci funcții, cele trei clase care reprezintă pășuni, umbre și păduri au fost ilustrate într-un mod foarte intuitiv și aproape de culoarea lor reală. Principala diferență între metode a fost constituită de valorile RGB ale clasei infrastructură, care înglobează elementele umane din imagine. Aceasta a fost ilustrată de funcția mod local prin culoarea galbenă, deoarece valoarea de albastru calculată a fost mult mai mică decât roșu și verde. Funcția mod general a reprezentat-o în alb murdar (253,253,227) din cauza zonei din împrejurul casei centrale, care a constituit majoritate de pixeli. Pe de altă parte, funcțiile de medie au nivelat această zonă cu restul drumului și acoperișurilor și au obținut diferite nuanțe de gri. Acestea sunt obținute atunci când valorile RGB sunt asemănătoare, așa cum sunt evidențiate în **Tabelul 5.2**.



**Figura 5.7** Reprezentarea clusteringului în patru clase; (a) Realitatea din teren; (b) AM; (c) GM; (d) HM; (e) Funcția mod locală; (f) Funcția mod generală;

Comparând diferențele de reprezentare dintre două și patru clustere, s-a putut concluziona faptul că cele mai mari abateri față de realitate vor surveni în clasele cu foarte mare diversitate, cum este în acest caz, clasa infrastructură, care poate conține case cu diferite acoperișuri, drumuri de diferite materiale sau varii construcții civile. Având în vedere toate cele de mai sus, s-a stabilit că cea mai sigură și cea mai puțin imprevizibilă funcție pentru etapa de operații asupra pixelilor este una dintre funcțiile de medie, respectiv GM. HM reprezintă clasele mult prea întunecat, iar AM conține chiar nuanțe de galben în unele situații. CEM a fost implementat asupra metodelor de clustering, iar dezvoltarea acestuia a ajutat considerabil la accelerarea cercetărilor și studiului general, în special interpretarea vizuală și evaluarea rezultatelor metodelor de clustering. Diverse reprezentări pentru 4 clase sunt ilustrate în **Figura 5.9**.



(a)



(b)

*Figura 5.9 Exemple de reprezentări finale folosind CEM; (a) Realitatea din teren; (b) Reprezentarea CEM a clusteringului în patru clase*

## Capitolul 6

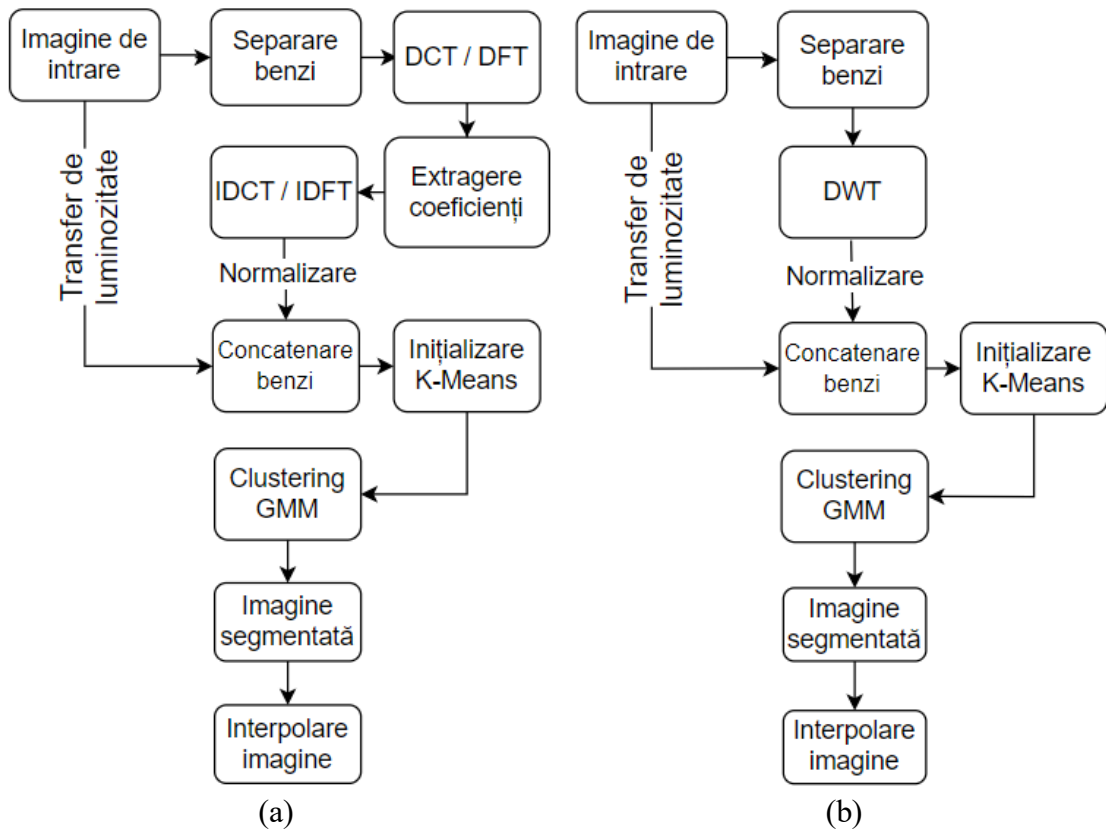
# Îmbunătățirea algoritmului GMM prin tehnici de compresie și procesare a semnalelor

După implementarea și compararea celor patru tehnici de clustering, la sfârșitul **Capitolului 4** s-a concluzionat faptul că cea mai bună bază pentru dezvoltarea unei metodologii pentru atingerea obiectivelor asumate este clusteringul GMM. Cu toate acestea, cea mai mare problemă identificată în aplicarea GMM a fost timpul îndelungat de calcul și alocările mari de memorie, indiferent de optimizările native ale algoritmului. Pentru a aborda această problemă, au fost studiate metode de compresie a imaginilor și procesare a semnalelor. Astfel, transformarea cosinus discretă (DCT), transformarea Fourier discretă (DFT) și transformarea wavelet discretă (DWT) au fost aplicate imaginilor de intrare înainte de clusteringul GMM.

În mod tradițional, transformările discrete sunt aplicate unor regiuni mai mici ale imaginii pentru a îmbunătăți etapa următoare de extragere de trăsături. Metodologia dezvoltată în cadrul acestui studiu propune o altă abordare, ilustrată în **Figura 6.2**, prin utilizarea unei ferestre de extragere pentru întreaga imagine. Mai întâi, imaginea de intrare trece printr-o etapă de descompunere pe benzi. Apoi, cele trei transformări discrete sunt calculate pentru fiecare bandă într-un spațiu bidimensional. Pentru a extrage doar coeficienții semnificativi, o fereastră de extragere este aplicată peste imaginea transformată. Poziția ferestrei depinde de tipul

de transformare. Pentru DCT, coeficienții cei mai importanți sunt concentrați în colțul din stânga jos al transformatei, în timp ce DFT, în urma translatării frecvențelor nule, îi concentrează în mijlocul imaginii. Mai multe scenarii au fost create prin modificarea dimensiunii ferestrei de extragere, pentru a decide cea mai bună valoare pentru obiectivele studiului și pentru a analiza diminuarea calității clusteringului și costurilor acestuia. Apoi, coeficienții extrași pentru fiecare bandă sunt converțiți înapoi folosind transformările inverse. Merită menționat faptul că acești coeficienți extrași nu sunt bordați cu valoarea zero pentru a se menține lungimea și lățimea originale ale imaginii, ceea ce înseamnă că în acest pas se realizează reducere de dimensionalitate. Pentru DWT, aceste două etape de extragere și transformare inversă nu există, deoarece această transformată extrage în mod natural coeficienți de aproximare, reducând dimensiunea imaginilor la jumătate. Astfel, nu a fost necesară implementarea unei ferestre de extragere și a unei transformări inverse.

După aplicarea normalizării și transferului de luminozitate asupra rezultatelor transformatelor discrete, benzile sunt concatenate și trec printr-o etapă rapidă de clustering K-Means. Aceasta servește ca inițializare pentru algoritmul GMM. Rapiditatea acesteia rezultă din stabilirea centrozilor inițiali în mod probabilistic, nu aleator, prin calcularea distanței dintre puncte. La final, rezultatele clusteringului GMM sunt redimensionate la mărimea intrării inițiale folosind o interpolare de tip cel mai apropiat vecin.



**Figura 6.2** Diagrama algoritmului de compresie: (a) DCT / DFT; (b) DWT

## 6.2 Evaluarea rezultatelor metodologiei propuse

Metodologia propusă a fost testată în cadrul unui clustering de cinci clase folosind un set de date generos de 1600 de imagini cu dimensiunea de  $1000 \times 1000 \times 4$ . Pentru a evalua rezultatele, am analizat metoda propusă din trei perspective: timpul de calcul, memoria alocată și performanța. Din păcate, pentru algoritmi de învățare nesupervizată nu se poate calcula o acuratețe în adevăratul sens al cuvântului, deoarece datele nu sunt etichetate. Din acest motiv, performanța a fost evaluată prin indicele Davies-Bouldin (DBI) [82]. Timpul de execuție, memoria alocată și scorurile DBI au fost măsurate pentru fiecare fereastră de extracție, dar și pentru algoritmul GMM convențional, nemodificat. Toate valorile au fost transformate în procente din algoritmul GMM de referință.

Astfel, procentele de timp și memorie ar fi trebuit să fie cât mai mici posibile, în timp ce procentele DBI ar fi trebuit să fie cât mai aproape de 100%. Chiar dacă toate cele trei scenarii de 500 au avut scăderi mici ale scorului DBI și timpi de calcul mai rapizi, rezultatele DFT 500 și DWT 500 au fost neglijate din cauza creșterii de memorie. DCT 500 a avut un timp de calcul de 32.82%, menținând în același timp calitatea clusteringului și memoria totală alocată. Pentru a doua grupă de 250, scorul DBI a scăzut cu aproximativ 10%, dar timpul total și memoria au avut îmbunătățiri substanțiale. De data aceasta, DCT 250 și DFT 250 au avut procente similare și au performanță mai bună decât DWT 250. În timp ce DTW 250 a avut un scor DBI mai bun, diferența de doar 2% nu este suficient de semnificativă pentru a revanșa alocarea mult mai mare a memoriei de 78.17%. Din păcate, în ciuda timpilor reduși de calcul, scenariile de 125 au avut o scădere de aproximativ 20% a scorurilor DBI. Per total, DCT s-a comportat mai bine decât celelalte două transformări, în principal pentru că furnizează informațiile sub forma unui singur set de coeficienți. DFT obține și coeficienți reali și coeficienți imaginari, în timp ce DWT furnizează o imagine de aproximare și trei imagini pentru detalii.

**Tabelul 6.3** Analiză procentuală a performanței scenariilor GMM

Scenariu	DBI (%)	Timp total (%)	Memorie totală (%)
DCT 500	95.35	32.82	93.53
DFT 500	94.31	31.42	115.63
DWT 500	95.89	32.02	109.97
DCT 250	89.76	8.94	53.37
DFT 250	89.00	9.26	56.87
DWT 250	91.78	8.89	78.17
DCT 125	80.99	2.78	41.78
DFT 125	79.02	3.22	42.59
DWT 125	87.66	3.13	66.04

Făcând o comparație între DCT 250 și DFT 250, putem observa că DCT 250 a avut o performanță ușor mai bună decât DFT 250 în toate cele trei măsurători. Luând în considerare toate informațiile prezentate mai sus, putem presupune că cele mai bune scenarii ale metodei propuse sunt DCT 500 și DCT 250. În funcție de condițiile

de implementare, ambele pot fi potrivite și luate în considerare. **Figura 6.8** prezintă imaginea de intrare și rezultatele celor două scenarii.



**Figura 6.8** Rezultatele celor mai bune scenarii: (a) Realitatea din teren; (b) DCT 500; (c) DCT 250

### 6.3 Concluzii

Acest studiu a încercat să implementeze un algoritm nesupervizat rapid și ieftin care să țină pasul cu fenomenul de defrișare și să ofere analize și avertizare în timp real instituțiilor abilitate în domeniu, prin îmbunătățirea unei metode de clustering anterior analizată și aplicată aceluiași set de date. Studiul s-a concentrat pe reducerea timpului de calcul al clusteringului GMM, metodă aplicată anterior pentru segmentarea pădurilor, propunând un algoritm care a exploatat avantajele transformărilor discrete folosite în mod convențional în comprimarea imaginilor și procesarea semnalelor.

Algoritmul propus a fost testat folosind un set de date de 1600 de imagini aeriene de  $1000 \times 1000 \times 4$  și rezoluție spațială de 50 cm fiecare, prin măsurarea timpului de calcul, scorului DBI și memoriei alocate pentru fiecare dintre scenariile considerate. Toate rezultatele au fost comparate cu algoritmul GMM anterior, care a servit ca bază de referință pentru comparații, performanța sa fiind măsurată în aceleași condiții. Scenariile au avut timpi de execuție mult mai buni, însă doar DCT 500, DCT 250 și DFT 250 au menținut un scor DBI apropiat de algoritmul GMM de referință, îmbunătățind în același timp și alocarea de memorie. În medie, algoritmul GMM a avut un timp total de execuție de 19.9158 s și o alocare de memorie de 37.1 Mb. DCT 500 a reușit să reducă aceste costuri la 6.5360 s și 34.7 Mb, cu o deprecieri foarte redusă a scorului DBI.

În concluzie, această metodologie a atins obiectivele studiului și poate fi considerată o soluție viabilă pentru clasificările cu resurse reduse. Putem afirma cu tărie că algoritmul descris rezolvă problema timpului de execuție și chiar a memoriei alocate de unitatea de calcul.

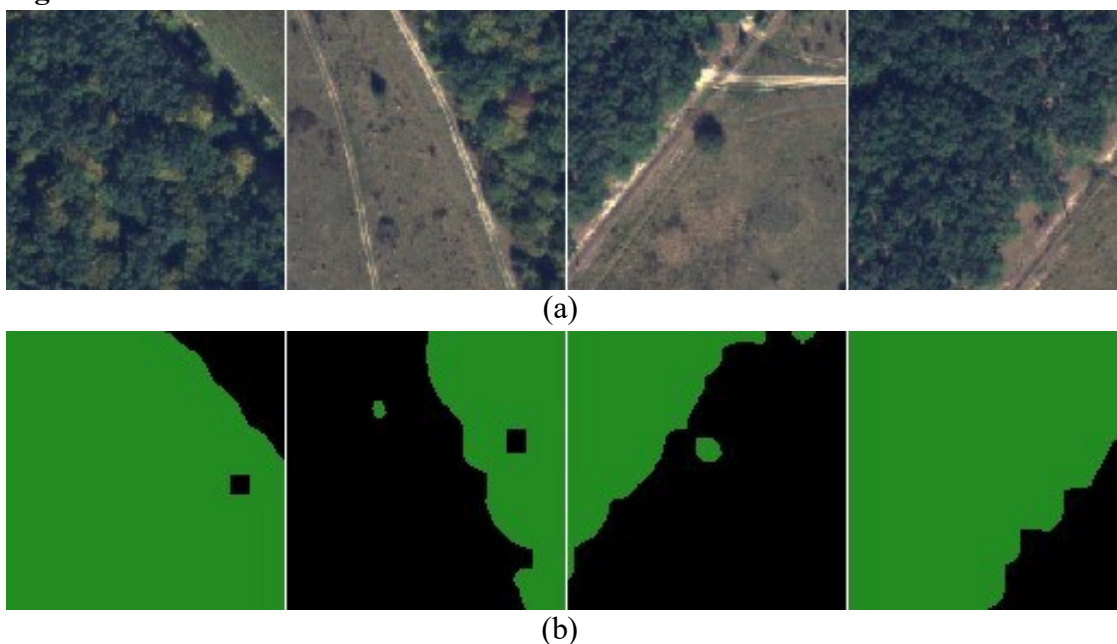


# Capitolul 7

## Învățarea supervizată prin arhitectura U-Net

### 7.1 Etichetarea automată a datelor

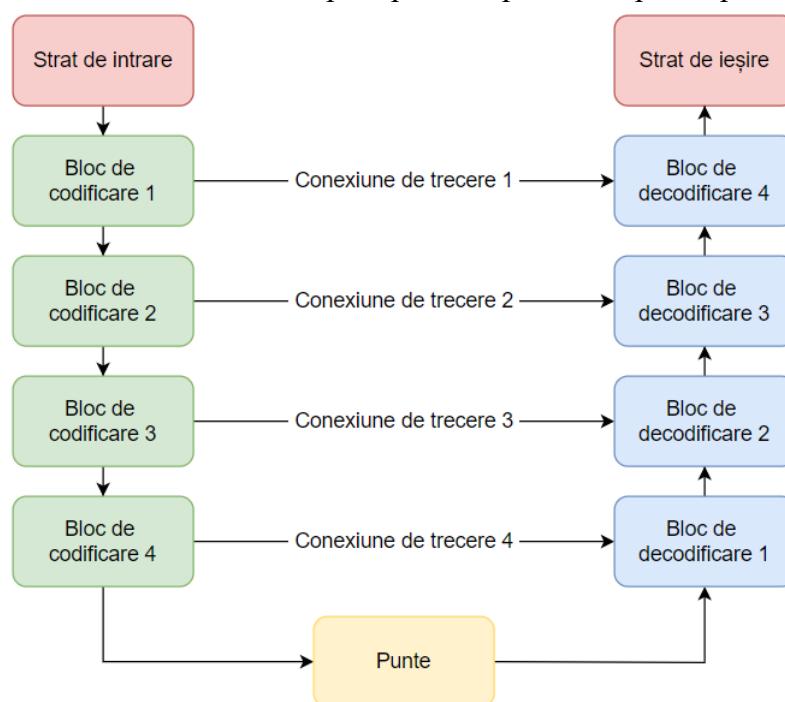
Acest studiu propune o abordare folosind clusteringul GMM pentru etichetarea semantică a imaginile, înainte de antrenarea arhitecturilor U-Net pentru clasificarea pădurilor. Pentru stabilirea setului de date, s-au aplicat concluziile **Capitolului 4**, respectiv utilizarea a 100 de imagini aeriene multispectrale de  $4000 \times 4000 \times 4$  din zona Târgu-Mureș. Segmentarea prin GMM este augmentată prin mai multe tehnici pentru a îmbunătăți etichetarea, cum ar fi calculul criteriilor informaționale, unificarea clusterelor și filtrarea imaginilor rezultate. După aplicarea GMM, sunt calculate scorurile criteriului informațional Akaike (AIC) și criteriul informațional Bayesian (BIC) pentru a valida numărul optim de cluster. Apoi, clusterelor rezultate sunt unite pentru a forma doar cele două clase ale studiului: pădure și non-pădure. După filtrarea rezultatelor, scorul DBI a fost folosit pentru a alege cele mai bine etichetate imagini, construind astfel seturile de date de antrenare, validare și testare a U-Net. În cele din urmă, setul de date etichetat este împărțit în loturi mai mici de  $128 \times 128$  și introdus în arhitecturi U-Net compuse din 19 straturi convoluționale și 4 conexiuni de trecere. În **Figura 7.7** sunt ilustrate mostre ale setului de date etichetat.



**Figura 7.7** Exemple ale etichetării automate: (a) Imagini de antrenare; (b) Etichetele acestora

## 7.2 Descrierea arhitecturii U-Net

U-Net utilizează o arhitectură tip codificare-decodificare, compusă din două căi principale. Căia de codificare reduce sistematic dimensiunile imaginii de intrare, extrăgând astfel caracteristici de nivel superior. Pe de altă parte, căia de decodificare restaurează caracteristicile codificate prin supraeșantionare, conducând la o mască de segmentare îmbunătățită cu o rezoluție spațială ridicată. Această arhitectură unică permite rețelei să înțeleagă atât contextul mai larg, cât și detaliile locale în timpul procesului de învățare. Arhitectura originală U-Net este compusă dintr-un strat de intrare, patru blocuri de codificare, o punte, patru blocuri de decodificare, patru conexiuni de trecere și un strat de ieșire (**Figura 7.8**). U-Net folosește funcția de activare ReLU (Rectified Linear Unit) pentru toate straturile din model, cu excepția ultimului, care necesită o funcție sigmoid [86]. De asemenea, la sfârșitul căii de codificare, este folosit un strat de dropout pentru a preveni supraadaptarea modelului.



**Figura 7.8** Arhitectura originală U-Net

## 7.3 Optimizarea hiperparametrilor nonstructurali

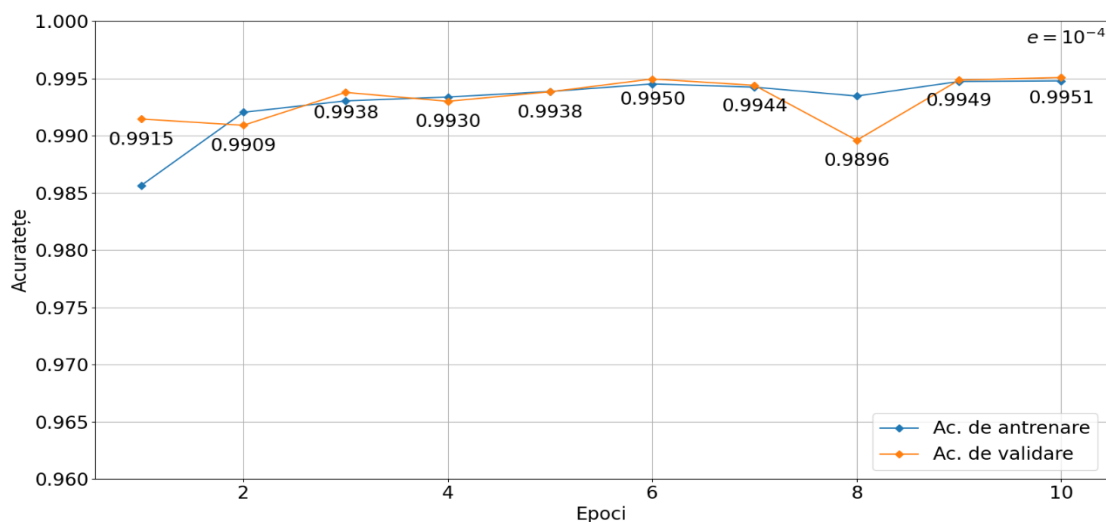
Multitudinea de hiperparametri ai unei rețele CNN poate fi împărțită în două grupuri. Hiperparametrii structurali sunt cei prezentați în secțiunea anterioară, respectiv: numărul de straturi convoluționale, numărul de nuclee, dimensiunea nucleului din fiecare convoluție, funcțiile de activare și dimensiunile de votare. Pentru ajustarea celorlalți hiperparametri nonstructurali, trebuie să se ia în considerare puterea de calcul. Pentru a aborda problema, dimensiunea lotului de antrenare, care determină numărul de eșantioane procesate într-o trecere înainte și înapoi pentru fiecare iterație, a fost indexat constant până când sistemul a returnat o eroare OOM (Out of Memory).

În acest fel s-a putut determina valoarea maximă posibilă suportată de unitatea de calcul. În această analiză s-a setat modelul să calculeze doar o epocă și s-a monitorizat timpul de execuție pentru acea epocă. Rezultatele sunt prezentate în **Tabelul 7.1**. Astfel, dimensiunea lotului maxim acceptat de GPU a fost 16, care a avut și cel mai bun timp de antrenare conform tabelului.

**Tabelul 7.1** Timpii de execuție pentru diferite dimensiuni ale lotului de antrenare

Dimensiune lot de antrenare	2	4	8	16	32
Timp de execuție (s)	1502	1112	805	520	OOM
Acuratețea de validare	0.9897	0.9912	0.9914	0.9903	OOM

Având o acuratețe inițială atât de bună, s-a concluzionat ca restul ajustărilor să fie realizate prin microantrenare de-a lungul a 10 epoci. Stabilirea dimensiunii lotului și a optimizerului ne-au îndreptat către următorul hiperparametru de ajustat, respectiv rata de învățare. Cea mai bună acuratețe de validare de 0.9951 a fost obținută pentru rata de învățare de  $10^{-4}$ . De asemenea, pentru această valoare, acuratețea de validare a fost menținută mai aproape de acuratețea de antrenare pe parcursul celor 10 epoci.



(c)

**Figura 7.10** Acuratețea microantrenării U-Net pentru diferite rate de învățare:

(a)  $10^{-2}$ ; (b)  $10^{-3}$ ; (c)  $10^{-4}$ ; (d)  $10^{-5}$

## 7.4 Simplificarea modelului U-Net

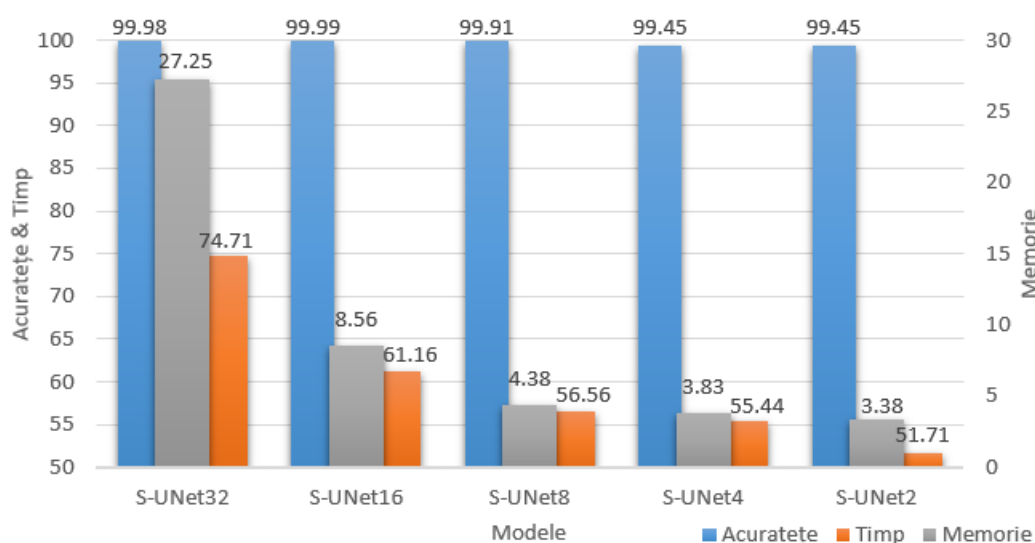
Din păcate, arhitectura originală, așa cum a fost prezentată anterior în **Secțiunea 7.2** și cum a fost propusă inițial în [25], a fost mult prea complexă pentru obiectivele acestui studiu. Având toți hiperparametrii nonstructurali ajustați, s-a stabilit ca abordarea simplificării arhitecturii U-Net reducerea sistematică a numărului de filtre prezente în

fiecare strat de convoluție la jumătate. Astfel, au fost create mai multe scenarii menținând regulile de simetrie și indexare ale arhitecturii originale.

Pentru a analiza în detaliu performanțele modelelor simplificate, s-a măsurat memoria alocată, acuratețea și timpul de execuție pentru o singură imagine aleatoare. Pentru o mai bună înțelegere a îmbunătățirilor scenariilor propuse, măsurătorile au fost reprezentate grafic în **Figura 7.12**. Valorile corespunzătoare graficului se regăsesc în **Tabelul 7.5** și reprezintă procente din scenariul original U-Net. Astfel, un scenariu performant va fi descris de o acuratețe cât mai apropiată de 100%, și de un timp și o memorie alocată cât mai redusă. Graficul are două axe verticale, acuratețea și timpul sunt proiectate pe axa din stânga, în timp ce memoria este proiectată cea din dreapta. Putem afirma cu tărie că toate scenariile au avut rezultate foarte bune. Acuratețea menține procente ridicate în timp ce timpul și memoria scad constant până la punctul în care S-UNet2 calculează de două ori mai rapid și utilizează doar 3.38% din memoria inițială. Analizând cele două expuneri se poate observa faptul că S-UNet8 are cea mai bună balanță între acuratețe și reducerea de timp și memorie. Acesta a reușit să mențină 99.91% din acuratețea modelului U-Net inițial folosind doar 4.38% din memoria acestuia și având un timp de execuție de aproape două ori mai rapid.

**Tabelul 7.5** Analiză procentuală a performanței modelelor U-Net simplificate

Model	Acuratețe (%)	Timp (%)	Memorie (%)
S-UNet32	99.98	74.71	27.25
S-UNet16	99.99	61.16	8.56
S-UNet8	99.91	56.56	4.38
S-UNet4	99.45	55.44	3.83
S-UNet2	99.45	51.71	3.38



**Figura 7.12** Graficul analizei procentuale a performanței modelelor U-Net simplificate

În ceea ce privește reducerea de cost a metodei propuse, o analiză mult mai complexă poate fi realizată. În **Tabelul 7.7** sunt descrise memoria utilizată, numărul total de parametri și numărul total de filtre ale LCU-Net [41], Half-Unet [43] și ale

modelelor U-Net simplificate. Chiar dacă LCU-Net are același număr de blocuri și creștere a filtrelor ca al treilea scenariu, respectiv de la 16 la 256, datorită implementării tehnicii Inception, numărul total de filtre și parametri este mult mai mare. Pe de altă parte, Half-UNet are 6 blocuri cu câte 64 de filtre fiecare, deoarece dezactivează 3 dintre cele 4 blocuri ale decodificatorului, rezultând doar 212.576 parametri. Cu excepția S-UNet32, metoda propusă a avut rezultate mai bune în ceea ce privește reducerea costurilor pentru toate celelalte modele. S-UNet16 a reușit să folosească doar 55.7 Mb față de nevoile LCU-Net și Half-UNet, respectiv 103.5 Mb și 137.3 Mb. În ceea ce privește timpul mediu, această măsurătoare poate fi puternic influențată de resursele hardware. Nu au fost prezentate date pentru Half-UNet, dar LCU-Net a prezis rezultatele în medie în 0.15 secunde, folosind un procesor Intel(R) Core(TM) i7-8700 cu 3.20 GHz, 32 Gb RAM și o placă grafică Nvidia GeForce RTX 2080 cu 8 Gb. Pe lângă faptul că modelele U-Net simplificate au avut timpi de predicție mult mai buni, de sub 0.10 s, unitatea de calcul folosită în cazul nostru a avut și o placă video mai puțin performantă, ceea ce ar duce la rezultate și mai rapide în cazul unor resurse disponibile similare cu LCU-Net.

*Tabelul 7.7 Comparație cu studii conexe a costurilor și complexității modelelor*

<b>Model</b>	<b>Memorie (Mb)</b>	<b>Timp mediu de predicție (s)</b>	<b>Nr. total de filtre</b>	<b>Nr. total de parametri</b>
LCU-Net	103.5	0.15	5392	3469393
Half-UNet	137.3	-	768	212576
U-Net	650.7	0.142	6848	31032321
S-UNet32	177.3	0.106	3424	7760385
S-UNet16	55.7	0.087	1712	1941249
S-UNet8	28.5	0.081	856	485889
S-UNet4	24.9	0.079	428	121761
S-UNet2	22	0.074	214	30585

## 7.5 Concluzii

Acest capitol a abordat o problemă comună în antrenarea rețelelor neurale, lipsa datelor etichetate, și dezavantajele obținerii acestora. Pentru a rezolva această problemă, s-a utilizat metoda de clustering GMM pentru etichetarea automată a datelor și s-a implementat un flux de lucru pentru obținerea unui set de date consistent pentru antrenarea supervizată a unei rețele neuronale convoluționale care să identifice cu o acuratețe foarte ridicată pădurile în imaginile aeriene. Consistența setului de date a fost validată prin antrenarea de modele CNN bazate pe arhitectura U-Net. Beneficiind de calitatea setului de date, acuratețea rezultată în cadrul antrenării, validării și testării a fost extrem de ridicată. Hiperparametrii non-structurali, cum ar fi rata de învățare, optimizatorul sau dimensiunea lotului, au avut o flexibilitate neobișnuită, demonstrând că etapa de clustering GMM a îmbunătățit și mai mult antrenarea supervizată. Modelul U-Net original a atins o acuratețe de validare de 0.9951 și o acuratețe de testare de 0.9969, doar pentru o microantrenare timp de 10

epoci. În concluzie, putem afirma cu tărie că metoda GMM ajută la etichetarea și augmentarea datelor și la reducerea resurselor de calcul necesare pentru antrenarea supervizată a unui CNN pentru monitorizarea rapidă și ieftină a defrișărilor. În ceea ce privește optimizarea hiperparametrilor, studiul a arătat că un model S-UNet8 cu 856 de filtre are o acuratețe mai mult decât satisfăcătoare și reduce utilizarea memoriei cu 95.62%, obținând astfel un model cu cost redus pentru segmentarea semantică a unor zone extinse la o întârziere de doar 5 secunde.

## Capitolul 8

### Concluzii

Această lucrare a urmărit dezvoltarea unui algoritm complet autonom de identificare a acoperirii terenului, aplicat pentru monitorizarea defrișărilor, care să nu necesite intervenție umană. În acest mod, metoda să poată segmenta o imagine aeriană cu o precizie cât mai bună, fără a fi necesare etichetarea manuală a pixelilor sau obiectelor sau alte intervenții umane externe. Motivația din spatele alegerii acestei aplicabilități a fost că fenomenul de defrișare reprezintă o problemă globală, scăpată de sub control și cu consecințe grave asupra mediului și a tuturor, iar monitorizarea tradițională prin prezența activistului de mediu sau organelor abilitate la teren, este costisitoare, consumatoare de timp și are un randament scăzut.

#### 8.2 *Contribuții originale*

Contribuțiile originale ale tezei de doctorat sunt:

- Studiu asupra unor metode cunoscute de clustering aplicate imaginilor aeriene multispectrale, influența parametrilor specifici acestora asupra rezultatelor clasificărilor și compararea lor în materie de memorie alocată și timp de execuție pentru a determina cea mai performantă metodă (C1, C2, C4, C5).
- Implementarea unui modul de extragere a culorilor pentru reprezentarea clusterelor identificate de metodele nesupervizate (C3). Chiar dacă CEM a fost necesar pentru nevoile interne ale tezei, acesta poate fi aplicat oricărui studiu care folosește date neetichetate pentru reprezentarea rezultatelor și pentru o mai bună analiză a operatorului.
- Reducerea drastică a timpului de predicție al algoritmului GMM prin implementarea unei metodologii de procesare a imaginilor multispectrale bazată pe DCT și extracția celor mai importanți coeficienți (J2). În mod tradițional, GMM poate avea răspunsuri foarte lente, în special dacă există prea multe cluster de identificat sau imaginile sunt mult prea mari. Spre

deosebire de tehnicile supervizate, unde puteți implementa modelul antrenat oriunde este necesar pentru predicții viitoare, în clusteringul imaginilor, preprocesarea datelor și calcularea centroizilor sunt etape realizate împreună cu predicția, nu înainte de implementarea soluției. Chiar dacă a existat o creștere a timpului de preprocesare, metodologia propusă a redus semnificativ timpul de antrenare și predicție.

- Etichetarea automată a pixelilor fără nicio intervenție umană (J1). Această teză a implementat o tehnică complet nesupervizată pentru a eticheta imaginile multispectrale, fără nicio altă antrenare supervizată sau rețea pre-antrenată ca punct de plecare. Comparativ cu alte studii, setul de date folosit a fost alcătuit din imagini aeriene proprii înregistrate cu capacități moderne. Pentru a demonstra succesul tehnicii, realitatea din teren și etichetele corespunzătoare au fost testate în raport cu mai multe arhitecturi U-Net pentru a dovedi utilitatea acestora și a metodei. Studiul a demonstrat că etichetele sunt suficient de precise astfel încât U-Net să poată învăța prin intermediul lor și să prezică rezultate foarte bune fără intervenție umană de etichetare manuală.
- Reducerea complexității modelului U-Net (J1). Majoritatea implementărilor în condiții reale ale modelelor AI sunt realizate în condiții dificile sau cu capacități hardware foarte slabe, fără a menționa necesitatea de analiză a datelor și predicție rapidă. Studiul prezentat a efectuat teste pe modele U-Net simplificate denumite S-Unet și a obținut rezultate precise similare cu arhitectura originală, reducând în același timp complexitatea modelului de până la 60 de ori. Reducerea complexității a redus de asemenea, timpul mediu de execuție, și memoria alocată de sistem, astfel s-a obținut o reducere de cost semnificativă.
- Segmentare rapidă de zone extinse (J2). Imaginile satelitare și aeriene conțin un număr foarte mare de pixeli din cauza câmpului lor larg de vedere (FOV) și rezoluției ridicate. Acest lucru face ca segmentarea acestor imagini să fie mult mai lentă decât a imaginilor cotidiene, care conțin obiecte. Metodele propuse pot prezice mii de kilometri pătrați de imagini multispectrale într-un interval de timp rezonabil.
- Dezvoltarea unui algoritm rapid, cu costuri reduse și prietenos cu mediu, de monitorizare a defrișărilor, care poate fi implementat pe o platformă aeriană de mici dimensiuni (J1). Spre exemplu, S-Unet8 necesită 28.5 Mb pentru a prezice  $10^6$  pixeli în aproximativ 5 s. Acest algoritm poate ajuta substanțial combaterea defrișărilor ilegale. Prin aplicarea lui într-un mediu real se creează un flux de lucru pentru monitorizarea zonelor de interes, identificarea infractorilor și aplicarea legilor.

## **8.3 Lista lucrărilor originale**

### **8.3.1 Articole publicate în jurnale științifice internaționale**

(J1) **A.-T. Andrei**, O. Grigore, „*Low-Cost Optimized U-Net Model with GMM Automatic Labeling Used in Forest Semantic Segmentation*”, *Sensors*, 23(21),

p. 8991, 2023, **WOS:001099507300001**, Articol Q2, Factor de Impact: 3.9, eISSN:1424-8220, DOI:10.3390/s2321899

(J2) **A.-T. Andrei**, O. Grigore, „*Development of a Very Low-Cost Deforestation Monitoring System Based on Aerial Image Clustering and Compression Techniques*”, *Advances in Electrical and Computer Engineering*, 24(2), pp. 73-84, 2024, **WOS:001242091800008**, Articol Q4, Factor de Impact: 0.8, eISSN: 1844-7600, DOI:10.4316/AECE.2024.02008

(J3) **A.-T. Andrei**, O. Grigore, „*Study of Clustering Algorithms for the Development of a Fast Deforestation Monitoring Tool*”, *Forests*, Articol Q1, Factor de Impact: 2.7, eISSN: 1999-4907 - trimis spre publicare

### **8.3.2 *Lucrări în volume de conferințe științifice internaționale indexate WOS***

(C1) **A.-T. Andrei**, O. Grigore, „*Unsupervised Machine Learning Algorithms Used in Deforested Areas Monitoring*”, 2021 International Conference on e-Health and Bioengineering (EHB), pp. 1-4, Iași, România, 2021, **WOS:000802227900197**, ISSN:2575-5145, ISBN:978-1-6654-4000-4, DOI:10.1109/EHB52898.2021.9657737.

### **8.3.3 *Lucrări în volume de conferințe științifice internaționale indexate BDI***

(C2) **A.-T. Andrei**, O. Grigore, „*Mean Shift Clustering with Bandwidth Estimation and Color Extraction Module Used in Forest Segmentation*”, 2023 13th International Symposium on Advanced Topics in Electrical Engineering (ATEE), pp. 1-6, București, Romania, 2023, ISSN:1843-8571, ISBN:978-1-4799-7514-3, DOI:10.1109/ATEE58038.2023.10108106

(C3) **A.-T. Andrei**, O. Grigore, „*Color Extraction Module for Unsupervised Image Classification Representation*”, 2023 13th International Symposium on Advanced Topics in Electrical Engineering (ATEE), pp. 1-5, București, Romania, 2023, ISSN:1843-8571, ISBN:978-1-4799-7514-3, DOI:10.1109/ATEE58038.2023.10108115

(C4) **A.-T. Andrei**, O. Grigore, „*Gaussian Mixture Model Application in Deforestation Monitoring*”, 2022 International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), pp. 26-31, Ankara, Turcia, 2022, ISSN:2770-7962, ISBN:978-1-6654-7013-1, DOI:10.1109/ISMSIT56059.2022.9932845

(C5) **A.-T. Andrei**, O. Grigore, „*Combating Deforestation Using Different AGNES Approaches*”, 2022 14th International Conference on Communications (COMM), pp. 1-5, București, România, 2022, ISBN:978-1-6654-9485-4, DOI:10.1109/COMM54429.2022.9817217

### **8.3.4 *Rapoarte de cercetare științifică***

(R1) **A.-T. Andrei**, Coordonator: O. Grigore, „*Studiul și pregătirea datelor aeriene multispectrale*”

(R2) **A.-T. Andrei**, Coordonator: O. Grigore, „*Algoritmi aplicați imaginilor multispectrale*”



(R3) **A.-T. Andrei**, Coordonator: O. Grigore, „*Studiul metodelor de învățarea supervizată*”

(R4) **A.-T. Andrei**, Coordonator: O. Grigore, „*Studiul metodelor de învățarea nesupervizată*”

## Bibliografie

- [6] S. Bhagwat, *The History of Deforestation and Forest Fragmentation: A Global Perspective*, Global Forest Fragmentation, pp. 5-19, 2014.
- [7] J. Shroder, R. Sivanpillai, *Biological and Environmental Hazards, Risks, and Disasters*, Elsevier, pp. 313-315, 2015.
- [8] A. Angelsen, D. Kaimowitz, *Rethinking the Causes of Deforestation: Lessons from Economic Models*, The World Bank Research Observer, 14(1), p. 73, 1999.
- [10] B. Usman, *Satellite Imagery Land Cover Classification using K-Means Clustering Algorithm Computer Vision for Environmental Information Extraction*, Elixir Comp. Sci. & Engg., 2013.
- [11] X. Zheng, Q. Lei, R. Yao, Y. Gong, Q. Yin, *Image segmentation based on adaptive K-means algorithm*, EURASIP Journal on Image and Video Processing, 2018.
- [13] K. Pearson, *Contributions to the Mathematical Theory of Evolution*, University College, London, 1894.
- [14] A. P. Dempster, N. M. Laird, D. B. Rubin, *Maximum Likelihood from Incomplete Data via the EM Algorithm*, Royal Statistical Society, 1977.
- [15] Y. Tarabalka, J. A. Benediktsson, J. Chanussot, *Spectral–Spatial Classification of Hyperspectral Imagery Based on Partitional Clustering Techniques*, IEEE Transactions on Geoscience and Remote Sensing, 47(8), pp. 2973-2987, 2009.
- [16] H. Permuter, J. Francos, I. Jermyn, *A study of Gaussian mixture models of color and texture features for image classification and segmentation*, Pattern Recognition, 39(4), pp. 695-706, 2006.
- [17] R. Farnoosh, B. Zarpak, *Image Segmentation Using Gaussian Mixture Model*, IUST International Journal of Engineering Science, 19(1-2), pp. 29-32, 2008.

- [21] Y. LeCun et al., *Backpropagation Applied to Handwritten Zip Code Recognition*, *Neural Computation*, 1(4), pp. 541-551, 1989.
- [22] A. Krizhevsky, I. Sutskever, G.E. Hinton, *ImageNet classification with deep convolutional neural networks*, *Communications of the ACM*, 60(6), pp. 84-90, 2012.
- [25] O. Ronneberger, P. Fischer, T. Brox, *U-Net: Convolutional Networks for Biomedical Image Segmentation*, *Medical Image Computing and Computer-Assisted Intervention*, Springer, 9351, pp. 234-241, 2015.
- [41] J. Zhang et al., *LCU-Net: A novel low-cost U-Net for environmental microorganism image segmentation*, *Pattern Recognition*, Springer, 115(4), 2021.
- [43] L. Haoran, S. Yifei, T. Jun, X. Shengzhou, *Half-UNet: A Simplified U-Net Architecture for Medical Image Segmentation*, *Frontiers in Neuroinformatics*, 16, 2022.
- [67] D. Comaniciu, P. Meer, *Mean shift: a robust approach toward feature space analysis*, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(5), pp. 603-619, 2002.
- [82] J. E. Cavanaugh, A. A. Neath, *The Akaike information criterion: Background, derivation, properties, application, interpretation, and refinements*, *WIREs Computational Statistics*, 11(3), p. 1460, 2019.
- [86] M. Yaqub et al., *State-of-the-Art CNN Optimizer for Brain Tumor Segmentation in Magnetic Resonance Images*, *Brain Sci*, 10(7), p. 427, 2020.