



Norges miljø- og
biovitenskapelige
universitet

Masteroppgave 2024 30 stp
Fakultetet for realfag og teknologi

Landdekkeklassifisering fra hyperspektrale bilder

Landcover and landuse based on hyperspectral
images

Kristoffer Kihl
Tormod Strand
Geomatikk

Forord

Dette er vår avsluttende oppgave på sivilingeniørstudiet i Geomatikk ved Norges miljø- og biovitenskapelige universitet (NMBU). Oppgaven er avlagt ved fakultetet for realfag og teknologi våren 2024.

Et godt samarbeid med Forsvarets Forskningsinstitutt har gitt oss en spennende og meget utfordrende oppgave.

Vi vil takke våre veiledere for god oppfølging og en opplevelse av at vi har utgjort et veldig godt og utfyllende team. En stor takk til vår biveileder, Thomas Olsvik Opsahl, for engasjement og støtte gjennom hele prosessen. Den faglige kompetansen og verdifulle innspill har vært til stor hjelp. Mange takk til hovedveileder, Ivar Maalen-Johansen, for gode avklaringer og presiseringer når målet har vært litt diffust.

Kristoffer Kihl
Tormod Strand

Ås
15/5.2024

Sammen drag

Landdekke klassifisering basert på hyperspektrale bilder og informasjon har blitt, og blir benyttet, i stor grad med informasjon hentet fra sensorer båret av satellitter. Landdekketklassifisering basert på høyoppløselige hyperspektral bilder er ikke så mye brukt, grunnet tilgjengelighet.

Denne oppgaven undersøker mulighetene og utfordringene ved å benytte høyoppløselige hyperspektrale bilder og data for landdekke- og arealklassifisering. Hyperspektrale sensorer kan gi svært detaljert informasjon om jordoverflaten gjennom mange smale spektrale bånd. Dette åpner for nye anvendelser innen kartlegging og overvåking av ulike areal typer og materialer. Oppgaven vår ser nærmere på bruk av hyperspektrale data for generell klassifisering av areal typer. Ulike maskinlæringsmetoder som Support Vector Machine og Random Forest benyttes for å trene klassifiseringsmodeller i Catalyst, basert på treningsdata. Resultatene viser at det er mulig å oppnå akseptable klassifiseringsnøyaktigheter for disse areal typene. Videre belyses utfordringer knyttet til hyperspektrale data, som stor datamengde, dimensjonalitetsreduksjon, datasettbalansering og effekter av skygger. Manuelle analyser av slike data er svært krevende, og oppgaven ser derfor på bruk av tilgjengelig programvare og verktøy for effektiv prosessering og visualisering. Landdekketklassifisering basert på hyperspektrale bilder og informasjon har tidligere, i stor grad, vært begrenset til data fra satellitter. Oppgaven undersøker muligheten for å utnytte høyoppløselige hyperspektrale data fra fly for mer detaljert kartlegging av arealet.

Abstract

“Land Cover Classification Based on Hyperspectral Imagery and Information”

Land cover classification utilizing hyperspectral imagery and information has been extensively employed, drawing upon data acquired from satellite-borne sensors. The application of land cover classification using high-resolution hyperspectral images has not been as prevalent, owing to issues of accessibility.

This thesis explores the potential and challenges associated with the use of high-resolution hyperspectral images and data for land cover and area classification. Hyperspectral sensors can provide highly detailed information about the Earth’s surface through numerous narrow spectral bands. This opens up new possibilities for mapping and monitoring various types of land cover and materials. Our study delves into the application of hyperspectral data for the general classification of land cover types. Various machine learning methods such as Support Vector Machine and Random Forest are employed to train classification models in Catalyst, based on training data. The results indicate that it is possible to achieve acceptable classification accuracies for these types of land cover. Furthermore, the thesis highlights challenges related to hyperspectral data, such as large data volumes, dimensionality reduction, dataset balancing, and the effects of shadows. Manual analyses of such data are exceedingly demanding, hence the thesis examines the use of available software and tools for efficient processing and visualization. Historically, land cover classification based on hyperspectral imagery and information has largely been confined to satellite data. The thesis investigates the feasibility of utilizing high-resolution hyperspectral data from aircraft for more detailed mapping of the terrain.

Innhold

Forord	i
Sammendrag	ii
Abstract	iii
Innhold	iv
Figurer	vi
Tabeller	vii
1 Innledning	1
2 Teori	3
3 Metode og datagrunnlag	8
3.1 Datasett	10
3.2 Programvare	13
3.3 Databearbeiding	13
3.4 Klassifiserings og analysemetoder	17
4 Resultater	18
4.1 Parallelepiped	19
4.2 K-means	20
4.3 SVM	24
4.4 Random Forest	29
5 Diskusjon	36
5.1 Algoritmer og klassifisering	36
5.2 Veien videre	40
6 Konklusjon	42
Referanser	45
A Vedlegg	46

Figurer

2.1	Support vector machine modell som viser hvordan klassifiseringen blir gjort ved å finne maksimal avstand mellom data for å dele inn i klasser (Larhman, 2018).	4
2.2	Viser oppbygningen til Random Forest(Random trees) basert på flere beslutningstrær (Catalyst.Earth, 2024d).	5
2.3	K-means visualisering fra Javapoint som viser resultat av k-means med klassene fordelt etter sentroider (Javapoint, 2024).	6
2.4	Et eksempel på parallellpipet klassifisering der de som ligger i overlappende bokser blir tildelt ubestemt klasse. Figuren viser det her i et 2d plan	7
3.1	Forenklet modell av prosessen fra rådata til resultat.	8
3.2	Dataflyten, for 16 båndts bildet, fra rådata via bearbeiding i Spyder og analyse i Catalyst.	9
3.3	Dataflyten, for 80 båndts bildet, fra rådata med bearbeiding i Spyder og analyse i Catalyst	10
3.4	Oversiktsbildet over interesseområdet med Bjørkelangen midt i bildet, Øyeren i vest, Lillestrøm i nordvest og grensen til Sverige i øst.	11
3.5	Oversiktsbildet av interesseområdet med Bjørkelangen midt i bildet dato tatt 5.5.2023 (Earth, 2023).	12
3.6	Spektralsignatur av et piksel fra skogområdet med normalfordeling i spektert for levende skog.	14
3.7	Spektralsignatur av piksel som har metning i deler av spekteret (bånd 7 - 43). Verdien er satt til -1 for mettede pikseler.	14
4.1	RGB av området som har blitt klassifisert.	19
4.2	Klassifiseringen gjort med Parallellpipet på et avgrenset område på 16 båndts bildet.	19
4.3	RGB av området som har blitt klassifisert.	20
4.4	Utsnitt av resultatet basert på k-means klassifiseringen gjort på 16 båndts bildet.	21

4.5	Utsnitt av klassifiseringen gjort av k-means på resampled 16 bånd bildet.	22
4.6	Utsnitt av klassifiseringen gjort med k-means basert på det segmenterte 16 båndes bildet.	23
4.7	Utsnitt av klassifiseringen gjort med k-means i det segmenterte 80 båndes bildet.	24
4.8	RGB av området som har blitt klassifisert.	25
4.9	Utsnitt av klassifiseringen gjort med SVM på 16 båndes segmentert bildet.	26
4.10	Utsnitt av klassifiseringen gjort med SVM på segmentert 80 båndes bildet.	28
4.11	RGB av området som har blitt klassifisert.	30
4.12	Utsnitt av klassifiseringen gjort med RF på 16 båndes segmentert bildet. .	31
4.13	Utsnitt av klassifiseringen gjort med RF på segmentert 80 båndes bildet. .	33
4.14	Resultater av hele området fra venstre: SVM 16, SVM 80, RGB, RF 16, RF 80.	35
5.1	viser forskjellen i spektralsignatur i 80 og 16 bånd	39
5.2	Spektral signatur forskjell på vann og skygge på to forskjellige punkter . .	40

Tabeller

3.1	Differanse i båndbredde ($\Delta\lambda$), i nanometer (nm)	16
4.1	Nøyaktighetsstatistikk fra SVM klassifisering av 16 båndes bildet, statistikken er for hele bildet.	27
4.2	Forvirringsmatrise for SVM klassifiseringen av 16 båndes bilde, data fra hele bildet. Referanse data er antall segmenter som er klassifisert.	27
4.3	Nøyaktighetsstatistikk fra SVM klassifisering av 80 båndes bildet, statistikken er for hele bildet.	29
4.4	Forvirringsmatrise for SVM klassifiseringen av 80 båndes bilde, data fra hele bildet. Referanse data er antall segmenter som er klassifisert.	29
4.5	Nøyaktighetsstatistikk fra RF klassifisering på 16 båndes bildet, statistikken er for hele bildet.	32
4.6	Forvirringsmatrise for RF klassifisering på 16 båndes bilde, data fra hele bildet. Referanse data er antall segmenter som er klassifisert.	32
4.7	Nøyaktighetsstatistikk fra RF klassifisering på 80 båndes bildet, statistikken er for hele bildet.	34
4.8	Forvirringsmatrise for RF klassifisering på 80 båndes bilde, data fra hele bildet. Referanse data er antall segmenter som er klassifisert.	34

Kapittel 1

Innledning

Bruken av hyperspektrale- og multispektrale kamera har økt betydelig de siste 20 årene. Tidligere var det hovedsaklig begrenset til spesialiserte anvendelser innen for eksempel geologi og forsvaret. Den teknologiske utviklingen, særlig innen områder som sensorer, prosessorkraft og lagringskapasitet, har ført til at denne typen kamera har blitt mer tilgjengelige og anvendbare for et bredere spekter av brukere og bruksområder. Økt datakraft har muliggjort mer avansert dataanalyse og bildeprosessering av de store data-mengdene som genereres av hyperspektrale og multispektrale kamera. Denne teknologien har vist seg nyttig innen en rekke fagfelt:

- Innen geologi har hyperspektrale kamera blitt brukt til å identifisere ulike mineraler (Kruse, 2012).
- I forsvars- og etterretningssektoren har denne teknologien blitt brukt til innsamling av luftbåren etterretnings-, overvåkings- og rekognoseringsinformasjon (Stein mfl., 2001).
- I presisjonslandbruket har hyperspektrale- og multispektrale kamera vist seg nyttige for å måle egenskaper som fett, protein, modenhet, parasitter og fuktighet i planter og avlinger på en skånsom måte. De brukes også for overvåking av avlingens helsetilstand (Sethy mfl., 2022).
- Atmosfærisk vitenskap bruker denne typen avanserte kamera for blant annet deteksjon av klimagasser som karbondioksid, metan og andre typer luftforurensing (Liu mfl., 2022).

Generelt sett har utviklingen innen bildespektrometri åpnet opp for nye muligheter til å analysere og forstå ulike materialer og fenomener på en mer detaljert og nyansert måte enn hva som er mulig med tradisjonelle kamera. Luftbårne og rombaserte spektrometre inneholder titalls eller hundrevis av smale spektrale bånd som gir nesten kontinuerlige

refleksjonsspektre av jordoverflaten og dens egenskaper. Disse spektrene dekker bølglengdeområdet fra synlig til nærinfrarødt (VNIR) og kortbølgeinfrarødt (SWIR). Mens multispektrale bilder kan brukes til å skille mellom ulike overflatematerialer, gir hyperspektral avbildning enda flere muligheter. Dette inkluderer utnyttelse av spektrale signaturer og absorpsjonsegenskaper hos materialer for å estimere sub-piksel-mengden av materialer, eller for å oppdage spektrale mål av interesse. Denne teknologien muliggjør en langt mer detaljert og nyansert analyse av jordoverflaten sammenlignet med tradisjonelle multispektrale bilder (Wolfe mfl., 2023).

Manuell analyse av hyperspektrale bilder er veldig krevende og ofte umulig på grunn av den store datamengden. Å bearbeide og tolke hver enkelt piksel i et hyperspektralt bilde er en meget tidkrevende prosess. Derfor er 'lavnivåbrukere', som ikke har god kompetanse og erfaring innen hyperspektral bildeanalyse, avhengige av å benytte tilgjengelig programvare for å utføre analysene. Programmer som Catalyst, ENVI og lignende tilbyr brukervennlige grensesnitt og verktøy for å prosessere, analysere og visualisere flerspektrale datasett.

Vår problemstilling har følgende hypotese:

Hvor godt egnet er høyoppløselig hyperspektrale bilder til landdekkklassifisering?

Spørsmål:

- Kan tilgjengelig programvare nyttes til klassifiseringen?
- Kan klassiske maskinlæringsmodeller gi en god klassifisering?

Denne masteroppgaven er satt opp i seks kapitler for å gi en oversiktlig og enkel fremstilling av arbeidet vi har gjort. Det innledende kapitlet gir en introduksjon til temaet og problemstillingen som danner grunnlaget for oppgaven. Her presenteres bakgrunn og målsettinger. I teorikapitlet gjennomgår vi relevant grunnlag for det som kommer senere i oppgaven. Det er lagt vekt på teori som går på databearbeiding, modeller og algoritmer som benyttes. Metodekapitlet beskriver detaljert hvilke metoder, teknikker og verktøy som er benyttet for databearbeiding og analyse. Her redegjøres det for den praktiske tilnærmingen som er valgt for å besvare problemstillingen. Resultatene fra databearbeiding og analysene presenteres i et eget kapittel. Her legges funnene frem på en oversiktlig og etterprøvable måte. I diskusjonskapitlet drøftes resultatene opp mot det teoretiske grunnlaget og tidligere forskning på feltet. Styrker, svakheter og implikasjoner av funnene diskuteres på en kritisk og reflektert måte. Avslutningsvis sammenfatter konklusjonskapitlet de viktigste funnene og konklusjonene som vi får av oppgaven.

Kapittel 2

Teori

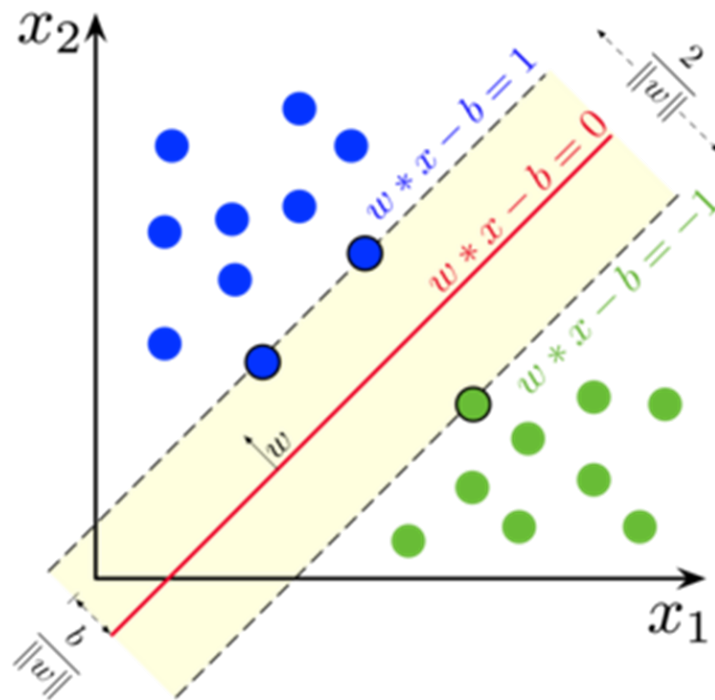
Bearbeiding av rådata fra en sensor er avgjørende for alle fjernmålingsoppgaver. Det innebærer prosessering og bearbeiding for å kunne utnytte data til det prosjektet eller oppgaven som skal gjøres. Prosessering av rådata innebærer blant annet kalibrering, radiometrisk korrigerings, geometrisk korrigerings og avdekking av støy og filtrering av denne. Prosesseringen gir grunnlag for tilpassede data til den oppgaven som skal løses. Hvilke variabler og algoritmer som er tenkt nyttet er avgjørende for hvordan data må formes for å kunne tjene formålet. I de aller fleste sammenhenger vil også statistisk grunnlag for analyse kunne trekkes ut av resultatet fra de forskjellige algoritmer og variabler som blir brukt.

Det er flere metoder som nyttes for landdekkeklassifisering, men tre peker seg ut som mest brukt. Det er Random Forest (RF), Support Vector Machine (SVM) og Spectral Angle Mapper (SAM). Andre metoder er K-means, Nearest Neighbour og Parallelepiped (Christovam mfl., 2019).

Support Vector Machine(SVM) er en av klassifiseringsmetodene som er benyttet (Geomatics, 2024b). Catalyst sin referanse til algoritmen er (Chang & Lin, 2011). SVM er en maskinlærings metode som brukes i styrt klassifisering av høydimensjonale data. Dette er en av de vanligste metodene som benyttes til landdekkeklassifisering (Christovam mfl., 2019).

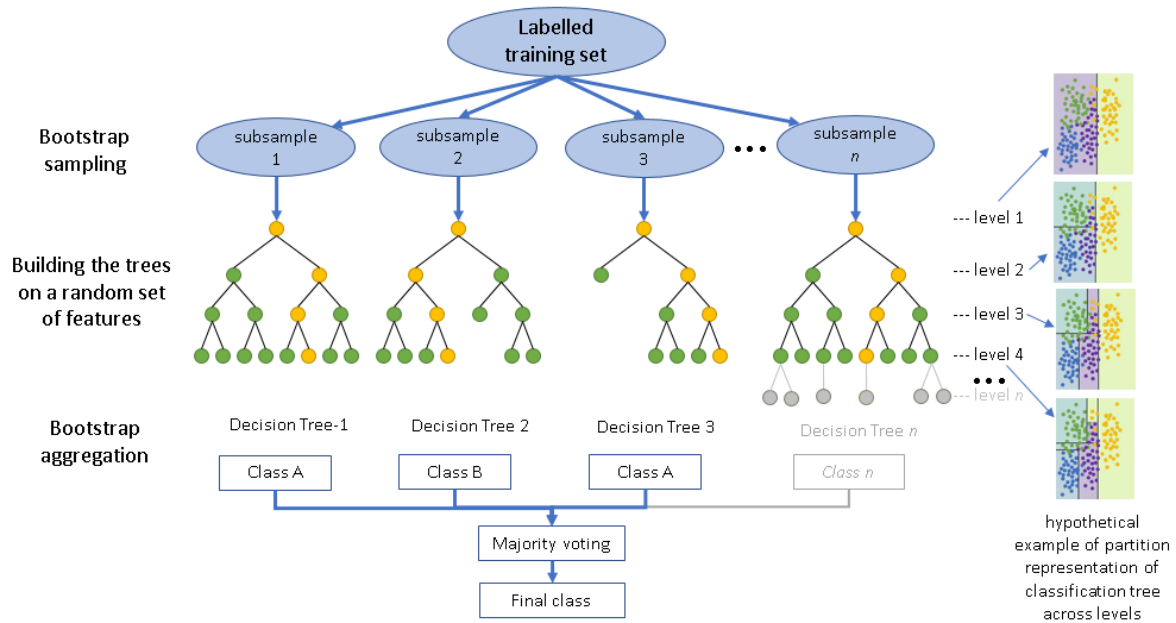
SVM sitt hovedmål er å finne det beste hyperplanet som skiller klassene ved å maksimere avstanden mellom dem. Dette oppnås ved å analysere treningsdata som ligger på kanten av de potensielle klassene, og kalles støttevektorer. Algoritmen ignorerer hovedsakelig treningsdata som ikke kan brukes til disse støttevektorene. Dette resulterer i at det optimale hyperplanet tilpasses med færre treningsdata. Det gjør at SVM oppnår bedre klassifiseringsresultater selv med et mindre sett med treningsdata og er en relativ effektiv metode (Shi & Yang, 2015).

Tilnærmingen til SVM som Catalyst bruker kalles en mot alle. Utdataene er klassen som tilsvarer SVM med den største marginen. Med flerklasser, må den tolke mange (n) hyperplan. Dette krever (n) kvadratisk programmering og optimaliseringsproblemer, hver av hyperplanene skiller en klasse fra de gjenværende klassene (Geomatrics, 2024b).



Figur 2.1: Support vector machine modell som viser hvordan klassifiseringen blir gjort ved å finne maksimal avstand mellom data for å dele inn i klasser (Larhman, 2018).

Random Forest (RF) er den andre styrte klassifiseringsmetoden som vi har benyttet oss av. Random Forest referert som Random Trees i Catalyst, på grunn av varemerkehensyn. Det er en maskinlæringsalgoritme som bygger på beslutningstrær for å utføre klassifisering og regresjon. Denne metoden utnytter en samling av beslutningstrær for å forbedre nøyaktigheten og adressere høy varians som enkelttrær ofte lider av. Hvordan egenskapene til disse trærne blir er mulig å justere med parametere i Catalyst. Der dybden og antall trær er noe som må prøves ut for å se hva som gir best resultat. Antall trær vist som subsample og levels viser dybden til trærne 2.2(Catalyst.Earth, 2024d). Catalyst sin referanse til algoritmen er (docs.opencv.org, 2024).

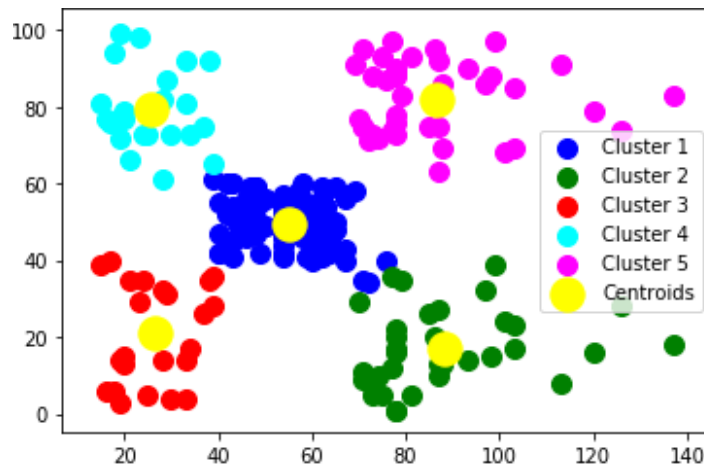


Figur 2.2: Viser oppbygningen til Random Forest(Random trees) basert på flere beslutningstrær (Catalyst.Earth, 2024d).

K-means er en ofte benyttet 'klynge' algoritme. Denne algoritmen er ikke styrt og benytter seg dermed ikke av treningsdata. K-means opererer med å lagre sentroider som den bruker for å definere klynger. En sentroide er et massepunkt basert på avstandene i en samling av punkter. Et punkt anses for å være i en bestemt klynge hvis det er nærmere en klynges sentroide enn en annen klynge sin sentroide. For hvert datapunkt i datasettet, beregnes avstanden til hvert sentroide. Deretter tildeles hvert datapunkt til klyngen med den nærmeste sentroiden.

Målet med algoritmen er å forbedre kvaliteten på den opprinnelige klyngen. Det innebærer at klyngetilhørighet endres for å finne de optimale løsningene. K-means finner de beste sentroidene ved å veksle mellom tildeling av datapunkter til klynger basert på de nåværende sentroidene.

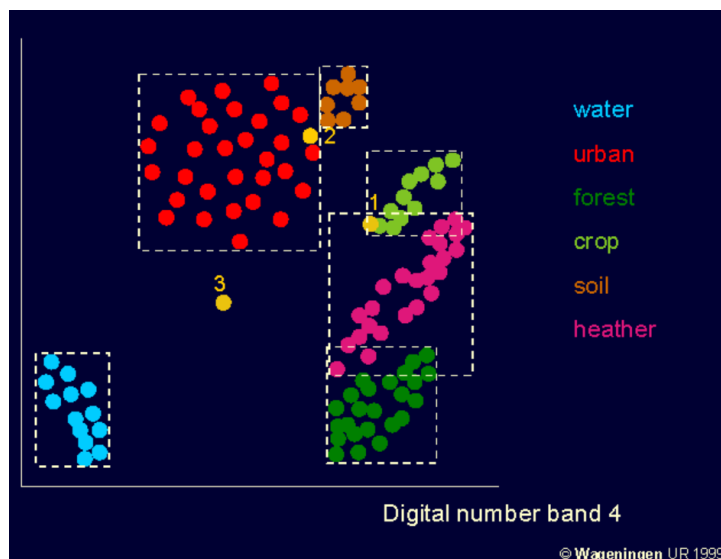
I landdekkklassifisering kan K-means clustering brukes til å gruppere like piksler basert på deres spektrale signaturer. Fordelene med K-means clustering er hvor relativt enkel og effektiv den er som gjør den egnet til å håndtere store datasett, noe som gjør det til et godt valg for landdekkklassifisering med høy oppløsning. Etersom algoritmen baserer seg på spektral signaturen deler den inn i klasser der pikslene som ligger nærmest spektralt, blir en klasse (Catalyst.Earth, 2024a). Derimot kan k-means fortsatt benyttes til klassifisering av landdekk da den gir en indikasjon på hvilke piksler og områder som ligger nærmest hverandre basert på spektralsignatur.



Figur 2.3: K-means visualisering fra Javapoint som viser resultat av k-means med klassene fordelt etter sentroider (Javapoint, 2024).

Parallellpipet klassifisering er en styrt klassifisering basert på treningsdata. Klassifiseringssystemet benytter seg av nedre og øvre grenseverdier, spektral verdier, for å fastslå om en piksel faller innenfor den aktuelle klassen. Disse grenseverdiene spesifiserer dimensjonene, målt i standardavviksenheter, for hver side av et parallellpipet (Leksikon, 2024) som omslutter klassens gjennomsnittsverdi (nv5geospatialsoftware, 2024). Når en piksel befinner seg innenfor dette definerte volumet, blir den tildelt den korresponderende klassen. Dersom en piksel imidlertid overlapper med flere klasser, blir den plassert i en overlappklasse eller ubestemt. Piksler som ikke faller innenfor noen av de definerte klassene, blir tildelt nullklassen, representert ved kode 0.

Bruken av parallellpipet-klassifisering er særlig fordelaktig når det er behov for rask behandling av data. Det er en kjent ulempe med denne metoden at den ofte produserer dårlig nøyaktighet. Dette skyldes ofte at et betydelig antall piksler blir klassifisert i den ubestemte klassen. Denne utfordringen må derfor vurderes når parallellpipet-klassifisering anvendes i en analyse (Catalyst.Earth, 2024b).



Figur 2.4: Et eksempel på parallelliped klassifisering der de som ligger i overlappende bokser blir tildelt ubestemt klasse. Figuren viser det her i et 2d plan (geo-informatie.nl, 1999).

Kapittel 3

Metode og datagrunnlag

I dette kapitlet går vi gjennom hvordan vi har satt opp og organisert oppgaven vår. Det tar for seg bruken av programvare og metoder for analyse. Målet er å gi en god forståelse av hvordan arbeidet er gjennomført og hvilke hjelpemidler vi har disponert. En forenklet modell av dataflyten er vist i figur 3.1. Vi har også benyttet oss av støttevektøy på nett som Ai verktøy som Perplexity.ai og Copilot for GotHub. Kunstigintelligens er brukt som støtte til noe språkvask og som en samtalepartner i diskusjoner rundt problemstillinger og linker til vitenskaplige kilder.

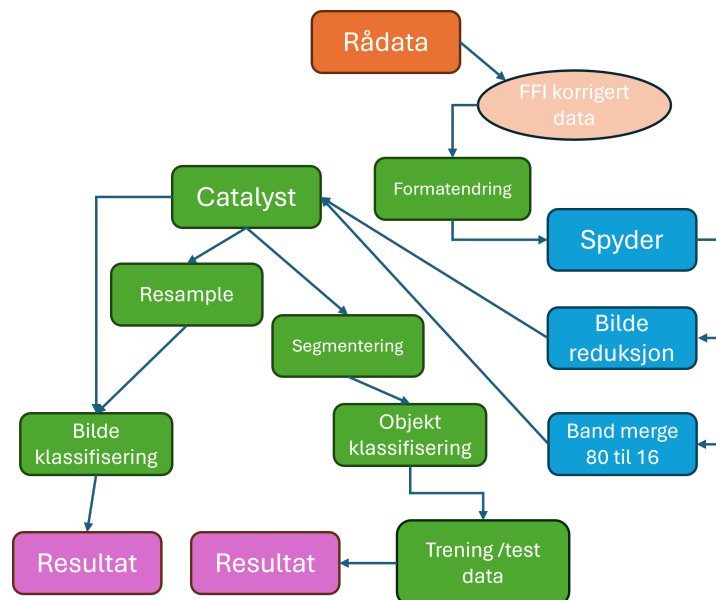


Figur 3.1: Forenklet modell av prosessen fra rådata til resultat.

Rådata fra sensoren er radiometrisk korrigert av Forsvarets Forsknings Institutt (FFI) før de oversendes oss. Vi tar de inn i Catalyst og åpner bildene der. Vi lager et .pix

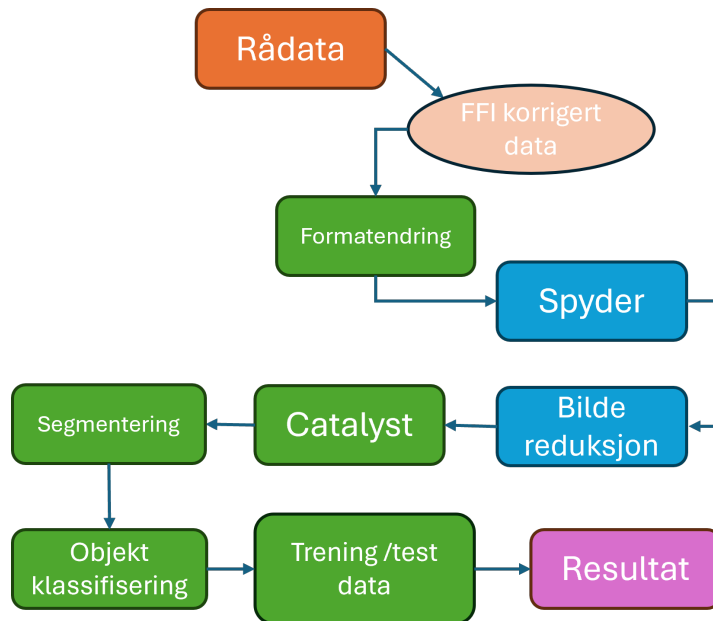
format som brukes i Spyder. Her klippes bildet og spektral binding gjøres før grunnlaget brukes i Catalyst til analyse.

Algoritmer i Catalyst brukes til en pikselbasert og en resampled bildeklassifisering. Et segmentert grunnlag lages for å gjennomføre objektbasert analyse. Vi bruker maskinlæringsalgoritmer til klassifisering basert på trenings og test data. Under visese en modell for hvordan data blir brukt og behandlet steg fro steg. Modellen tar for seg data som fører fram til resultatene basert på 16 båndts bildet.



Figur 3.2: Dataflyten, for 16 båndts bildet, fra rådata via bearbeiding i Spyder og analyse i Catalyst.

Modellen under viser hvordan data blir brukt og behandlet steg fro steg. Modellen tar for seg data som fører fram til resultatene basert på 80 båndts bildet.



Figur 3.3: Dataflyten, for 80 båndts bildet, fra rådata med bearbeiding i Spyder og analyse i Catalyst

3.1 Datasett

Vi har mottatt to datasett fra FFI. Hvert sett består av tre filer med data. Det er ei hovedfil av typen .hyspex som er selve bildefilen. Den andre filen er ei .hdr fil som gjør det mulig å lese .hyspex filen i forskjellige behandlingsprogram, eks Catalyst Pro og Spyder. Den tredje filen er ei .geo fil som gir geografisk referanse til .hyspex filen.

De to datasettene dekker det samme geografiske området, men de har motsatt opptaksretning og litt forskjellig opptakstidspunkt. Denne oppgaven har bare tatt for seg datasett 1 Datasettene er like i filnavn og struktur bare skilt med tallene 22 og 23.

Datasett 1:

- Bjoerkelangen2008-09-24_22_VNIR_1600_SN0005_raw_rad.hyspex
- Bjoerkelangen2008-09-24_22_VNIR_1600_SN0005_raw_rad.hdr
- Bjoerkelangen2008-09-2_22_VNIR_1600_SN0005_raw_rad.geo

Datasett 2:

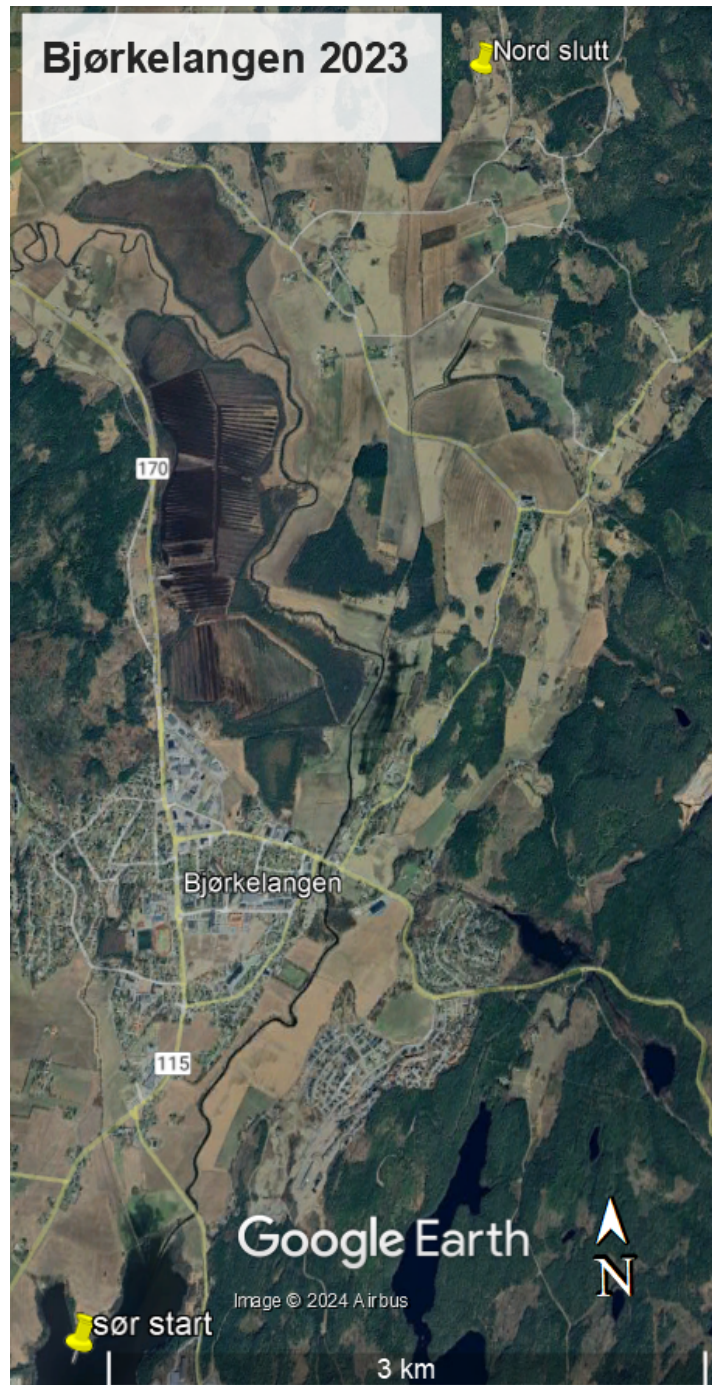
- Bjoerkelangen2008-09-24_23_VNIR_1600_SN0005_raw_rad.hyspex
- Bjoerkelangen2008-09-24_23_VNIR_1600_SN0005_raw_rad.hdr
- Bjoerkelangen2008-09-2_23_VNIR_1600_SN0005_raw_rad.geo

Opptakene er gjort med HySpex VNIR-1600 som er produsert av Norsk Elektro Optikk (elktro optikk, 2024). Det er et hyperspektralt kamera som tar opp bilder med høy oppløsning i spekteret fra 400 nm til 1000 nm. I dette tilfelle er antall bånd redusert til 80. Vanligvis gjøres opptakene med 160 bånd med denne typen kamera. Opptaksmetoden er med fast prisme og en oppdateringshastighet på 100 Hz. Prismet sprer det hvite lyset ut over hele detektoren i kameraet slik at “ett frame” med kameraet avbilder kun en linje på bakken. Fotonene i lyset fra denne linja fordeler seg utover hele detektoren i henhold til fotonenes bølgelengde. Detektorens bredde representerer den fysiske bredden på linja på bakken. Detektorens bredde er delt opp i 1600 pikseler som tar opp informasjon gjennom en spalteåpning på 17 grader på tvers av flyretningen og 0,36 mrad i lengderetningen. GSD er 0,36 mrad X 0,19 mrad (Skauli mfl., 2010).

Interesseområdet i denne oppgaven er rundt tettstedet Bjørkelangen basert på FFI sine datasett med høyoppløselige hyperspektrale bilder av området. Bjørkelangen ligger i Aurskog-Høland kommune i Akershus fylke, ved nordenden av innsjøen Bjørkelangen, langs Lierelva. Avbildet området er variert skog og åkerlandskap med innsjøen Bjørkelangen i sør, spredt bebyggelse og tettstedet Bjørkelangen.



Figur 3.4: Oversiktsbildet over interesseområdet med Bjørkelangen midt i bildet, Øyeren i vest, Lillestrøm i nordvest og grensen til Sverige i øst.



Figur 3.5: Oversiktsbildet av interesseområdet med Bjørkelangen midt i bildet dato tatt 5.5.2023 (Earth, 2023).

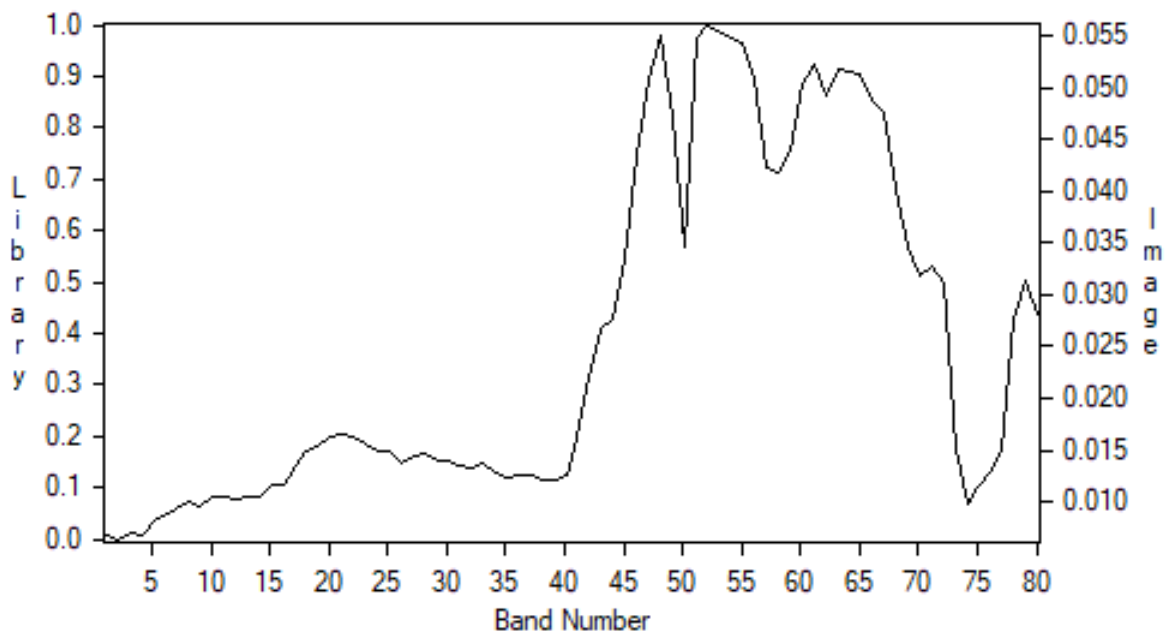
3.2 Programvare

Vi har benyttet oss av programvare som NMBU tilbyr med universitetslisenser eller open source programmer. Vi har brukt Catalyst Pro, fra PCI Geomatics, i det meste av analysen vår. Det har blitt litt programmering i Spyder i forbindelse med bearbeiding av datagrunnlaget. QGIS har blitt benyttet som bearbeidingsverktøy for å få data for analyser i Catalyst. Catalyst pro, er et program som er rette mot bildebehandling og fjernanalyse. Programmet er hovedsaklig laget for multispektrale bilder med en begrensning oppad på 16 bånd. Enkelte algoritmer håndterer flere spektrale bånd, eks; SVM.

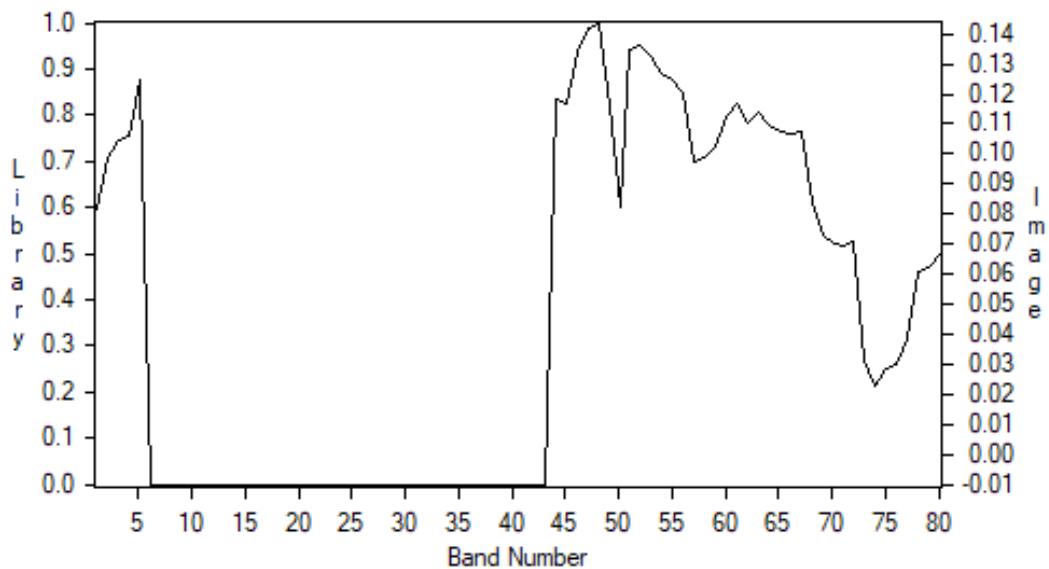
Spyder er en gratis og “opensource” kodeoperator skrevet i Python, for Python, og designet av og for forskere, ingeniører og dataanalytikere. Den har en kombinasjon av avanserte redigerings-, analyse-, feilsøkings- og profileringsfunksjonaliteter til et omfattende utviklingsverktøy med datautforskning, interaktiv utførelse, inspeksjon og visualiseringsfunksjoner i en vitenskapelig ramme (IDE, 2023).

3.3 Databearbeiding

Bearbeiding av data er viktig i alle sammenhenger der vi skal gjøre noe annet enn å se på dem. Før denne prosessen starter må vi som skal behandle data være veldig sikre på hva vi er på jakt etter og hvordan data bør se ut for å oppnå det vi søker. Mer informasjon gir ikke alltid bedre resultat. Vi kan fort gjøre den vanlige feilen å stole for mye på applikasjonen for å produsere et raskt svar. Skjulte prosesser i applikasjonen har stor betydning for resultatet, en annen faktor er begrensningene i dataene. Ofte er resultatene ikke det du forventer på grunn av feil forutsetninger og manglende kunnskap om applikasjonen, innsamlingsplattformen og dataene. Når det gjelder bildebehandling er det ofte pikseler som ikke har normale verdier i det elektromagnetiske spekteret. Disse pikslene må det gjøres noe med. De kan gis en fiktiv verdi så de enkelt kan kategoriseres, eller skilles ut på annet vis. Figur 3.6 og 3.7 viser dette for et piksel med normalfordeling og et med metning i spekteret for synlig lys.



Figur 3.6: Spektralsignatur av et piksel fra skogområdet med normalfordeling i spektret for levende skog.



Figur 3.7: Spektralsignatur av piksel som har metning i deler av spektret (bånd 7 - 43). Verdien er satt til -1 for mettede pikseler.

Datasettene fra FFI var radiansjustert (gsp.humboldt.edu, 2020) når vi fikk de. Disse filene er et ENVI-format “hyspex” som Norsk Elektro Optikk (NEO) bruker i sine kamera. For å kunne bruke filen i Catalyst konverterte vi de til .pix format. .pix er Catalyst sitt eget format.

Catalyst er bygd opp slik at bildeanalyse er delt inn i to metoder: bildeklassifisering og objektklassifisering. Hovedforskjellen mellom disse ligger i deres tilnærming til klassifiseringsprosessen, og hvilke klassifiseringsalgoritmer som blir gjort tilgjengelig i de forskjellige metodene.

Bildeklassifisering er en pikselbasert klassifiseringsmetode. Denne metoden involverer algoritmer som arbeider direkte med pikslene i bildet. På den annen side, er objektklassifisering avhengig av en forutgående segmenteringsprosess. Denne segmenteringen tar utgangspunkt i pikslene og deres verdier, som vi kan tilpasse og påvirke. Dette inkluderer også å bestemme hvilke bånd som skal brukes i segmenteringsprosessen.

Det er viktig å merke seg at valget av segmenteringsparametere sterkt påvirker sammenligningsgrunnlaget mellom de to klassifiseringsmetodene. Dermed, selv om begge metodene bruker piksler som grunnlag for klassifisering, er deres tilnærminger og resultater forskjellige på grunn av hvordan grunnlaget for klassifiseringa blir endret (Catalyst.Earth, 2024c).

Support Vector Machine og Random Forest (random trees i Catalyst) baserer seg dermed ikke på en enkelt piksel for analyse, men en homogen gruppe av piksler og blir til bildeobjekter. Hvordan disse gruppene/segmentene oppstår kan justeres ved å endre parametere i Catalyst.

Vi har tilpasset segmenteringen til å passe best mulig ved å justere parameterne “scale”, “shape” og “compactness”. Disse parameterne og hvordan resultatet blir må testes ut på bildene ved å justere forskjellige verdier for å se hva som passer best til det enkelte bildet. “Scale” referer til størrelsen til objektene, “shape” referer til hvordan mye den skal vekte farge og compactnes påvirker grensene. I alle segmenteringene som er utført er vektning av farge i “shape” justert for å tilpasse spektral signaturen i segmentene, målet er å få de så like som mulig.

Etter segmenteringen og før treningsdata velger vi hva slags verdier du vil ta med i klassifiseringen under “Attribute calculation”. Her velges piksel statistikk i bildet og så legger vi til NDVI (Normalized Difference Vegetation index) og SAVI (soil adjusted vegetation index).

Videre lager vi treningsdata av segmentene og lager en fasit basert på hva som er forventet resultat på klassifiseringen. I utvalg av trenings segmenter og fasit er dette selektert ved å velge segmenter som er mest mulig rene/ensartede innefor klassen. Vi har også

eksperimenter med skygger som egen klasse men endte med, pga av resultater, å utelate skygge segmenter. Segmenter med høy usikkerhet, om hva klassen inneholdt, er ikke brukt som treningsdata eller til verifisering av klassifiseringen.

Siden Catalyst har en grense oppad på 16 spektrale bånd og våre data hadde 80, prøvde vi å redusere antall bånd ved sammenslåing. Vi lagde en kode i Spyder som tok inn .hdr og .img (konvertert .hyspex) filene. Sammenslåing av data kan gjøres på forskjellige måter. Vi så på hvordan vi best kunne bevare mest mulig “riktig” data og samtidig redusere den. Etter en sammenligning av båndbreddene i datagrunnlaget fant vi at en midling av informasjonen ville gi et godt resultat for vår oppgave og analyse. Sammen slått og midlet informasjon gav oss et nytt bilde med 16 bånd og redusert til ca 1/5 av datamengden.

Tabell 3.1: Differanse i båndbredde ($\Delta\lambda$), i nanometer (nm)

Maksimum $\Delta\lambda$ differanse for orginaldata	7,2803 nm
Minimum $\Delta\lambda$ differanse for orginaldata	7,1981 nm
Gjennomsnitt $\Delta\lambda$ differanse for orginaldata	7,2516 nm
Maksimum $\Delta\lambda$ differanse for fembånds gruppene	0,0608 nm
Minimum $\Delta\lambda$ differanse for fembånds gruppene	0,0000 nm
Gjennomsnittlig $\Delta\lambda$ differanse for fembåndsgruppene	0,0159 nm

Vurderingsgrunnlaget for sammenslåing og midling av båndinformasjon er at vi tok utgangspunkt i oppgitte bølgelender for bildet med 80 bånd og hentet ut differansen i båndbredde, “ $\Delta\lambda$ ” . Deretter ble maks og min verdiene funnet for differansene og den gjennomsnittlige differansen for alle båndene. Deretter ble maks og min differansen for 5 og 5 bånd tatt ut. Differansen for maks og min verdi til 5 båndsgroppene ble brukt som grunnlag for vurdering 3.1. Vi vurderte differansen så lav at det ikke ville påvirke vår bruk. For å få helt rett verdier måtte vi ha vektet sammenslåingen basert på “ $\Delta\lambda$ ” for hvert av båndene. I vår analyse anser vi ikke det som nødvendig.

Resampling ved bruk av “resamp” algoritmen i Catalyst ble gjennomført slik at vi fikk resultater med endret romligopløsning (GSD) (Catalyst.Earth, 2024e). GSD endringen som kom fra “resamp” algoritmen baserer seg på nearest neighbour (Geomatics, 2024a). Det er ikke en optimal måte å resample på, men vi valgte å ta det med da resultatet er interessant i vurderingen av arbeidet. Vi har også gjort et forsøk på og resample bildet med koding i Spyder, men har ikke fått grunnlag godt nok for å kunne bruke det. En evt resampling ville ha blitt gjort med gjennomsnittsverdier. Dette hadde sannsynligvis gitt et annet bilde en det som kommer fram fra Catalyst.

3.4 Klassifiserings og analysemetoder

K-means klassifiseringen ble utført på et 16 bånd bilde, både for de små bildene og hele bildet. På grunn av måten K-means fordeler til klasser/klynger, ble det bestemt å bruke 7 klynger. Dette er en mer klasse enn det som har vært benyttet i de andre klassifiseringene, dette fordi K-means prøver å unngå å klassifisere noe til den siste klassen. Dette er for å sikre at alt blir klassifisert og ikke blir tildelt den gjenværende klassen.

Det ble også utført styrt klassifisering med Parallelepiped. Her ble treningsgrunnlaget valgt ut på samme måte, og med de samme prinsippene som ved segmentering nevt tidligere. Forskjellen er at her blir områder markert og pikslene innad blir brukt som treningsdata.

Det er mulighet for å justere “bias” og “terskel”. Terskel er et relativt mål som brukes til å endre terskelverdiene. Det gjør at sjansen for at piksler blir klassifisert i mer enn en klasse reduseres. Bias er en verdi fra 0 til 1, der høyere verdi veier en klasse foran en annen. Det kan også brukes til å løse overlapp mellom klasser. Begge verdiene ble justert, men det ble til slutt bestemt å benytte standardinnstillingene (Catalyst.Earth, 2024f, bias and thershold).

K-means-algoritmen ble anvendt på 16 og 80 bånd i forskjellige formater, inkludert hele bildet, et utklipp og et resampled bilde. Selv om prosedyrene var identiske for alle, varierte segmenteringen, som klassifiseringen er basert på, fra bilde til bilde. Utgangspunktet for dette arbeidet var den samme tankegangen som ved bildeklassifisering med hensyn til klynger og klasser. Derfor ble det valgt 7 klynger, en mer enn det som tidligere har vært benyttet som klasser. På grunn av mindre variasjon av forskjellige klasser i det lille bildet, ble antallet klynger redusert. Dette er noe som må tilpasses da dette er noe som sterkt påvirker resultatet. For få klasser og det blir flere klasser innad i en klasse og for mange og en klasse kan deles opp for mye og gi en ikke menigfull klassifisering.

Styrt klassifisering ble også utført på de samme bildene som den ikke-styrte klassifiseringen, med 16 og 80 bånd på hele bildet, utsnitt og resampled.

Support Vector Machine (SVM) ble kjørt med standard innstillingene, ettersom det ikke er noen parametere som kan justeres.

Random Forest algoritmen ble testet med forskjellig antall trær og dybder. Det ble til slutt bestemt å redusere dybden på trærne til fem, og det viste seg at det beste resultatet ble oppnådd med rundt 100 trær på 16 bånd. Forskjellen ble derimot med 80 bånd bildet, hvor det var nødvendig å øke antallet trær for å forbedre resultatet. Endelig antall trær ble 2000 trær etter testing av forskjellige verdier. Dybden ble den samme på begge, med fem i dybden. Dette kom etter at forskjellige dybder ble testet.

Kapittel 4

Resultater

I resultatkapitlet presenteres klassifiseringsresultatene for hver av metodene beskrevet i metodekapitlet. For hver metode vises klassifiseringskartet sammen med statistikk for analysen. Catalyst ha ikke god statistikk for alle metodene. For flere metoder er en feilmatrix og ulike statistiske mål på klassifiseringsnøyaktighet presentert. Feilmatrixen viser antall piksler som er korrekt eller feilklassifisert for hver klasse. Basert på denne kan overordnede nøyaktighetstall som total nøyaktighet og Kappa koeffisienten beregnes. Disse målene gir en indikasjon på den generelle klassifiseringsnøyaktigheten for hver metode. I tillegg benyttes klassespesifikke mål som produsentens og brukerens nøyaktighet for å vurdere hvor godt de enkelte klassene er gjenkjent. Dette er nyttig for å identifisere spesifikke utfordringer og svakheter ved metodene (gsp.humboldt.edu, 2024).

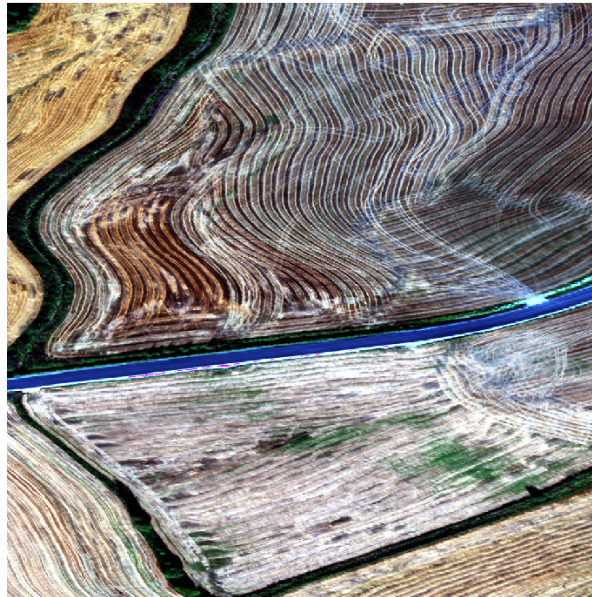
I denne oppgaven er bildeutsnittet og bildeoppsettene valgt for å gi et best mulig grunnlag for sammenligning og evaluering av de ulike klassifiseringsmetodene. Bildeutsnittet er valgt for å representere et tilstrekkelig stort og variert område, slik at det inneholder eksempler på de fleste arealtypene som skal klassifiseres. Dette gir mulighet til å vurdere metodenes evne til å skille mellom ulike klasser. Samtidig er bildeutsnittet av en slik størrelse at detaljnivå kommer fram og vi får en god visualisering.

Liste over data grunnlaget i hver figur:

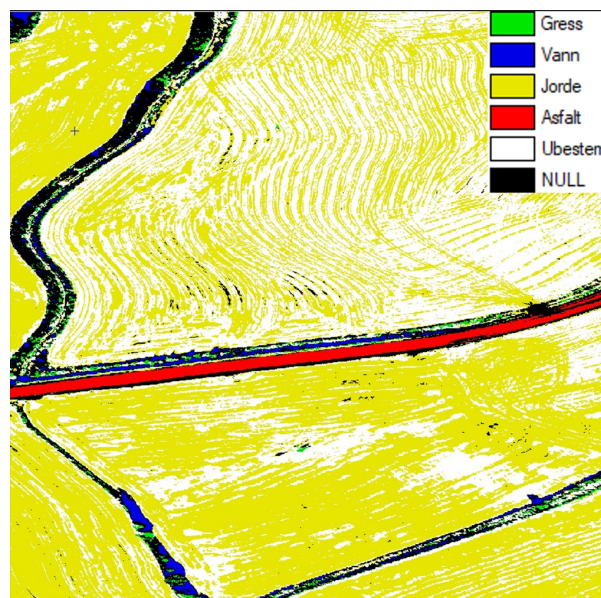
- 16 båndets bildet: ?? 4.4 4.9 4.12
- 16 båndets bildet resampled: 4.6
- 80 båndets bildet: 4.7 4.10 4.13

4.1 Parallelepiped

Det ble gjennomført en styrt klassifisering med parallelepiped, dette for å teste hvilke områder som var utfordrene å klassifisere. Dette er et utsnitt som vi testet på. Resultatet her ble svært bra. Det ble også gjort et forsøk med parallelepiped på hele bildet, men på grunn av store variasjoner i kartet så returnerte parallelepiped omtrent hele bildet som konflikt eller ubestemt.



Figur 4.1: RGB av området som har blitt klassifisert.



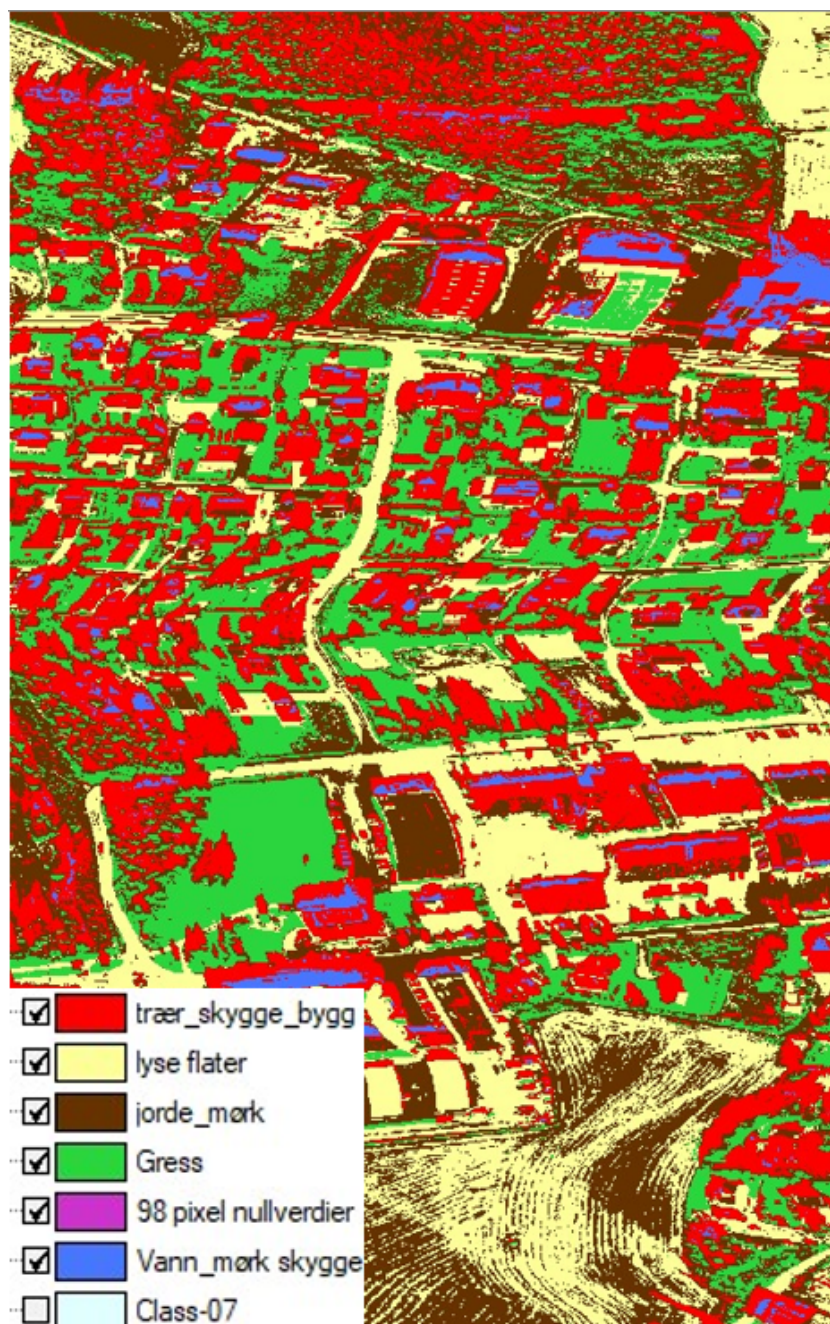
Figur 4.2: Klassifiseringen gjort med Parallelepiped på et avgrenset område på 16 båndes bildet.

4.2 K-means

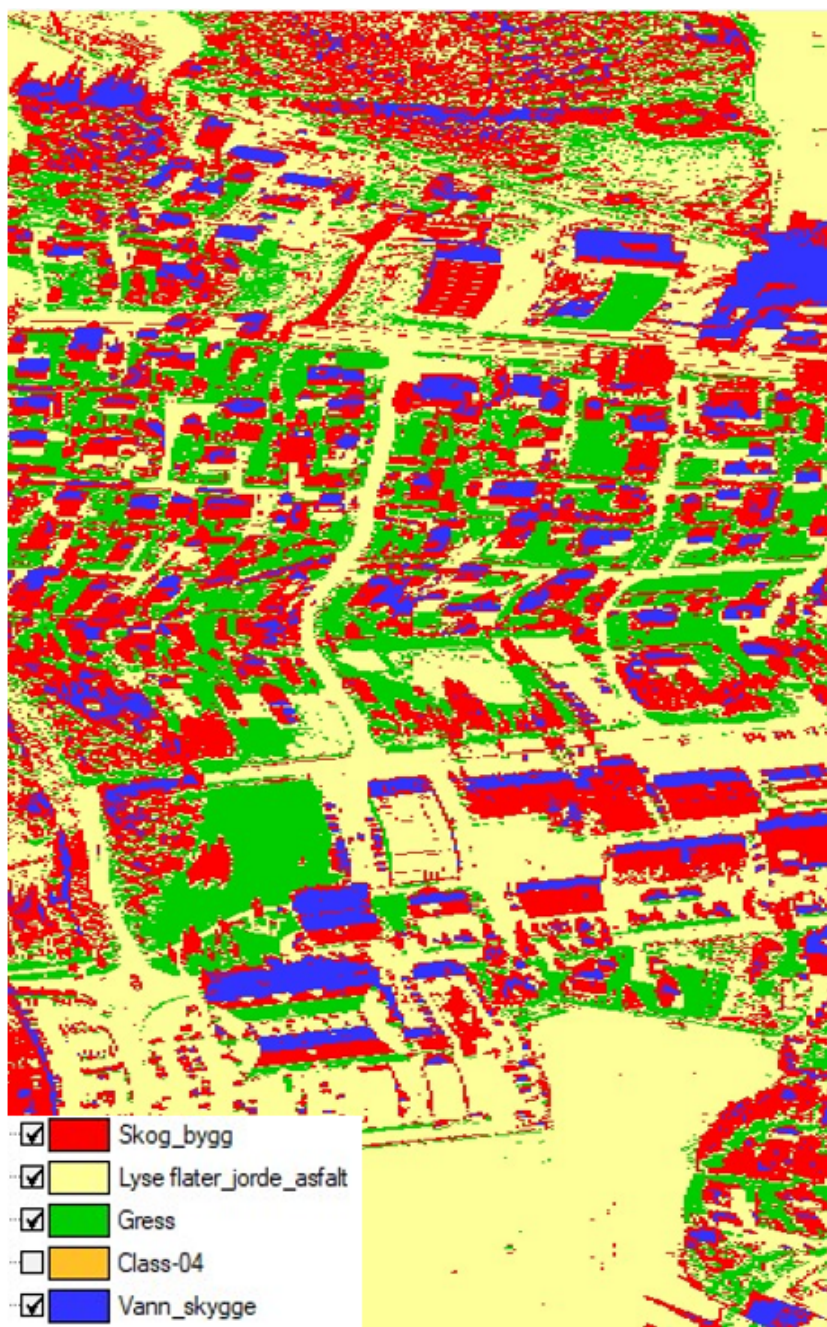
Relativt bra resultater med k-means basert på piksler med stor enighet mellom 16-bånd 4.4 bildet og resampeled 4.5. Som forventet blir resampelede bildet mere homogent. Resultatene med segmentering viser litt forskjell mellom 80 og 16. Her påvirker segmentering grunnlaget for klassifiseringen og klassifisering til forskjellige klynger viser litt forskjell mellom klassene, men innad i klassene er det stor likhet.



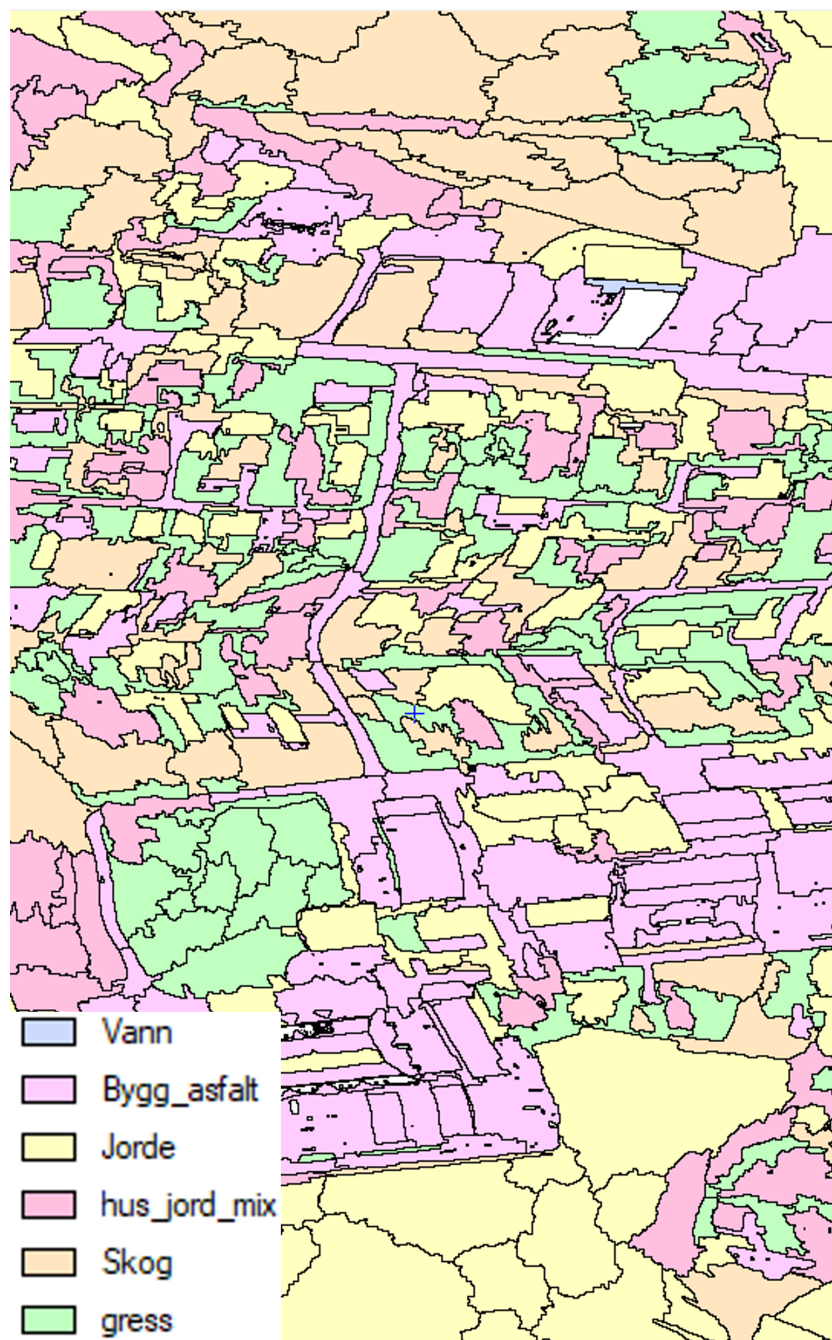
Figur 4.3: RGB av området som har blitt klassifisert.



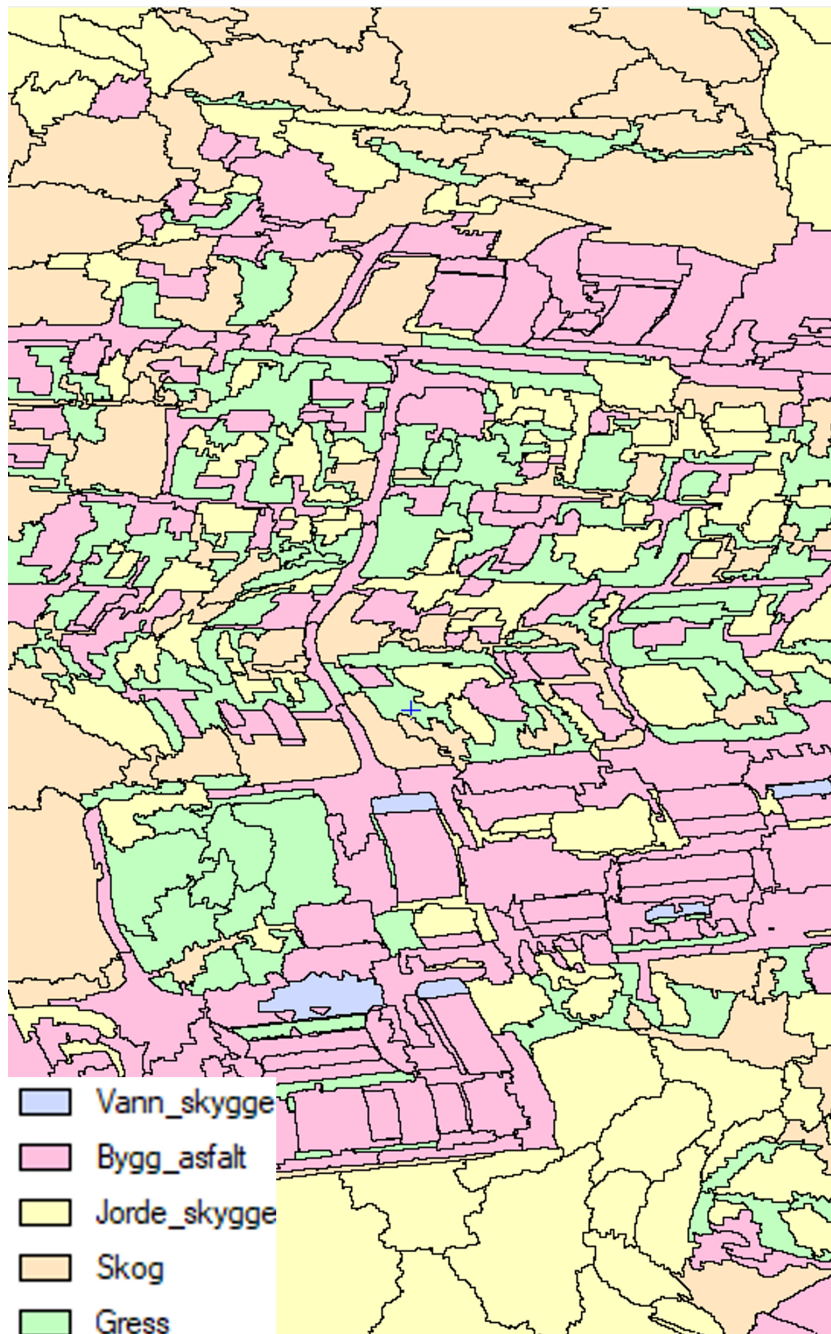
Figur 4.4: Utsnitt av resultatet basert på k-means klassifiseringen gjort på 16 båndsbildet.



Figur 4.5: Utsnitt av klassifiseringen gjort av k-means på resampled 16 bånd bildet.



Figur 4.6: Utsnitt av klassifiseringen gjort med k-means basert på det segmenterte 16 båndets bildet.



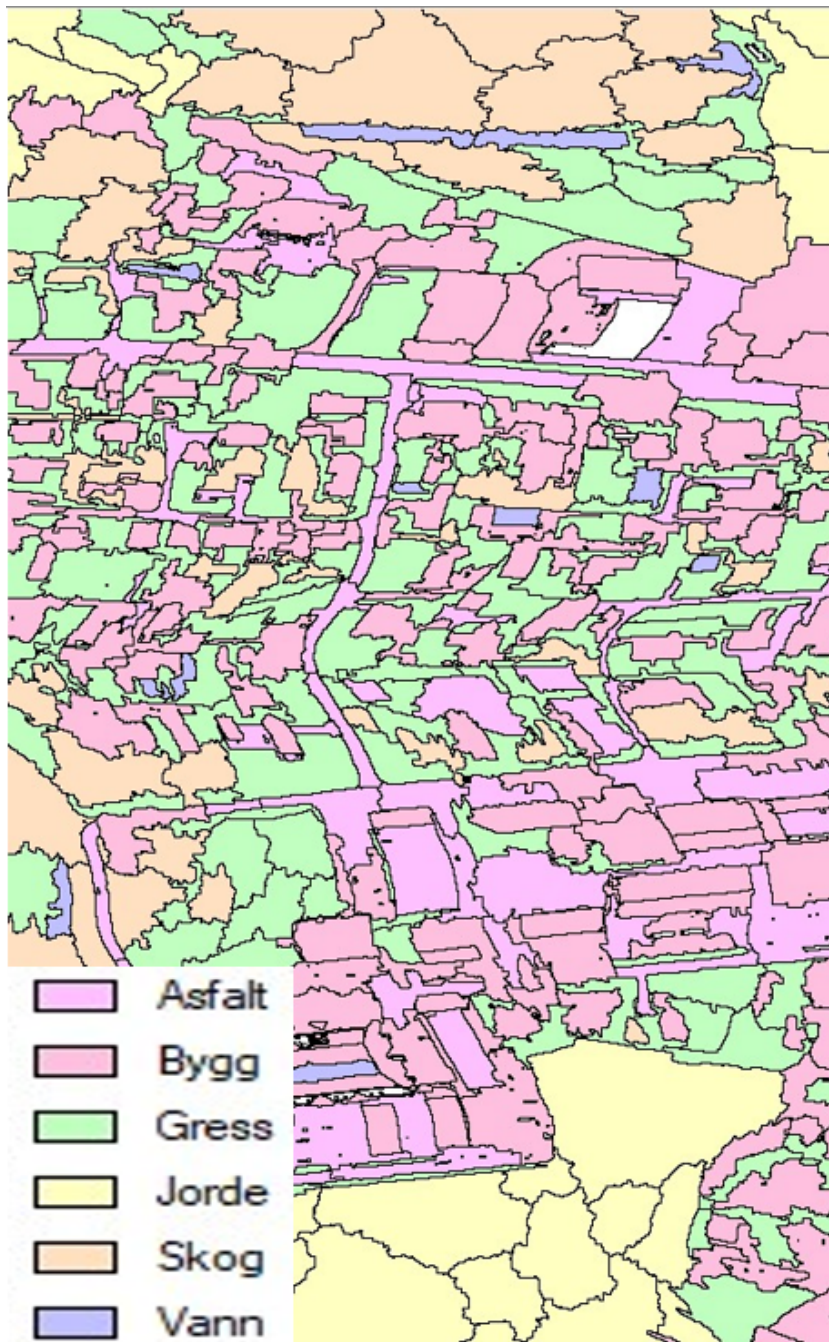
Figur 4.7: Utsnitt av klassifiseringen gjort med k-means i det segmenterte 80 båndes bildet.

4.3 SVM

Support Vector Machine er der vi har fått best resultater og muligheten til å hente ut statistikk i form av statistikk og en forvirringsmatrise. Bra resultater fra begge men ser at i 80-båndes bildet har hele asfaltklassen blitt tildelt byggklassen. Dette er noe som drar ned prosentnøyaktigheten betraktelig.



Figur 4.8: RGB av området som har blitt klassifisert.



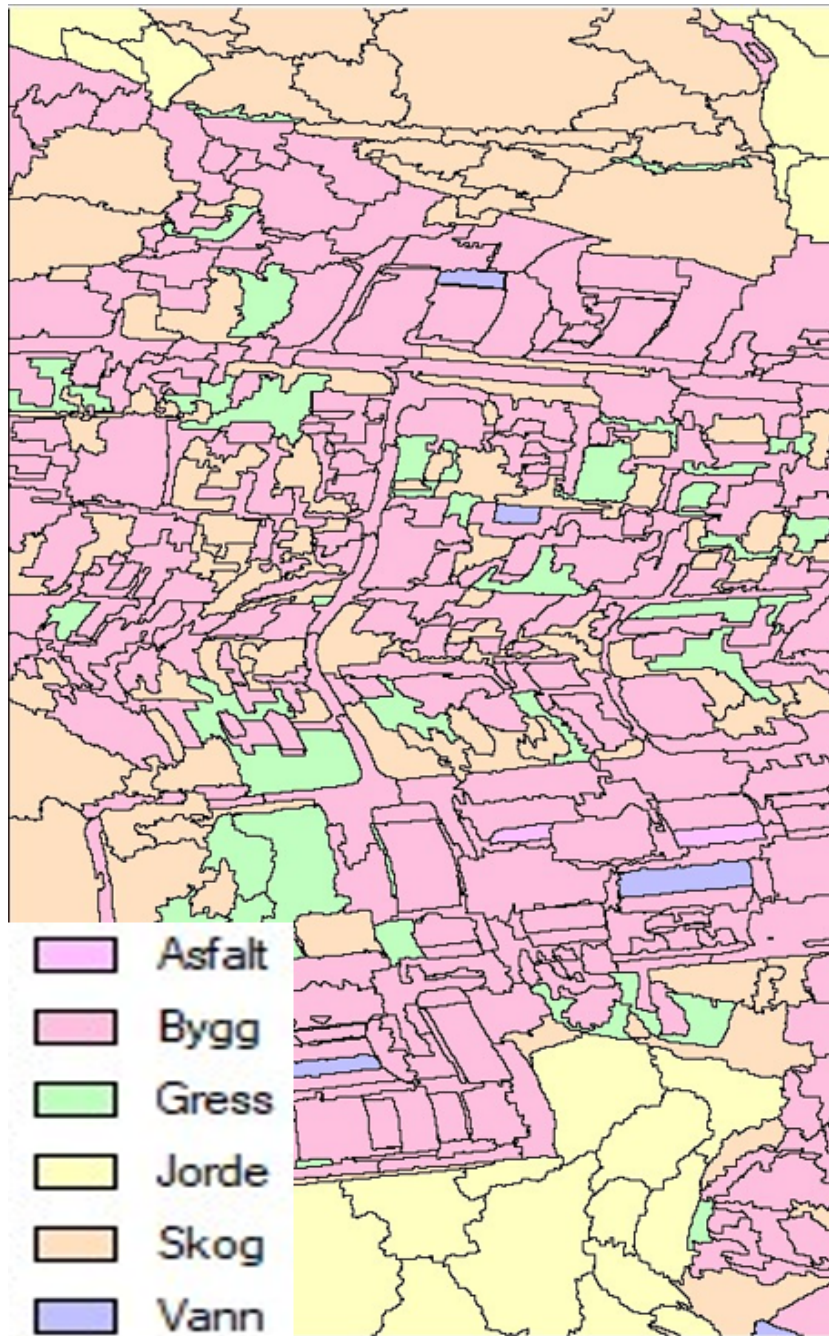
Figur 4.9: Utsnitt av klassifiseringen gjort med SVM på 16 bånders segmentert bildet.

Tabell 4.1: Nøyaktighetsstatistikk fra SVM klassifisering av 16 båndes bildet, statistikken er for hele bildet.

Overall Accuracy		87.50 %	
Overall Kappa Statistic		0.840	
Quantity Disagreement		7.23 %	
Allocation Disagreement		5.27 %	
Class name	Producer's Accuracy	User's Accuracy	Kappa Statistic
Skog	81.08 %	91.46 %	0.8896
Jorde	88.57 %	99.60 %	0.9939
Gras	95.48 %	77.08 %	0.7171
Asfalt	90.00 %	95.75 %	0.9515
Bygg	85.71 %	61.54 %	0.5870
Vann	75.00 %	76.92 %	0.7573

Tabell 4.2: Forvirringsmatrise for SVM klassifiseringen av 16 båndes bilde, data fra hele bildet. Referanse data er antall segmenter som er klassifisert.

Klassifiserings data	Referanse data						
	Skog	Jorde	Gress	Asfalt	Bygg	Vann	Totalt
Skog	150	1	5	0	1	7	164
Jorde	0	248	1	0	0	0	249
Gress	29	14	148	0	1	0	192
Asfalt	0	0	0	90	4	0	94
Bygg	0	17	0	10	48	3	78
Vann	6	0	1	0	2	30	39
Ukjent	0	0	0	0	0	0	0
Totalt	185	280	155	100	56	40	



Figur 4.10: Utsnitt av klassifiseringen gjort med SVM på segmentert 80 båndes bildet.

Tabell 4.3: Nøyaktighetsstatistikk fra SVM klassifisering av 80 båndts bildet, statistikken er for hele bildet.

Overall Accuracy		68.52 %	
Overall Kappa Statistic		0.607	
Quantity Disagreement		28.81 %	
Allocation Disagreement		2.66 %	
Class name	Producer's Accuracy	User's Accuracy	Kappa Statistic
Skog	97.40 %	70.09 %	0.6324
Jorde	91.70 %	96.95 %	0.9541
Gras	43.94 %	100.0 %	1.0000
Asfalt	1.80 %	18.18 %	0.0548
Bygg	97.92 %	20.80 %	0.1591
Vann	68.42 %	100.0 %	1.0000

Tabell 4.4: Forvirringsmatrise for SVM klassifiseringen av 80 båndts bilde, data fra hele bildet. Referanse data er antall segmenter som er klassifisert.

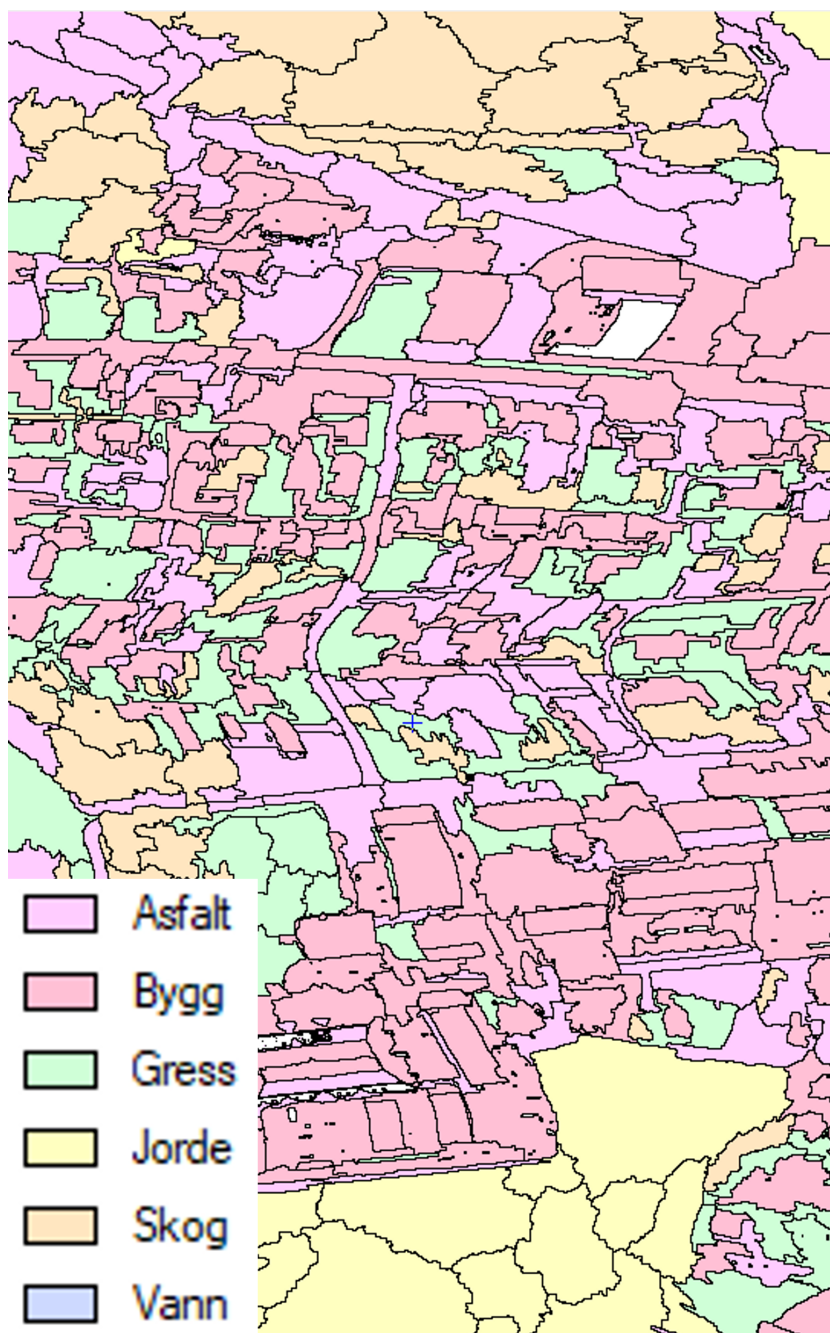
Klassifiserings data	Referanse data						
	Skog	Jorde	Gress	Asfalt	Bygg	Vann	Totalt
Skog	150	0	52	0	0	12	214
Jorde	0	254	8	0	0	0	262
Gress	0	0	87	0	0	0	87
Asfalt	0	8	0	2	1	0	11
Bygg	4	15	51	109	47	0	226
Vann	0	0	0	0	0	26	26
Ukjent	0	0	0	0	0	0	0
Totalt	154	277	198	111	48	38	

4.4 Random Forest

Random Forest resultatene er lavere enn SVM. Vi hadde forventet et bedre resultat sammenlignet med tidligere forskning, 16 båndts bildet gir bedre resultater enn 80 båndts bildet. Asfalt klassen i 80 båndts bildet har blitt klassifisert i bygg klassen på samme måte som i SVM. Det drar ned prosent nøyaktigheten mye.



Figur 4.11: RGB av området som har blitt klassifisert.



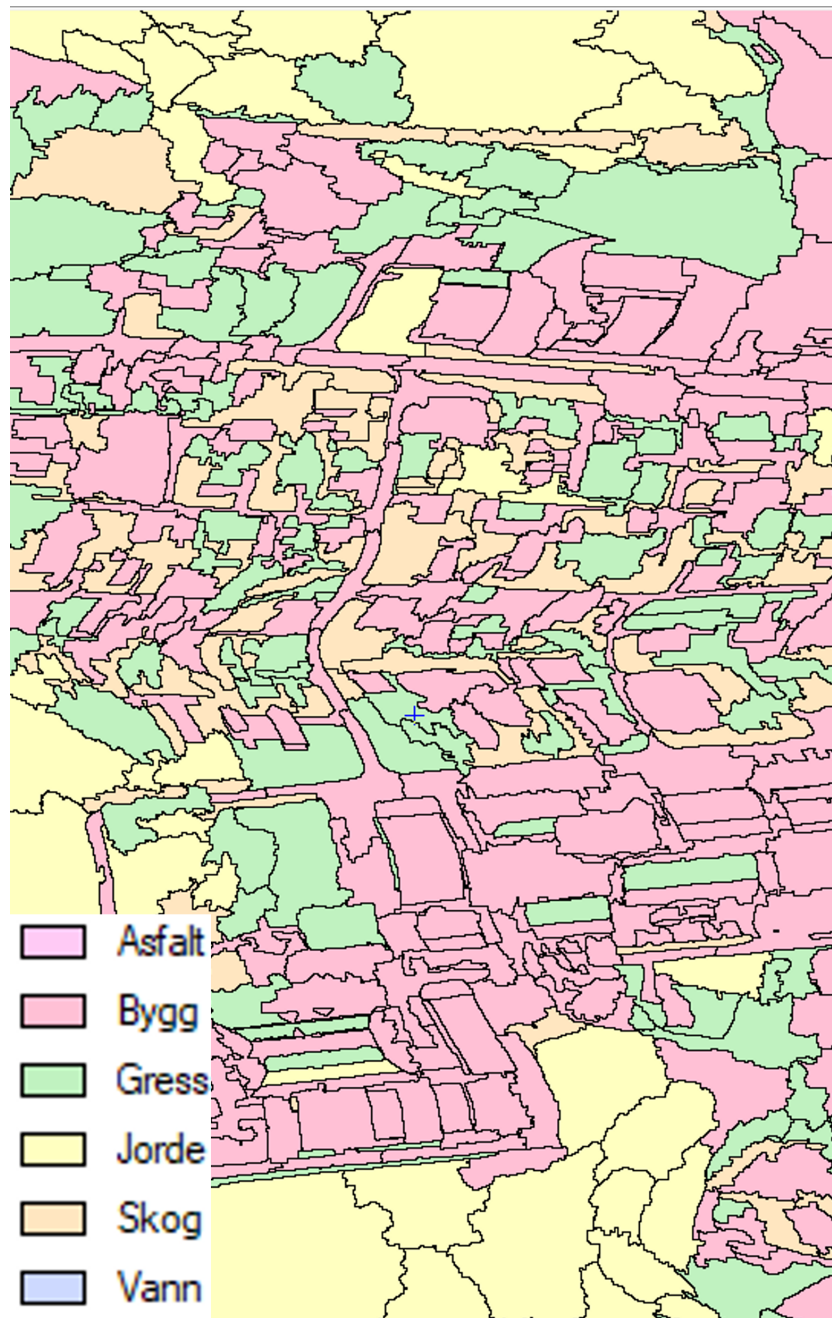
Figur 4.12: Utsnitt av klassifiseringen gjort med RF på 16 bånders segmentert bildet.

Tabell 4.5: Nøyaktighetsstatistikk fra RF klassifisering på 16 båndes bildet, statistikken er for hele bildet.

Overall Accuracy		57.60 %	
Overall Kappa Statistic		0.481	
Quantity Disagreement		27.08 %	
Allocation Disagreement		15.32 %	
Class name	Producer's Accuracy	User's Accuracy	Kappa Statistic
Skog	77.30 %	93.46 %	0.9155
Jorde	53.57 %	98.68 %	0.9800
Gras	47.74 %	67.27 %	0.5960
Asfalt	47.00 %	17.74 %	0.0625
Bygg	80.36 %	40.18 %	0.3577
Vann	27.50 %	45.83 %	0.4304

Tabell 4.6: Forvirringsmatrise for RF klassifisering på 16 båndes bilde, data fra hele bildet. Referanse data er antall segmenter som er klassifisert.

Klassifiserings data	Referanse data						
	Skog	Jorde	Gress	Asfalt	Bygg	Vann	Totalt
Skog	143	0	1	0	0	9	153
Jorde	0	150	0	1	0	1	152
Gress	18	0	74	1	3	14	110
Asfalt	11	130	66	47	8	3	265
Bygg	0	0	14	51	45	2	112
Vann	13	0	0	0	0	11	24
Ukjent	0	0	0	0	0	0	0
Totalt	185	280	155	100	56	40	



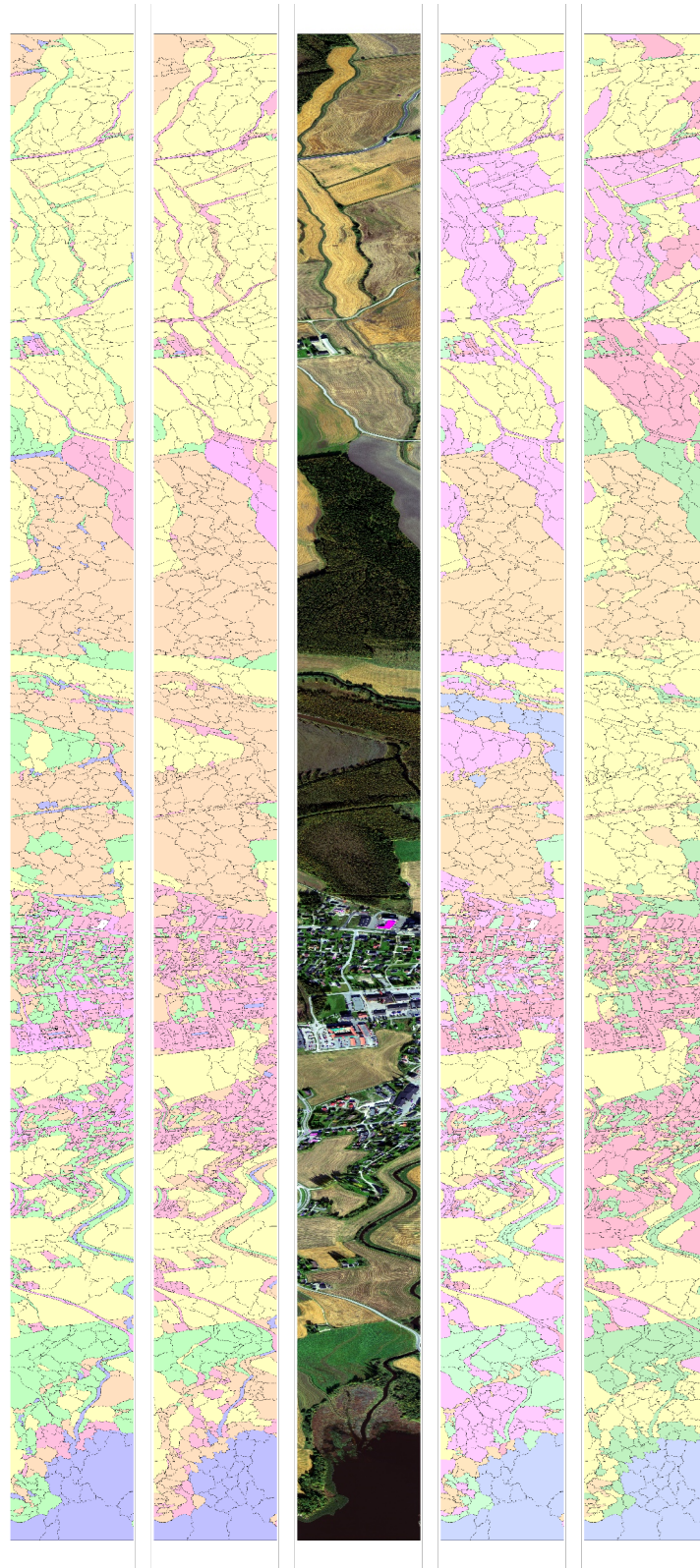
Figur 4.13: Utsnitt av klassifiseringen gjort med RF på segmentert 80 båndes bildet.

Tabell 4.7: Nøyaktighetsstatistikk fra RF klassifisering på 80 båndes bildet, statistikken er for hele bildet.

Overall Accuracy		38.86 %	
Overall Kappa Statistic		0.237	
Quantity Disagreement		24.82 %	
Allocation Disagreement		36.32 %	
Class name	Producer's Accuracy	User's Accuracy	Kappa Statistic
Skog	29.87 %	50.00 %	0.3854
Jorde	43.68 %	45.49 %	0.1798
Gras	47.98 %	65.52 %	0.5465
Asfalt	0.90 %	1.79 %	-0.1346
Bygg	91.67 %	17.39 %	0.1229
Vann	36.84 %	100.00 %	1.0000

Tabell 4.8: Forvirringsmatrise for RF klassifisering på 80 båndes bilde, data fra hele bildet. Referanse data er antall segmenter som er klassifisert.

Klassifiserings data	Referanse data						
	Skog	Jorde	Gress	Asfalt	Bygg	Vann	Totalt
Skog	46	0	45	0	1	0	92
Jorde	97	121	46	0	1	1	266
Gress	10	15	95	0	2	23	145
Asfalt	1	53	1	1	0	0	56
Bygg	0	88	11	110	44	0	253
Vann	0	0	0	0	0	14	14
Ukjent	0	0	0	0	0	0	0
Totalt	154	277	198	111	48	38	



Figur 4.14: Resultater av hele området fra venstre: SVM 16, SVM 80, RGB, RF 16, RF 80.

Kapittel 5

Diskusjon

I dette kapitlet vil vi diskutere og evaluere metodene og resultatene som ble presentert i de foregående kapitlene. Vi vil sammenligne de ulike tilnærmingene og vurdere deres styrker og svakheter i lys av den teoretiske bakgrunnen og forventningene vi hadde på forhånd. Gjennom analysen av resultatene vil vi identifisere hvilke metoder som ga mest presise og pålitelige klassifiseringer, og hvilke som hadde større utfordringer. Vi vil drøfte mulige årsaker til avvik mellom forventede og faktiske resultater, og vurdere hvorvidt disse kan forklares ut fra de teoretiske prinsippene eller om det er andre faktorer som spiller inn. Analyse av hyperspektrale data krever betydelige ressurser og spesialisert programvare, noe som har vært en begrensende faktor i våre undersøkelser.

5.1 Algoritmer og klassifisering

Parallelepiped var en algoritme som var ny for oss men som vi valgte å ta den med i vår oppgave for å finne ut hvor godt den egnet seg til klassifisering av laddekk. På grunn av algoritmens struktur, viste resultatene at den ikke er optimal for klassifisering av landdekk. Dette skyldes at den unngår å klassifisere piksler som den finner utfordrende å kategorisere. Vi testet den på et utsnitt som vist i oppgaven 4.2, men også på hele bildet. Når algoritmen ble testet på hele datasettet ble det som Catalyst beskriver i sin omtale av algoritmen, problemer oppstår på store datasett. Grunnen til dette er den store variasjonen som er i større datasett, som fører til stor ubesluttetshet i algoritmen. Resultat blir at svært lite blir klassifisert.

Vi har utnyttet Parallelepiped som en “før klassifisering”. Vi har fått en forståelse av hvor vi har problemområder og må være ekstra observante i de andre analysene. En av de mest verdifulle aspektene ved å bruke denne algoritmen var evnen til å bekrefte observasjoner og diskusjoner om hvor problemene ligger. Dette inkluderer områder med

gradvise overganger fra gress til jord, hvor det kan være spektralsignaturer som faller mellom to klasser, eller dette kan være på grunn av blandingspikslers. Figur 4.1 og ?? viser RGB bildet og Parallelepiped klassifisering der for eksempel hjulspor på jorden blir ubestemt kategori. Algoritmen identifiserer også områder som den ikke har mottatt tilstrekkelig informasjon til å klassifisere, merket som NULL-klasser. Dette viser seg ofte å være vann, skygge eller mørk jord der reflektansen blir for lav til å klassifisere. Mettede pikslers blir også plassert i NULL-klassen.

Blandings pikslers er alltid i alle varianter av bilder uansett hvor god eller dårlig romlig, radiometrisk og spektral oppløsning vi har. Det vil alltid være pikslers som treffer overgangen mellom to typer overflate. I vår oppgave har vi hatt høy romlig oppløsning i forhold til analysene som skulle gjennomføres. Vi har ikke hatt store problemer knyttet til blandingspikslers, men det er uunngåelig at bildet inneholder blandingspikslers. Problemet med blandingspikslers er når to forskjellige typer objekter deler et piksel i overgang mellom to forskjellige gjenstander. Eksempler på dette kan være asfalt og gress eller bil og asfalt. Dette fører til to forskjellige spektralsignaturer skal bli til en samlet for pikselen. Dette skaper utfordringer fordi det gjør det vanskelig å nøyaktig klassifisere hvilken type landdekke som er til stede i det gitte pikselet. Klassifiseringsalgoritmen kan dermed få problemer med å tildele klasser til disse.

Vi gjorde et forsøk med å resample bilde for å se hvordan resultatene ble påvirket av den romlige oppløsningen. i forhold til antall bånd. Da dette skulle tilsvare en oppløsningen om opptaket ble gjennomført fra satellitt. Resamplingen ble derimot ikke helt optimal ettersom vi endte med å benytte oss av Catalyst sin egne resamplings metode, den baserer seg på nærmeste nabo og ikke en midling som vi hadde ønsket. likevell viser resultatet basert på piksel klassifiseringen med k-means lovende ut. Resultatet presenterer hva som forventes etter a bildet har blitt resamplet, med et mer homogent resultat etter den dominerende klassen i området. Det er også svært lite uenighet i klassifiseringen mellom bildet som er resamplet og det vanlige 16 bånd bilde.

Et av problemene som har vist seg underveis og et problem med landdekkeklassifisering er å klassifisere områder til et begrenset antall klasser. Dette er fordi mange relativt like fysiske gjenstander skal til samme klasse men spektralsignaturen kan variere vesentlig, konsekvensene av dette kan være en tilsynelatende stor variasjon innad i klassene eller feil klassifisering. Dette er for eksempel bygninger der takdekke varierer stort og dermed spektral signaturen. Dette forekommer også med jorder der det kan være stor variasjon i hva som er på disse i form av jordtyper, om det er høstet og type planter. Dette gir klassen relativt stor variasjon i spektralsignaturer som vi skal klassifisere til en klasse.

Sammenlignings grunnlaget i denne oppgaven blir litt oppdelt grunnet Catalyst sin inndeling i “image classification” og “object classification”. Dette fører til vanskeligheter med

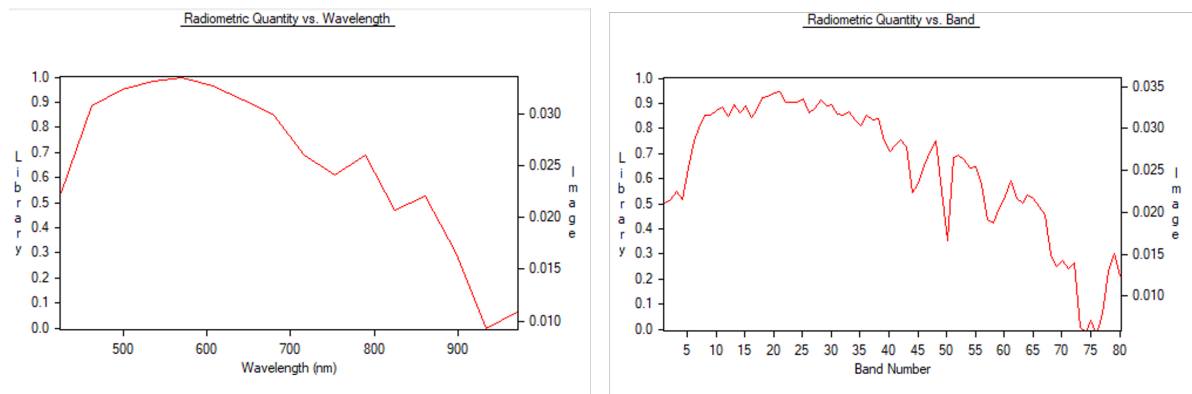
direkte sammenligning. K-means er det separate klassifiserings metoder fordelt på de to fremgangsmåtene, det gir grunnlag for sammenligning av resultatene. Segmentering som er nødvendig med objekt basert klassifisering har vist seg å være en kritisk komponent i vår oppgave, grunnet Catalysts sine begrensinger og hvordan programmet er bygget opp. Denne prosessen, som innebærer å dele et bilde eller datasett inn i forskjellige segmenter, har hatt en betydelig innvirkning på hvordan vi har kunnet klassifisere dataene. I vårt tilfelle har segmenteringen fungert som en form for forhåndsklassifisering. Dette har ført oss til å forenkle klassifiseringsprosessen ved å redusere kompleksiteten av dataene før vi anvender RF, SVM og K-means. For å sikre at vi oppnår de beste resultatene, har vi gjennomført en rekke tester med forskjellige innstillinger. Dette har involvert justering av ulike parametere, eksperimentering med forskjellige metoder for segmentering og klassifisering, og kontinuerlig evaluering av resultatene.

Random Forest og Support Vector Machine kan dermed sammenlignes, fordi det er likt grunnlag i 16 båndts bildet og det er likt grunnlag i 80 båndts bildet. Vi ble litt overasket over resultatene da det er annerledes enn det som var forventet basert på tideligere forskning (Christovam mfl., 2019), der RF har prestert bedre enn SVM. Våre resultater viser at SVM gir best resultater. Det er vanskelig å gi en konkret grunn til det, det kan være flere. En kan være at våre treningsdata er bedre egnet for SVM, da den skal gi bedre resultater på mindre mengde treningsdata.

Det ble forskjell på 16 båndts bildet og 80 båndts bildet når RF skulle utføres. Muligheten for å endre dybde og antall trær ble endret på for å få best mulig resultat. Vi observerte at 80 båndts bildet trengte flere trær, enn 16 båndts bildet, for å få best mulig resultater. Dette gir god mening da spektral oppløsning er forskjellig og datamengden som skal klassifiseres er mye større i 80 båndts bildet. Det er derfor behov for flere trær for å komme fra til mer entydige resultater. For bildet med 16 bånd var det ideelt med omtrent 100 trær, mens for 80 båndts bildet, var det nødvendig å øke antallet trær til 2000 for å forbedre klassifiseringsresultatene. Dybden til trærne ble beholdt på 5 i begge analysene da dette viste de beste resultatene.

Sammenligningen av disse to metodene viser at Random Forest krever en mer detaljert tilpasning av parametere for å oppnå optimale resultater, mens SVM tilbyr en enklere tilnærming da det ikke er parametere å justere.

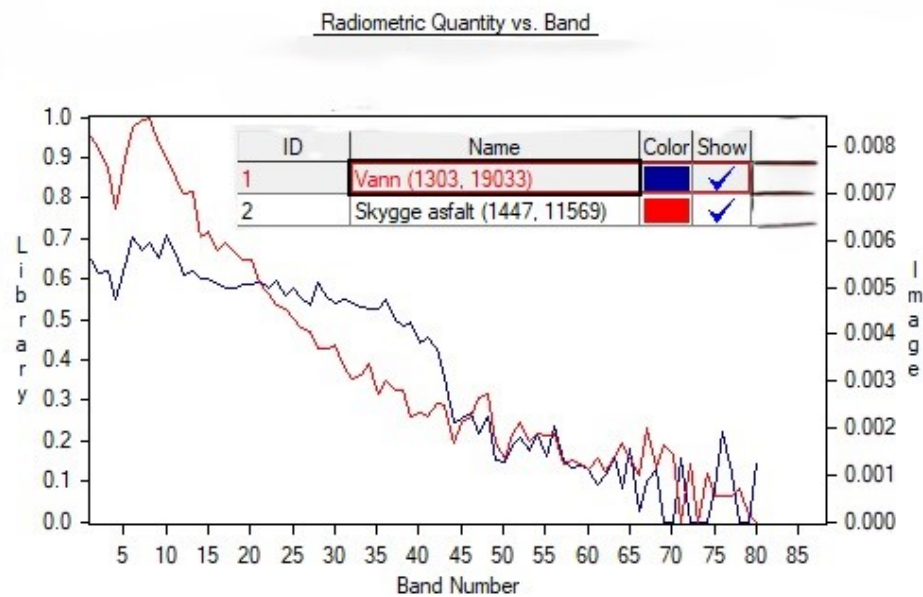
Ser vi på forskjellene i resultater mellom 16 og 80 båndts bildet viser det seg at 16 bånd presterte bedre med både RF og SVM. Dette er nok fordi spektralbinding ble gjort når dataene ble konvertert fra 80 til 16 bånd. Dette fører til en utglatting av spektralsignaturen, vist for et tilfeldig valgt piksel 5.1 fra bildene. Utglattingen av spektralsignatur gir større likhet på pikslene og det blir enklere å klassifiser dem.



Figur 5.1: viser forskjellen i spektralsignatur i 80 og 16 bånd

Skygge er et problem når bilder skal klassifiseres. Tidligere forskning viser at skygge er et evig problem område innen fjernmåling. (Zhang mfl., 2020) Skygge er et fenomen som oppstår på en overflate, som følge av at lyset blir forhindret fra å nå fram til den. Lyskilden vi er avhengige av i fjernmåling er sola. Lysmengden (intensitet) påvirker kvaliteten på de innhentede data, andre faktorer som påvirker kvaliteten er solvinkelen. Solvinkelen endres med årstiden og når på dagen datainnhenting er gjennomført.

Skygge er både det lysfattige rommet bak en ugjennomsiktig gjenstand som det faller lys på, og den mørke figuren, slagskyggen, som gjenstanden danner på en overflate på motsatt side av lyskilden. I dagligspråket sies det at ugjennomsiktige legemer «kaster skygge», svarende til at lysende legemer «kaster lys». (Holtmark, 2023) Det har vi oppdaget i vår klassifisering. Et eksempel er 5.2 der spektral signaturene blir særdels like. Det er også problematisk at det ikke med sikkerhet kan si hva som er i skyggen. Dette gjør det vanskelig å klassifisere skygge da skyggen for eksempel er over en bil. Vil det være en annerledes spektral signatur en om det skulle vært gress. Det andre problemet er mindre informasjon i spektralsignaturen å jobbe med på grunn av den lave reflektansen, som gjør at klassifiseringen innad i skygge vil være svært krevende å differensiere.



Figur 5.2: Spektral signatur forskjell på vann og skygge på to forskjellige punkter

Postprosessering er en del av landdekke klassifisering som vi ikke har tatt for oss i oppgaven. Dette er noe som kunne forbedret resultatene visuelt og hadde ført til et penere produkt, men grunnet tid og prioriteringer har vi valgt å fokusere på hvordan algoritmene har fungert til å klassifisere høyoppløselige hyperspektrale data.

5.2 Veien videre

Det har visst seg at Catalyst ikke passer til bearbeiding av hyperspektrale høyoppløselige bilder. Det vi har fått til av programmering selv, har gitt gode resultater og bidratt til at vi har kunne brukt programvaren på en bedre måte. Fortsettelsen på vårt arbeid ville vært å gjøre de samme analysene, som vi nå har fått til, på det andre datasettet. Da kan vi sammenligne resultatene og få en bedre og grundigere analyse. Et annet alternativ er å finne programvare som er mer egnet for å håndtere hyperspektrale datasett. Vi fikk tips om at ENVI har slik programvare (ENVi, 2024). Vi prøvde å få tak i lisens til det men produsenten hadde ikke noen løsning for studenter. Den mest naturlige veien videre og sannsynligvis den beste løsningen på oppgaven hadde vært å lage alle koder selv i Spyder. Da hadde vi hatt kontroll på alle parametre selv.

Med et utgangspunkt i en ferdigstilt analyse basert på en av framgangsmåtene over vil gi et godt utgangspunkt for et langsiktig mål; spektralbibliotek. Å bygge et stort

spektralbibliotek med reflektans kalibrerte data, åpner for muligheten til å drive landdekkeanalyse uten å benytte eget treningsdata for hvert enkelt prosjekt.

Kapittel 6

Konklusjon

I denne studien ble det utført ulike klassifiseringsmetoder på hyperspektrale bilder for å vurdere hvor godt egnet høyoppløslige hyperspektrale bilder er til landdekkeklassifisering.

K-means klassifiseringen viste bra visuelle resultater på den pikselbaserte klassifisering men mindre godt på segmentert baserte objektklassifiseringen.

Klassifisering med Support Vector Machine (SVM) og Random Forest (RF) viste at SVM hadde best resultater. Høyeste verdi på generell nøyaktighet er 87,5% og en κ verdi på 0,840. Ingen av algoritmene vi testet klarte å hjelpe til med problemene som skygge og mørke piksler fører med seg når det gjelder klassifisering basert på fjernmåling, selv med hyperspektrale data.

Det har vist seg i denne oppgaven at hyperspektrale data kan brukes til landdekkeklassifisering, men datasettet som er brukt i oppgaven har høyere oppløsning og flere bånd enn det som er nødvendig. Ser vi på forskjellene mellom 80 bånd bildene og 16 ser vi bedre klassifiseringsresultater med 16 bånd i både RF og SVM. Dette peker kanskje på at datasettet blir for komplisert for oppgaven den testes på.

Referanser

- Catalyst.Earth. (2024a). K-means clustering. Hentet 6. mai 2024, fra https://catalyst.earth/catalyst-system-files/help/concepts/focus_c/oa_classif_intro_unsuperClass_kmns.html
- Catalyst.Earth. (2024b). OASVMCLASS. Hentet 8. mai 2024, fra https://catalyst.earth/catalyst-system-files/help/references/pciFunction_r/easi/E_oasvmclass.html
- Catalyst.Earth. (2024c). Overview: Object versus pixel. Hentet 24. april 2024, fra https://catalyst.earth/catalyst-system-files/help/concepts/focus_c/oa_basicCpts.html
- Catalyst.Earth. (2024d). Random Trees classifier. Hentet 25. april 2024, fra https://catalyst.earth/catalyst-system-files/help/concepts/focus_c/oa_classif_intro_rt.html
- Catalyst.Earth. (2024e). RESAMP. Hentet 30. april 2024, fra https://catalyst.earth/catalyst-system-files/help/references/pciFunction_r/modeler/M_resamp.html
- Catalyst.Earth. (2024f). Training sites and ground cover. Hentet 30. april 2024, fra https://catalyst.earth/catalyst-system-files/help/tasks/focus_t/Classification_SS12.html
- Chang, C.-C., & Lin, C.-J. (2011). LIBSVM: A library for support vector machines. *ACM Transactions on Intelligent Systems and Technology*, 2(3), 1–27. <https://doi.org/10.1145/1961189.1961199>
- Christovam, L. E., Pessoa, G. G., Shimabukuro, M. H., & Galo, M. L. B. T. (2019). LAND USE AND LAND COVER CLASSIFICATION USING HYPERSPECTRAL IMAGERY: EVALUATING THE PERFORMANCE OF SPECTRAL ANGLE MAPPER, SUPPORT VECTOR MACHINE AND RANDOM FOREST. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences*, XLII-2/W13, 1841–1847. <https://doi.org/10.5194/isprs-archives-XLII-2-W13-1841-2019>
- docs.opencv.org. (2024). OpenCV: Machine Learning Overview. Hentet 6. mai 2024, fra https://docs.opencv.org/4.x/dc/dd6/ml_intro.html

- Earth, G. (2023 mai). Google Earth. Hentet 7. mai 2024, fra <https://earth.google.com/web/@59.93953463,11.60228388,187.25647029a,9733.55856643d,35y,6.32103369h,0t,0r/data=OgMKATA>
- elktro optikk, N. (2024). NEO. Hentet 13. mai 2024, fra <https://www.neo.no/>
- ENVI. (2024). Image Processing & Analysis Software | Geospatial Analysis Software | ENVI®. Hentet 15. mai 2024, fra <https://www.nv5geospatialsoftware.com/Products/ENVI>
- geo-informatie.nl. (1999). Parallelepiped (box) classification. Hentet 9. mai 2024, fra <https://www.geo-informatie.nl/courses/grs20306/lectures/08imageprocessingparta/08imageprocessingparta28.htm>
- Geomatics, P. (2024a). Nearest (Nearest Neighbor Interpolation). Hentet 15. april 2024, fra https://catalyst.earth/catalyst-system-files/help/concepts/orthoengine_c/Chapter_814.html
- Geomatics, P. (2024b). Support Vector Machine classifier. Hentet 18. april 2024, fra https://catalyst.earth/catalyst-system-files/help/concepts/focus_c/oa_classif_intro_svm.html
- gsp.humboldt.edu. (2020). GSP 216. Hentet 14. mai 2024, fra https://gsp.humboldt.edu/olm/Courses/GSP_216/online/lesson7/radiometric.html
- gsp.humboldt.edu. (2024). Accuracy Metrics. Hentet 7. mai 2024, fra https://gsp.humboldt.edu/olm/Courses/GSP_216/lessons/accuracy/metrics.html
- Holtsmark, T. (2023 januar). skygge. Hentet 7. mai 2024, fra <https://snl.no/skygge>
- IDE, S. (2023). Home — Spyder IDE. Hentet 18. april 2024, fra <https://www.spyder-ide.org/>
- Javapoint. (2024). K-Means Clustering Algorithm - Javatpoint. Hentet 6. mai 2024, fra <https://www.javatpoint.com/k-means-clustering-algorithm-in-machine-learning>
- Kruse, F. A. (2012). Mapping surface mineralogy using imaging spectrometry. *Geomorphology*, 137(1), 41–56. <https://doi.org/10.1016/j.geomorph.2010.09.032>
- Larhman. (2018 oktober). English: Maximum-margin hyperplane and margin for an SVM trained on two classes. Samples on margins are called support vectors. Hentet 9. mai 2024, fra https://commons.wikimedia.org/wiki/File:SVM_margin.png
- Leksikon, S. N. (2024 januar). parallelepiped. Hentet 9. mai 2024, fra <https://snl.no/parallelepiped>
- Liu, C., Xing, C., Hu, Q., Wang, S., Zhao, S., & Gao, M. (2022). Stereoscopic hyperspectral remote sensing of the atmospheric environment: Innovation and prospects. *Earth-Science Reviews*, 226, 103958. <https://doi.org/10.1016/j.earscirev.2022.103958>
- nv5geospatialsoftware. (2024). Parallelepiped. Hentet 9. mai 2024, fra <https://www.nv5geospatialsoftware.com/docs/Parallelepiped.html>

- Sethy, P. K., Pandey, C., Sahu, Y. K., & Behera, S. K. (2022). Hyperspectral imagery applications for precision agriculture - a systemic survey. *Multimedia Tools and Applications*, 81(2), 3005–3038. <https://doi.org/10.1007/s11042-021-11729-8>
- Shi, D., & Yang, X. (2015). Support Vector Machines for Land Cover Mapping from Remote Sensor Imagery. I J. Li & X. Yang (Red.), *Monitoring and Modeling of Global Changes: A Geomatics Perspective* (s. 265–279). Springer Netherlands. https://doi.org/10.1007/978-94-017-9813-6_13
- Skauli, T., Haavardsholm, T. V., Kåsen, I., Arisholm, G., Kavara, A., Opsahl, T. O., & Skaugen, A. (2010 april). An airborne real-time hyperspectral target detection system. I S. S. Shen & P. E. Lewis (Red.). <https://doi.org/10.1117/12.850443>
- Stein, D., Schoonmaker, J., & Coolbaugh, E. (2001). Hyperspectral Imaging for Intelligence, Surveillance, and Reconnaissance. *Space and Naval Systems Warfare Center (SSC) San Diego Biennial Review*, 10.
- Wolfe, J. D., Black, S. R., & Boardman, J. (2023). Hyperspectral Analytics in ENVI®.
- Zhang, G., Cerra, D., & Müller, R. (2020). Shadow Detection and Restoration for Hyperspectral Images Based on Nonlinear Spectral Unmixing [Number: 23 Publisher: Multidisciplinary Digital Publishing Institute]. *Remote Sensing*, 12(23), 3985. <https://doi.org/10.3390/rs12233985>

Tillegg A

Vedlegg

Created on Fri Mar 15 09:49:13 2024

@author: Tormod

```
import numpy as np
```

```
from spectral.io import envi
```

```
#Åpne bilde
```

```
img = envi.open("Bjoerkelangen2008-09-24_23_VNIR_1600_SN0005_raw_rad.hdr",  
               "Bjoerkelangen2008-09-24_23_VNIR_1600_SN0005_raw_rad.img")
```

```
#Åpne bildet i mem map
```

```
array = img.open_memmap()
```

```
#Antall bånd som skal slås sammen
```

```
group_size = 5
```

```
#Lager lister for de sammenslåtte båndene og bølgelengdene
```

```
combined_bands = []
```

```
#combined_wavelengths = []
```

```
#'For løkke' som går gjennom og henter ut 5 og 5 bånd og slår dem sammen
```

```
for i in range(0, array.shape[2], group_size):
```

```
    #Trekker ut båndene i grupper på 5
```

```
    group_bands = array[:, :, i:i+group_size]
```

```
    #Slår sammen båndene og midler verdiene
```

```
    combined_band = group_bands.mean(axis=2)
```

```
    #Legger til det sammenslåtte båndet i listen
```

```
    combined_bands.append(combined_band)
```

```
#Konverterer listen av kombinerte bånd til en 'numpy' matrise
```

```
combined_image = np.dstack(combined_bands)*np.mean(combined_wavelengths)
```

```
#Lagre de sammenslåtte båndene som et bilde
```

```
envi.save_image("merged_band_1_80_stort_bilde_23.hdr", combined_image,  
               metadata=img.metadata)
```



Norges miljø- og biovitenskapelige universitet
Noregs miljø- og biovitenskapelige universitet
Norwegian University of Life Sciences

Postboks 5003
NO-1432 Ås
Norway