



Maskininlärning för inversa problem och metrikinlärning

Nya sätt att lösa svåra problem

DAVID GUSTAFSSON, LINNEA AHLINDER, DAVID BERGSTRÖM,
OSCAR BJÖRNHAM, ANDREAS BRORSSON,
MARKUS NORDBERG, LEIF PERSSON

David Gustafsson, Linnea Ahlinder, David Bergström,
Oscar Björnham, Andreas Brorsson, Markus Nordberg,
Leif Persson

Maskininlärning för inversa problem och metrikinlärning

Nya sätt att lösa svåra problem

Titel	Maskininlärning för inversa problem och metrikinlärning – Nya sätt att lösa svåra problem
Title	Machine Learning for Inverse Problems and Metric Learning – New ways to solve difficult problems
Rapportnr/Report no	FOI-R--5093--SE
Månad/Month	December
Utgivningsår/Year	2020
Antal sidor/Pages	55
ISSN	1650-1942
Kund/Customer	Försvarsdepartementet
Forskningsområde	Sensorer och signaturanpassning
FoT-område	Sensorer och signaturanpassning
Projektnr/Project no	A74016
Godkänd av/Approved by	Christian Jönsson
Ansvarig avdelning	Ledningssystem

Detta verk är skyddat enligt lagen (1960:729) om upphovsrätt till litterära och konstnärliga verk, vilket bl.a. innebär att citering är tillåten i enlighet med vad som anges i 22 § i nämnd lag. För att använda verket på ett sätt som inte medges direkt av svensk lag krävs särskild överenskommelse.

This work is protected by the Swedish Act on Copyright in Literary and Artistic Works (1960:729).
Citation is permitted in accordance with article 22 in said act. Any form of use that goes beyond what
is permitted by Swedish copyright law, requires the written permission of FOI.

Sammanfattning

Automatisk analys av data med maskininlärning (ML), framför allt bilder av olika modalitet, är ett ytterst aktivt forskningsområde där stora framsteg har gjorts på kort tid. Inom försvarsorienterad forskning och metodutveckling finns många tillämpningar där ML har stor potential och skulle kunna leda till nya förmågor. Rapporten presenterar några centrala försvarsorienterade tillämpningar där ML idag används i mer begränsad omfattning och lyfter fram hur ML skulle kunna användas inom dessa. Initiala experimentella resultat på sensordata eller simulerad data presenteras.

Tillgången på annoterad träningsdata av god kvalitet med tillräcklig variation (dataförsörjning) har varit helt avgörande för framgångarna med ML. Antalet träningsexempel som krävs för att nå framgång är tillämpningsberoende, men är nästan alltid mycket stor. Inom försvarstillämpningar är vanligen tillgången på träningsdata ytterst begränsad vilket utgör ett hinder vid tillämpning av ML. Experimentella resultat där simulerad data har använts för utvärdering av några ML-ansatser presenteras.

En central användning av ML är att analysera sensordata, dvs. metoder att extrahera information från sensorer. Måldetektering och målklassificering är frekvent förekommande problem där detta kan vara användbart. I rapporten presenteras:

- Ytor kontaminerade med mikroorganismer är svåra att sanera, speciellt sådana som består av sporer. Efter sanering av sporer på kontaminerade ytor vill man verifiera att alla sporer är döda. Metrikinlärningsbaserade ansatser presenteras som kan användas för klassificering av sporer som avdöda eller livsdugliga i hyperspektrala bilddata. Experimentella resultat på hyperspektrala bilder presenteras.

En annan typ av problem där ML frekvent har börjat användas under senare år är så kallade inversa problem. Vid inversa problem vill man återskapa data från en tidigare tidpunkt eller data innan degenerering. I rapporten diskuteras tre specifika exempel:

- Brusiga bilder med låg kontrast är frekvent förekommande inom mörkerseende, där bilder insamlats under dåliga ljusförhållanden. ML-metoder för brusreducering och avbrusning, dvs. för återskapande av brusfria bilder, har inom många tillämpningar visat sig ge lovande resultat. Av speciellt intresse är ML-metoder som inte kräver brusfria bilder för inlärning, utan metoder där uppsättningar av endast brusiga bilder räcker som träningsdata. Några ML-baserade avbrusningsmetoderna är experimentellt utvärderade på mörkerbilder.
- Vid invers spridningsmodellering vill man utifrån en uppsättning mätningar från olika mätpunkter bestämma en eller flera spridningskällors position och styrka. Traditionellt brukar invers spridningsmodellering lösas med explicita spridningsmodeller och regularisering. I rapporten ansätts och diskuteras istället en ML-baserad lösning för estimering av källornas position och styrka. ML-metoderna är implementerade och experimentellt utvärderade på simulerad data.
- Compressed Sensing kan användas för att rekonstruera hyperspektrala Raman signaturer från en uppsättning mätningar där spektral och spatial

information har blandats. Ramansignaturer kan användas för att detektera spårmängder av explosive ämnen. Rekonstruktion med standardmetoder är relativt tidskrävande vilket har begränsat tillämpningarna där tekniken har varit applicerbar. I rapporten presenteras ML-baserade ansatser för rekonstruktion och experimentella resultat på simulerad data. ML-baserade metoder är avsevärt mycket snabbare samtidigt som signaturrekonstruktion ofta är bättre än med klassiska metoder.

Nyckelord: AI, artificiell intelligens, ML, maskininlärning, DL, deep learning, försvarstillämpningar, metrikinlärning, hyperspektral data, avbrusning, atmosfärisk spridning, CS, compressed sensing, Raman spektrum, CASSI

Innehåll

1	Inledning.....	9
1.1	Bakgrund	9
1.2	Nya möjligheter	10
1.3	Dataförsörjning	11
1.4	Översikt av tillämpningar	11
2	Klassificering av sporer	13
2.1	Bakgrund	13
2.2	SWIR-bilder	14
2.3	Klassificering med metrikinlärning	15
2.4	Spektral förbehandling.....	16
	2.4.1Relevant komponentanalys (RCA)	16
	2.4.2k-NN17	
2.5	Resultat	17
2.6	Sammanfattning.....	19
3	Avbrusning av bilder från mörkerkamera.....	21
3.1	Bakgrund	21
3.2	Klassisk avbrusning.....	21
3.3	Avbrusning med djupinlärning	21
3.4	Experiment.....	24
	3.4.1Träningsdata	24
	3.4.2Avbrusningsmetoder	25
	3.4.3Bildkvalitetsmått	26
3.5	Resultat	26
	3.5.1Bildexempel.....	26
	3.5.2Bildkvalitetsmått	28
3.6	Slutsatser och diskussion.....	29
4	Maskininlärning för invers modellering av atmosfärisk spridning	31
4.1	Bakgrund	31
4.2	Invers modellering	32
4.3	Maskininlärning.....	32
	4.3.1Metodbeskrivning	33
	4.3.2Generering av träningsdata.....	33
4.4	Resultat	37
4.5	Diskussion	38
5	Deep Learning för avkodning av hyperspektrala Raman-bilder insamlade med Compressed Sensing.....	40
5.1	Bakgrund	40
5.2	Ramanspektroskopi.....	40

5.3 Coded Aperture Snapshot Spectral Imaging - CASSI.....	41
5.4 Experiment.....	42
5.5 Maskininlärning och Deep Learning	43
5.6 Metodbeskrivning	44
5.6.1Datagenerering	44
5.6.2Nätverksarkitektur	46
5.7 Resultat	48
5.8 Träning och rekonstruering av få Ramanspektra.....	49
5.9 Träning och rekonstruering med brett antal Ramanspektra	50
5.10 Slutsatser och diskussion.....	51
Referenser	52

1 Inledning

Automatisk analys av data med maskininlärning (ML), framför allt bilder av olika modalitet, är ett ytterst aktivt forskningsområde där stora framsteg har gjorts på kort tid. Dessa framsteg kommer även att påverka informationsinhämtning inom flertalet försvarstillämpningar, dels genom en uppsnabbning av analys av mycket stora mängder data och dels genom högre kvalitet i analyserna. Utvecklingen inom området går väldigt snabbt.

Inom försvarsorienterad forskning och metodutveckling finns många tillämpningar där ML har stor potential och skulle kunna leda till nya förmågor. Inom vissa tillämpningar finns etablerade metoder som genererar resultat av varierande kvalitet, medan det inom andra saknas etablerade lösningar. Rapporten presenterar några centrala försvarsorienterade tillämpningar där ML inte används idag eller i mer begränsad omfattning och lyfter fram hur ML skulle kunna användas inom dessa. Fokus för rapporten är analys av sensordata, dvs. metoder att extrahera information från sensorer. Information ska i detta sammanhang tolkas brett och inkluderar förbättring av sensordata för visuell analys av en operatör.

Rapporten presenterar resultat från projektet Avdelningsöverskridande AI-projekt vars syfte var att öka kunskapen om hur Artificiell Intelligens (AI), och då primärt ML, kan användas inom försvarstillämpningar för att förbättra eller uppnå nya förmågor.

1.1 Bakgrund

Maskininlärning och speciellt Deep Learning (DL) har på kort tid genererat anmärkningsvärt bra resultat inom ett stort antal tillämpningar, framför allt inom bildanalys. Inom flera bildbehandlingstillämpningar där man tidigare endast lyckats uppnå resultat som varit för dåliga för att vara praktiskt användbara har man på kort tid gjort anmärkningsvärda framsteg. Även inom närliggande områden, såsom textanalys och talanalys, har man rönt stora framgångar.

Inom bildanalys har möjligheten att detektera och klassificera utbredda objekt i visuella bilder nått närmast oanade nivåer. Framgången inom visuella bilder beror på en rad faktorer som delvis också stämmer på vissa försvarstillämpningar såsom automatisk analys av satellitbilder, hyperspektrala data, radar och måldetektion i termiska bilder. Det är framförallt tre faktorer som har varit avgörande för framgången med ML-baserad analys:

- **Dataförsörjning:** Tillgången på annoterad träningsdata av god kvalitet med tillräcklig variation har varit helt avgörande för framgångarna med ML. Antalet träningsexempel som krävs för att nå framgång är tillämpningsberoende men är nästan alltid mycket stor.
- **Algoritmteckning:** Algoritmer som nästan uteslutande används är baserade på Deep Learning, vilket är ett neuronät (NN) med många dolda lager. Nya träningsalgoritmer har möjliggjort träning av NN med många lager vilket har varit avgörande för många tillämpningar.
- **Hårdvara:** Stora NN som behöver tränas på stora mängder data kräver mycket beräkningskraft. Utvecklingen av nya grafikkort dedikerade för snabba beräkningar har öppnat upp för massiv träning av nät.

Den mest avgörande framgångsfaktorn är att tillgången på annoterad träningsdata av god kvalitet har ökat.

1.2 Nya möjligheter

Maskininlärning kommer påverka många olika delar inom informationsinhämtning från sensordata på olika sätt. Förmågan att automatiskt upptäcka och klassificera olika typer av objekt i sensordata av olika modalitet öppnar upp nya möjligheter för informationsinhämtning. Möjligheten att analysera stora mängder data i nära realtid samtidigt som mängden information som kan extraheras från data kommer att öka.

Centrala områden inom ML för automatisk informationsextraktion ur sensordata, och framför allt bilder av olika modalitet, exemplifierade med satellitbilder:

- **Detektion/klassificering av utbredda mål** – automatisk detektion och klassificering av objekt i satellitbilder där målet upptar åtminstone några pixlar. Exempel på tillämpningar är detektion av hus i rural miljö, storleksskattning på hus och typ av byggnad. Detektion av specifika hus så som kyrkor eller fabriker, eller tillfälliga bosättningar som husvagnar eller husbilar.
- **Detektion/klassificering av pixelmål** – klassificering av mål som enbart täcker en eller ett fåtal pixlar. Dessa metoder kräver ofta multi- eller hyperspektrala bilder samt att objekten har en distinkt spektral signatur.
- **Anomalidetektion** – detektion av objekt som avviker från det som vanligen förekommer i en given miljö. Vid användning av dessa metoder känner man inte till vad man letar efter, istället är målsättningen att upptäcka objekt som normalt inte förekommer i miljön – dvs. anomalier. Exempel på tillämpningar är att upptäcka tillverkade (mänskligt framställda) objekt i naturen så som föroreningar, plastpåsar, kläder eller fordon. Metoden kan också användas för att upptäcka fartyg eller utsläpp till sjöss.
- **Semantisk segmentering** – vid semantisk segmentering klassificeras varje pixel till en viss typ av objekt. Semantisk segmentering genererar en mask med samma upplösning som bilden, där maskens innehåll indikerar vilket objekt som pixeln innehåller. Tillämpningar kan vara att dela upp en satellitbild i olika regionstyper, så som barrskog, lövskog, torr åkermark, våt åkermark, snötäckt mark, plöjd mark, grusväg, asfaltväg, byggnader och sjöar.
- **Förändringsdetektion** – om en satellit regelbundet samlar in bilder innehållande samma region kan förändringar i bilden automatiskt upptäckas. Vanligen vill man inte upptäcka naturliga förändringar så som att gräset har växt eller att det har snöat. Istället vill man vanligen upptäcka mer permanenta förändringar så som nya vägar, nya byggnader, byggnader som har förvunnit eller skogar som har avverkats.

En klass av problem där ML på senare tid har använts mer frekvent är så kallade **inversa problem**. Vid inversa problem är målsättningen att återskapa sensordata eller ett tillstånd utifrån transformerad data eller information från en senare tidpunkt. Exempel på sådana problem är att återskapa en fokuserad, skarp bild utifrån en ofokuserad bild, en brusfri bild utifrån en brusig bild (avbrusning) eller att återskapa en hyperspektral bild utifrån en multispektral bild.

Centrala områden av ML för automatisk förbättring av bildinnehållet för visualisering inkluderar:

- **Bildförbättring** – ML har använts till att förbättra bilder i form av avbrusning och ökad kontrast i bilderna. Vissa ML-baserade avbrusningsmetoder kräver ingen explicit brusmodell eller brusfria bilder vid inlärningen vilket i många fall kan vara svårt att få tillgång till. Dessa metoder kan även användas för att ta bort moln som delvis skymmer intressanta mål, kompensera för atmosfärsstörningar och ljusvariationer i scenen.

- **Spatial superupplösning** – innebär att man utifrån en lågupplöst bild rekonstruerar en högupplöst bild. Mål med en liten spatial utbredning kan med superupplösning visas med högre upplösning vid visuell inspektion.
- **Spektral superupplösning** – innebär att man utifrån ett fåtal spektralband – t.ex. R (rött), G (grönt), B (blått) och NIR (nära infrarött) – rekonstruerar fler spektralband.

1.3 Dataförsörjning

Maskininlärning har varit framgångsrik inom ett stort antal väldigt olika tillämpningar inom bildanalys med bilder av olika modalitet. Framgångarna som initialt härrör från vanliga visuella bilder har generaliserats till andra bildmodaliteter och helt andra frågeställningar, framför allt inom medicin. I bildtillämpningar där man antingen har tillgång till stora mängder annoterad data eller kan använda simulerad data vid träning har DL ofta gett ytterst goda resultat. Givet att dataförsörjningsproblemet kan lösas finns ingenting som tyder på att ML inte skulle kunna generera goda resultat även inom många andra försvarsorienterade tillämpningar som inte bygger på bilddata.

Dataförsörjningsproblemet måste lösas för att framstegen inom ML ska kunna generaliseras till nya försvarsorienterade tillämpningar. Mängden öppen data har ökat inom vissa försvarstillämpningar och kommer troligen att öka även i framtiden även om den är långt ifrån vad som finns tillgängligt inom många civila, kommersiella tillämpningar. Inom andra försvarsorienterade tillämpningar är tillgången på annoterad data obefintlig och dessutom ytterst kostsam att ta fram. *Dataförsörjningsproblemet kommer att vara det största hindret på lång och kort sikt för tillämpning av ML.*

1.4 Översikt av tillämpningar

Detta avsnitt innehåller en kortfattad sammanfattning av de försvarsorienterade tillämpningar som presenteras i senare kapitel. Varje efterföljande kapitel innehåller en försvarsorienterad tillämpning, något konkret problem man vill lösa, en skiss på hur ML skulle kunna användas för att lösa problemet och initiala experimental resultat på sensordata eller simulerad data.

Hyperspektrala data används inom försvarstillämpningar både på makro- och mikronivå. På makronivå vill man upptäcka utbredda eller punktformade mål, såsom byggnader eller fordon, från hyperspektral data insamlad från flyg eller satellit. På mikronivå vill man analysera förekomsten av ämnen i labprover. I kapitel 2 används hyperspektral data för att klassificera sporer, dvs. bakterier som är försatta i viloläge. Ytor har varit kontaminerade av sporer som har sanerats och man vill avgöra om alla sporer är döda eller om några har överlevt saneringen. Målsättningen är att särskilja spektrala signaturer för levande och döda sporer med ML. För att särskilja spektrala signaturer från sporer krävs ett likhetsmått eller en metrik som mäter avståndet mellan spektrala signaturer. Det vanliga euklidiska avståndet kan användas för att särskilja spektrala signaturer men är i många tillämpningar en mindre lämplig metrik. Istället kan ML och så kallad metrikinlärning användas för att bestämma ett avståndsmått som framhäver de särskiljande spektrala egenskaperna hos signaturerna. Metrikinlärning är en datadriven ansats där träningsdata används för att bestämma ett avståndsmått som är problem och data specifikt. Olika metrikinlärningsansatser utvärderas för att särskilja levande och döda sporer.

För *mörkerförmåga* används ljuskänsliga sensorer där brusproblematik naturligt uppstår under användning vid väldigt dåliga ljusförhållanden. Uppgifter som detektion, klassificering och situationsanalys är ytterst krävande om bilder är väldigt brusade, både för

en mänsklig operatör och för automatiska detektions- och igenkänningsalgoritmer. Avbrusning med ML-baserade ansatser har gjort stora framsteg under senare tid. Särskilt intressanta är ansatser som inte kräver tillgång på brusfri träningsdata. Två ML-baserade avbrusningsmetoder har implementerats och utvärderats på mycket brusiga insamlade under dåliga ljusförhållanden. Då brusfria facit bilder för dessa bilder saknas är resultatutvärdering en utmaning och olika mått för att karaktärisera metoderna har används. Äldre klassiska bildkvalitésmått så väl som nyare rangordningsbaserade mått har används vid utvärderingen. De ML-baserade avbrusningsmetoderna har utvärderats mot väletablerade state-of-the-art metoder och står sig väl i mot dessa. ML-baserad avbrusning för lågljusbilder presenteras i kapitel 3.

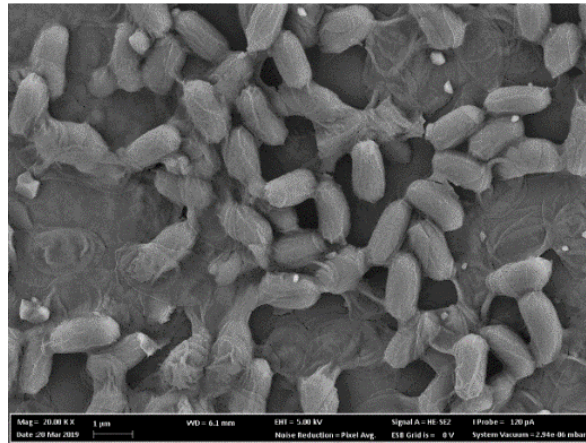
Inom *invers spridningsmodellering* vill man utifrån en sekvens av mätningar från geografisk utplacerade sensorer bestämma spridningskällans – eller spridningskällornas – position och styrka. Intresset kan vara på en lite skala inom, lokalt inom en stadsdel eller på större skala inom en region eller globalt. Traditionellt brukar invers spridningsmodellering lösas med explicita modellantaganden – s.k. regularisering – för att bestämma källornas position och styrka. En bildbaserad deep learning ansats där man utifrån mätpunkter (virtuella sensorer) inbakade i bilder försöker bestämma spridningskällans position och styrka har använts. Träningsdata utgörs av spridningssimuleringar där ett varierande antal mätpunkter har (virtuellt) placerats ut i scenen och utifrån dessa mätpunkter predikteras källans position och styrka. Ansatser och experimentella resultat för spridningsmodellering med ML presenteras i kapitel 4.

Improviserade bomber eller IED:er (Improvised Explosive Device) är ett ökande hot i samhället. Laserbaserad Ramanspektroskopi är en teknik som kan detektera och identifiera IED:er på ett säkert avstånd. FOI har utvecklat en avbildande hyperspektralt sensor för Ramanspektroskopi som kan användas för detektion av spårängdsdetektion av explosivämnesspår. Hyperspektralsensorn bygger på Compressed Sensing (CS) där man utifrån en uppsättning mätningar rekonstruerar en hyperspektral kub innehållande Ramanspektrum. Rekonstruktionstiden för en hyperspektral kub är dock lång och begränsar tillämpningsbarheten av tekniken. ML och DL kan användas för att rekonstruera högupplösliga bilder från ett fåtal parametrar. ML ansatser för rekonstruktion av hyperspektrala kuber från ett fåtal CS-mätningar studeras och utvärderas på simulerad data i projektet. Experimentella resultat på simulerad data indikerar att ML ansatser kan rekonstruera hyperspektrala kuber på några få millisekunder vilket öppnar upp för nya tillämpningar av tekniken. Rekonstruktion av Ramanspektrum från CS-mätningar presenteras i kapitel 5.

2 Klassificering av sporer

2.1 Bakgrund

Vissa typer av bakterier har förmågan att bilda sporer. I detta tillstånd befinner sig bakterierna i ett vilostadium, där de inte kan reproducera sig och där de är nästan helt metaboliskt inaktiva. Sporer har en stor motståndskraft mot såväl värme som uttorkning, strålning och saneringsmedel och de kan också befinna sig i detta tillstånd under mycket lång tid. Om livsmiljön ändras kan sporerna gå ur sporstadiet och orsaka sjukdom. Ett exempel på en sporbildande bakterie är *Bacillus anthracis*, som orsakar den allvarliga sjukdomen mjältbrand och som har använts som ett biologiskt terrorvapen. *Bacillus thuringiensis* (Figur 1) är en sporbildande bakterie som här används som ett icke-patogent similiämne för *B. anthracis*.



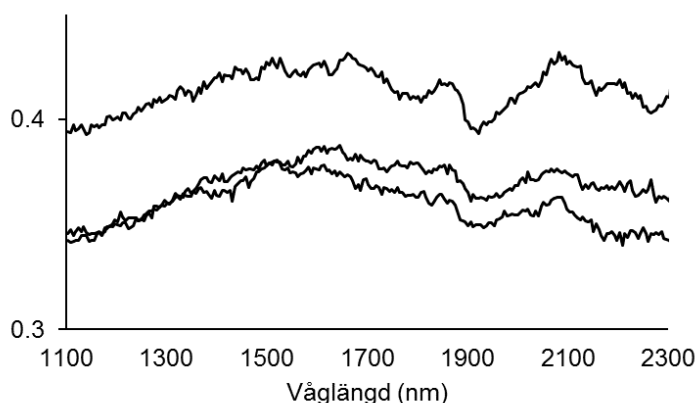
Figur 1 *B. thuringiensis*-sporer avbildade i ett svepelektronmikroskop (20 000× förstoring).

Sporer kan avdödas med exempelvis UV-ljus, vilket orsakar död genom DNA-skador [2], men för att säkerställa att avdödningen har fungerat krävs också metoder för att testa att sporerna ej kan återgå till sitt bakteriestadium och orsaka smitta[3, 4]. Detta är framför allt viktigt vid riskbedömningar och forskningsfrågeställningar, men kan också vara viktigt i miljöer som behöver vara fria från smittsamma ämnen, t.ex. sjukhusmiljöer. Normalt sett görs detta genom odling av uppsamlat material, vilket tar lång tid. En del mikroorganismer är dessutom mycket svåra att odla. Det finns därför ett behov av att utveckla nya, snabbare metoder. Som en delaktivitet i anslagsprojektet Kodek (kontaminering-dekontaminering), där det bedrivs forskning kring sanering av kemiska, biologiska och radiologiska ämnen, har FOI sedan ett antal år tillbaka undersökt om olika typer av spektroskopi skulle kunna användas för att klassificera sporer som levande eller döda[5].

En av de testade metoderna är avbildning med kortvågig infrarödkamera (SWIR-kamera), vilket möjliggör snabba mätningar av sporer på olika ytor. Droppar med intorkade sporer kan mätas på cirka 30 sekunder. Proverna placeras i det här fallet på ett motoriserat bord och belyses med ljus. En kamera samlar in reflekterat ljus med våglängder i SWIR-området, en linje i taget, och bygger på så sätt upp en bild, en så kallad hyperspektral bild, där varje bildpunkt består av ett SWIR-spektrum.

Tekniken bygger på att ljus av vissa våglängder absorberas av provet och därför saknas i det ljus som reflekteras, vilket visar sig som toppar i spektra. De absorberade

våglängderna motsvarar molekylära vibrationer, vilka är karaktäristiska för olika molekyler och SWIR-spektra utgör på så sätt ett kemiskt fingeravtryck. I SWIR-området är topparna breda och kan vara svåra att tolka eftersom de ofta består av flera överlagrade toppar, men det går exempelvis att urskilja toppar relaterade till vatten, fettsyror och protein[6, 7], vilka alla är viktiga beståndsdelar i sporer (Figur 2). Hyperspektral avbildning har använts i flera tillämpningar inom livsmedelssäkerhet [7-9] och det finns således stöd i litteraturen för att reflektansspektra i SWIR-området kan användas för att särskilja olika bakterier. Vår hypotes är att det också finns tillräckligt med kemisk information för att särskilja döda sporer från levande sporer. Det som i sådant fall kan förväntas är, beroende på avdödningsmetod, små förändringar kopplade till toppar relaterade till DNA eller sporens membran (i huvudsak lipider). Eftersom dessa spektrala skillnader kan förväntas vara mycket små, samtidigt som SWIR-avbildning genererar hundratusentals spektra, finns det stor potential i att använda maskinlärning för att klassificera spektra.



Figur 2 Exempel på SWIR-spektra från intorkade *B. thuringiensis*-sporer.

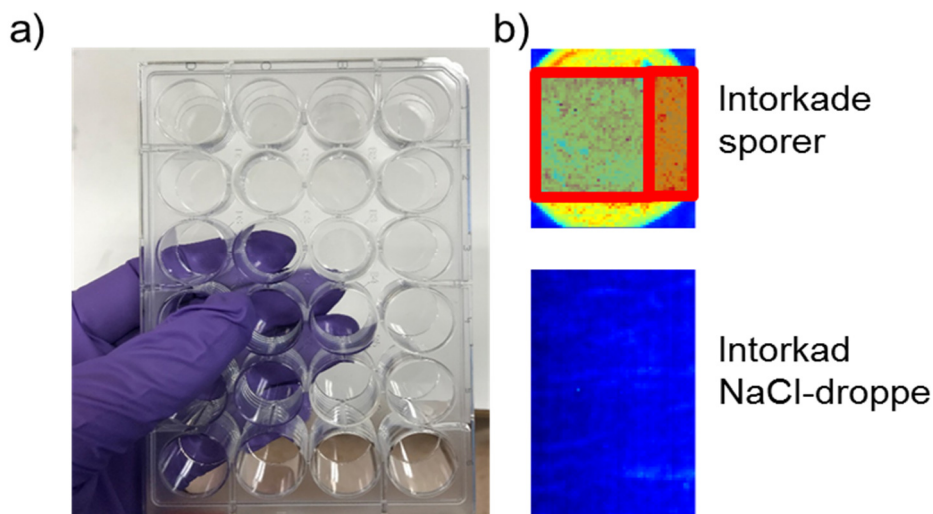
2.2 SWIR-bilder

Det finns flera mättekniska utmaningar vid SWIR-avbildning av sporer. Detaljer kring detta ligger utanför ramarna för att beskrivas här. Generellt kan dock nämnas att bakgrunden kan störa och att saneringsmedlet i sig självt kan ge toppar, vilket gör att en klassificeringsmodell riskerar att klassificera närvaro eller frånvaro av ett saneringsmedel snarare än levande eller döda sporer. Här fokuseras det därför på avdödning med UV-strålning, vilket innebär att några kemikalier ej tillsätts under saneringen, och att sporena torkas in på kalciumfluoridytar som ej ger några toppar i SWIR-området (Figur 3 a).

I projektet Kodek har det samlats in sådana data för *B. thuringiensis* och i nuläget finns 2 insamlade bilder från levande sporer och 13 bilder från sporer som avdödat med UV-strålning och som är kontrollerat avdödade (genom odling). Bilderna är insamlade med en SWIR-kamera (Specim, Finland) utrustad med makroobjektiv och monterad på en mätuppställning med ett motoriserat bord och halogenbelysning (ViaSpec Hyperspectral Imaging System, Middleton Spectral Vision, USA). Varje bild består av omkring 300×300 spektra, d.v.s. omkring 180 000–1 170 000 spektra per grupp, vart och ett med närmare 300 spektrala datapunkter. I de bilder som har samlats in har koncentrationen av *B. thuringiensis* varit hög och hela ytan med intorkade sporer kan förväntas vara täckt av sporer i varierande mängd och varje spektrum förväntas därför visa toppar från sporer

och eventuella toppar från bakgrunden, t.ex. små mängder kvarvarande vatten. Det kan påpekas att den spatiala upplösningen med SWIR-kameran är för dålig för att enskilda sporer ska kunna urskiljas i bilderna, men i och med att koncentrationen är hög går det tydligt att urskilja sporer från bakgrunden (Figur 3 b).

Eftersom det fanns ett mycket begränsat antal bilder från levande sporer delades samtliga bilder upp så att 80 % av dem användes för att träna och 20 % användes för att testa klassificeringsmodellerna (Figur 3 b). I klassificeringsmodellerna jämnades även grupperna ut, så att lika många spektra från levande och döda sporer användes för att träna modellerna.



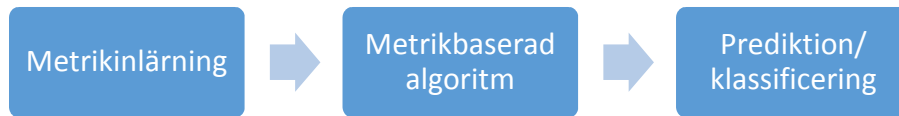
Figur 3 a) Kalciumfluoridytor med intorkade sporer (i de fyra översta brunnarna). b) SWIR-bilder från kalciumfluoridytor med intorkade sporer i NaCl-lösning respektive intorkad NaCl-lösning (samma volym). Bilderna visar komponent 1 i en principalkomponentanalys. Det större markerade området i den övre bilden motsvarar den del av bilden som använts för att träna klassificeringsmodellerna och det mindre markerade området i den övre bilden motsvarar den del av bilden som använts för att testa klassificeringsmodellerna.

2.3 Klassificering med metrikinlärning

I en klassificering med metrikinlärning sker analysen i flera steg (Figur 4), där det första steget handlar om att ta fram en metrik, det vill säga en funktion som beskriver avstånd eller likheter/olikheter mellan observationer[10]. I sin enklaste form används Euklidiskt avstånd, det vill säga det raka avståndet mellan två observationer. Inom metrikinlärning görs ofta en uppskattning av Mahalanobisavståndet, vilket beskriver avståndet mellan statistiska populationer. Det finns ett flertal metoder för att ta fram metrikfunktioner, där många bygger på att träningsdata delas upp i positiva och/eller negativa par, det vill säga par av observationer som antingen tillhör samma grupp eller ej tillhör samma grupp[10]. De flesta algoritmer ställs därefter upp som optimeringsproblem som syftar till att hitta parametrar till den funktion som ger bäst överensstämmelse med de positiva/negativa paren. Metriken används sedan i någon metrikbaserad algoritm, i sin enklaste form *k-Nearest neighbor* (*k*-NN), en metod avståndet mellan observationer beräknas med någon metrikfunktion och där observationer med okänd klasstillhörighet tilldelas den klass som majoriteten av dess *k* närmsta grannar tillhör.

I det här miniprojektet valdes metrikinlärning och metrikbaserad algoritm delvis utifrån vad som var enkelt att implementera och komma igång med. I första hand testades därför algoritmer som redan fanns implementerade i det statistiska programspråket R.

Analyserna har gjorts i R 3.5 [11] med tilläggs paketen `hyperSpec`[12], `prospectr`[13], `class` [14] och `dml`[15].



Figur 4 Översikt över stegen i en metrikbaserad klassificering.

2.4 Spektral förbehandling

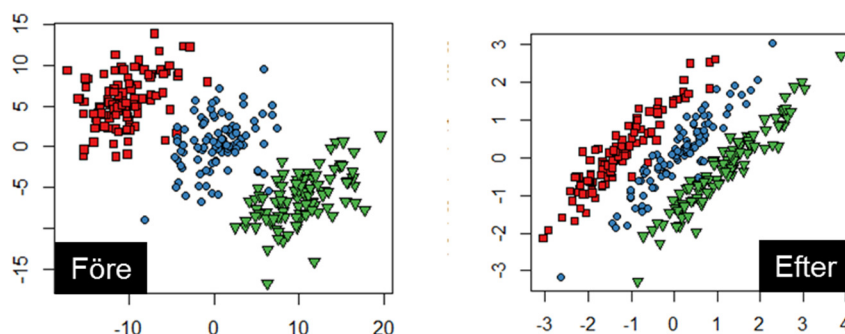
Det kan vara nödvändigt att förbehandla data innan de används i en klassificeringsmodell. Exempelvis kan en varierande bakgrund och brus störa en analys. Förbehandlingsmetoder testades i preliminära modeller och jämfördes med avseende på korrekt klassificering av träningsdata. Bäst modell erhöles trots allt med rådata, det vill säga obehandlade data. Förbehandlingsmetoder som testades är vektornormalisering, *standard normal variate* (SNV, en vanlig metod för att hantera spridningseffekter) och brusreducering med hjälp av Savitzky-Golay-filter.

2.4.1 Relevant komponentanalys (RCA)

Relevant komponentanalys (*relevant component analysis*, RCA) är en metrikinlärningsmetod där data transformeras i syfte att minska spridningen inom grupper och samtidigt förstärka avståndet mellan grupper [10, 16]. Metoden går ut på att data delas in i mindre grupper, *chunklets*, där alla tillhör samma grupp. Med hjälp av dem görs en global linjärtransformation där relevanta dimensioner tilldelas höga vikter. Detta sker genom att optimera en funktion så att den totala variansen mellan chunklets med observationer från skilda grupper maximeras medan variansen inom enskilda chunklets minimeras. Figur 5 visar hur en RCA-transformation kan se ut för exempeldata med tre grupper och två variabler.

I detta fall är det känt på förhand vilka spektra som är från levande sporer och vilka som är från döda sporer. En fördel med RCA är att hela datauppsättningen inte behöver ha kända klasstillhörigheter; Det räcker med att veta vilka grupper av data (chunklets) som tillhör samma klass. Ett exempel på ett annat sammanhang där RCA kan vara användbart är vid ansiktsigenkänning i videosekvenser. Det räcker i det fallet att identifiera en sekvens med ett antal bilder med en person som sedan kan kännas igen i andra bildsekvenser utan att någon etikett behöver sättas på data. I det här fallet, med SWIR-bilder med kända grupptillhörigheter, utnyttjas inte den egenskapen i RCA-algoritmen. Däremot har RCA applicerats för att transformera data och förhoppningsvis förbättra en klassificering.

Övningsdata med 3 grupper



Figur 5 Illustration av transformation med RCA. Figuren föreställer testdata med tre olika grupper och två variabler före och efter transformation. Data i figuren har ej någon koppling till SWIR-bilderna.

2.4.2 k-NN

Avstånden mellan RCA-transformerade data beräknades i en k-NN-modell, som därefter användes för att klassificera data i uppsättningen med träningsdata. k-NN innebär att observationer klassificeras till den klass som majoriteten av de k närmsta grannarna tillhör. Parametern k varierades och bäst klassificering, med avseende på antalet korrekt klassificerade spektra från uppsättningen med träningsdata, erhöles med $k=5$.

2.5 Resultat

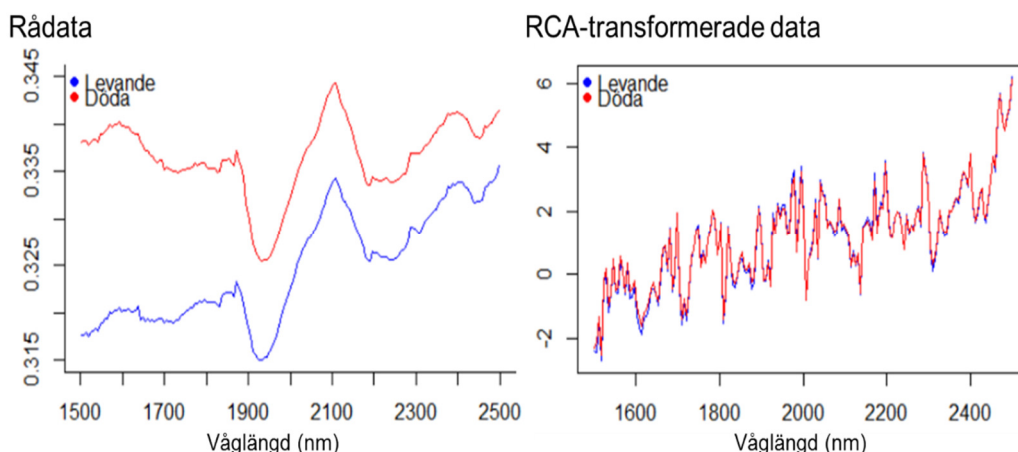
Eftersom toppar i SWIR-spektra är breda och överlappande och eftersom bakgrunden kan variera på grund av bland annat spridningseffekter kan det vara mycket svårt att tolka spektra. I Figur 6 visas medelvärdesspektra för levande och döda sporer före och efter RCA-transformation. Med blotta ögat är det svårt att se någon påtaglig skillnad mellan grupperna utöver en stor spridning i bakgrund. De toppar som främst kan urskiljas är 1925–1945 nm som är ett område med toppar från vatten och proteinbundet vatten. Runt 2200 nm finns även en bred topp som i huvudsak kan tillskrivas amider (protein)[6, 7]. Att dessa och andra toppar kan förefalla skilja sig åt mellan grupperna beror främst på att bakgrunden varierar. Efter RCA-transformation är dock den varierande bakgrunden mer utjämnad och medelvärdesspektra för de båda grupperna är mycket lika varandra. Inga tydliga toppar kan heller urskiljas efter transformationen.

RCA-transformerade data användes i en k-NN-modell som fick klassificera de utsnitt av bilderna som användes för testning. Figur 7 visar en exempelbild där den övre bilden är en SWIR-bild över en yta med levande sporer (komponent 1 i en principalkomponentanalys) med ett markerat område som motsvarar den del av bilden som använts för testning. Den undre delen av Figur 7 visar resultatet av klassificeringen. Klassificering av samtliga bilder i uppsättningen visas i Figur 8. I samtliga bilder, oavsett om de föreställer levande eller avdödade sporer, blir delar av bilderna klassificerade som döda sporer och delar blir klassificerade som levande sporer. I en del fall blir det av okänd anledning en skarp gräns mellan områden med levande och döda sporer, men i de flesta fall tycks områden klassificerade som levande sporer spridas ut mer slumpartat.

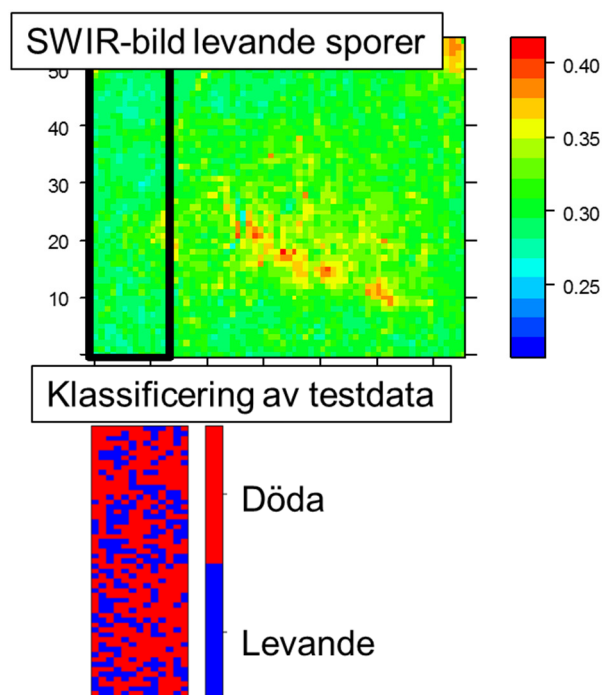
Andelen korrekt klassificerade bildpunkter är totalt sett endast 56 %, varav 29 % av bildpunkterna från levande sporer blir korrekt klassificerade och 60 % av bildpunkterna från de döda sporer blir korrekt klassificerade. Detta visar tydligt att modellen i sin nuvarande form inte kan användas för att klassificera sporer som levande eller döda.

En orsak till detta är troligen att datauppsättningen är för liten. Det skulle behövas långt fler än två bilder från levande sporer för att bygga en tillförlitlig modell, inte minst för att bakgrunden varierar kraftigt och bidrar till en stor spridning inom grupperna. Även om varje bild består av många spektra är spektra också till stor del korrelerade eftersom de härrör från samma prov och är insamlade vid samma tillfälle. Flera parametrar kan påverka bilderna, exempelvis påverkas spektra av eventuella spridningseffekter som beror på att ljuset kan ta olika banor. Även om proverna är hanterade så lika som möjligt kan de också ha små variationer i exempelvis fukttinhåll, vilket också påverkar spektra eftersom vatten ger flera toppar i SWIR-området. Även om flertalet förbehandlingsmetoder testades är det troligt att den här typen av variation mellan prover har stört klassificeringen. En möjlighet är också att bilderna inte har den spatiala och spektrala upplösning som krävs för att svara på frågan om sporer är levande eller döda.

RCA-transformationen förefaller dock ha utjämnat störande bakgrundseffekter. Det visar på potentialen att använda den här typen av metoder i andra sammanhang där spektra från exempelvis SWIR-området ska analyseras. Om mer data finns tillgängligt och/eller om skillnaderna mellan grupperna är mer påtaglig kan RCA eller andra metrikinlärningsmetoder troligen förbättra en klassificering genom att vikta ned den spridning i data som ej är relevant.

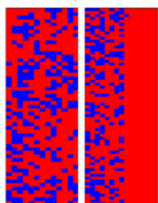


Figur 6 Medelvärdesspektra för levande (blå) och döda (röd) sporer. Vänster: rådata. Höger: data efter RCA-transformation.



Figur 7 Exempelbild av en SWIR-bild över en yta med levande sporer (komponent 1 i en principalkomponentanalys) med ett markerat område som motsvarar den del av bilden som använts för testning. Den undre delen figuren visar resultatet av klassificeringen i markerat område. Rött: bildpunkter som klassificerats som döda sporer, blått: bildpunkter som klassificerats som levande sporer.

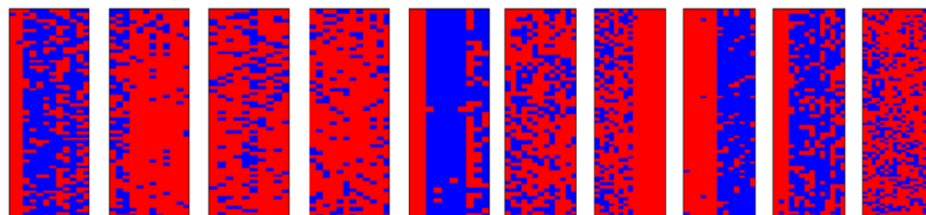
Bilder från levande sporer



Bildpunkter klassificerade som...

- ...levande sporer,
- ...döda sporer.

Bilder från döda sporer



Figur 8 Klassificering av uppsättningen med testdata. De två bilderna på den övre raden är utsnitt ur bilder från levande sporer. De åtta bilderna på den nedre raden är utsnitt ur bilder från döda sporer. Rött: bildpunkter som klassificerats som döda sporer, blått: bildpunkter som klassificerats som levande sporer.

2.6 Sammanfattning

Dagens metoder för att säkerställa att mikroorganismer är avdödade är långsamma och kräver ofta odling, vilket tar timmar eller dagar att utföra. Här har det undersökts om en snabbare analys kan göras genom att använda metrikinlärning för att klassificera SWIR-bilder från levande och döda sporer. SWIR-bilderna är insamlade med en linjeavbildande kamera, som samlar in reflektansspektra i SWIR-området och som kan avbilda proverna

på under en minut. I SWIR-området finns toppar från exempelvis vatten, lipider och protein, det vill säga ämnen som sporer består av och som kan tänkas förändras när sporer dör. Varje bild består av hundratusentals spektra med kemisk information och hypotesen är att denna information kan användas för att särskilja levande och avdöda sporer. Eftersom de spektrala skillnaderna kan förväntas vara mycket små och svårtolkade och eftersom metoden medger att förhållandevis mycket data kan samlas in på kort tid, finns det stor potential att analysera data med hjälp av maskininlärning.

Metrikinlärning innebär att likheter/olikheter, alternativt avstånd, mellan observationer beskrivs och används i någon klassificeringsalgoritm. Här har SWIR-spektra från de insamlade bilderna transformerats med hjälp av RCA, vilket är en metrikinlärningsmetod som maximerar avståndet mellan grupper samtidigt som den minimerar avståndet inom grupper. Dessa transformerade data har sedan använts i en k-NN-klassificeringsmodell.

Klassificeringen blev inte tillfredsställande och modellen kan ej användas för att klassificera sporer som levande eller döda. Det förefaller dock som att RCA-transformation kan vara effektivt för att minska irrelevant spridning i data. I det här fallet visade det sig genom att störande bakgrundseffekter kunde elimineras. Det räckte dock inte för att anpassa en tillförlitlig klassificeringsmodell till data. Detta kan bero på att skillnaderna mellan grupperna helt enkelt är för små eller på att långt fler bilder skulle behövas, inte minst av levande sporer eftersom endast två bilder fanns för den gruppen (jämfört med 8 bilder av döda sporer).

3 Avbrusning av bilder från mörkerkamera

3.1 Bakgrund

Temporalt och spatialt brus i bilder från elektrooptiska kameror begränsar bild- och videokvalitet, särskilt i scener med låg kontrast och/eller vid dålig belysning, vilket resulterar i en reducerad räckvidd för måldiskriminering (detektion, igenkänning och identifiering) och försämrade prestanda för olika målföljningsalgoritmer.

Brus uppstår i olika delar av signalbehandlingskedjan och utgörs normalt av en kombination av olika bruskällor. Skottbrus är en fundamental fysikalisk slumpmässighet som uppstår pga. partikelnaturen hos fotoner och elektroner. Mörkerströmsbrus uppstår genom termiska effekter i detektormaterialet och kan reduceras genom kylning, dock på bekostnad av större effektförbrukning, storlek och vikt. Mönsterbrus (även kallat icke-uniformitet) utgörs av känslighetsskillnader mellan pixlar. Utläsningbrus är brus från förstärkning och A/D-omvandling i utläsningselektroniken. Utöver detta kan brus även uppstå från bild- och videokompression och/eller en brusig transmissionskanal.

Avbrusning och brusreduktion är signalbehandlingsmetoder som syftar till att minska brus under bevarande av viktiga särdrag så som kanter och texturer och utan att tillföra problematiska artefakter.

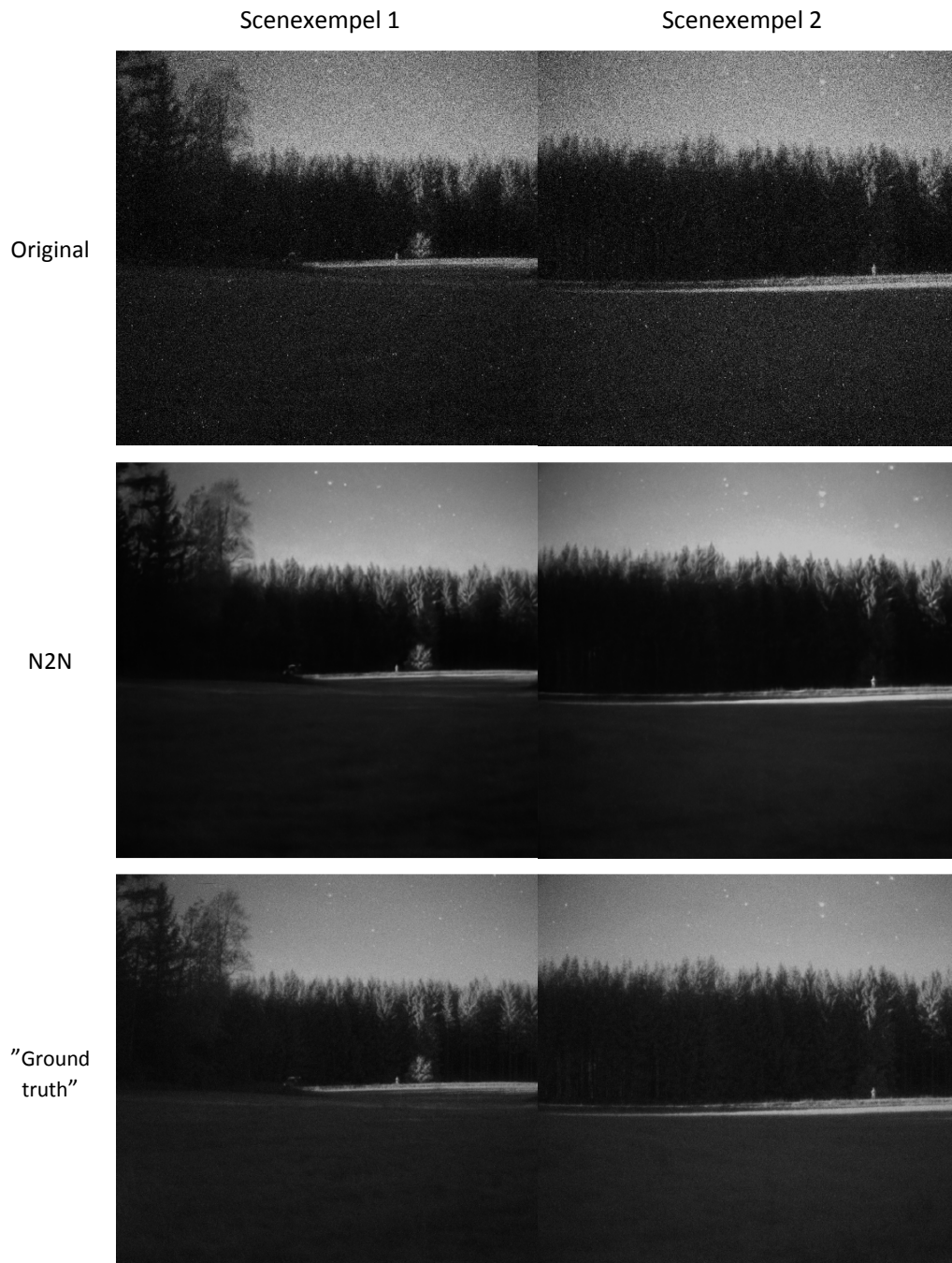
3.2 Klassisk avbrusning

Avbrusning (*eng.* denoising) är exempel på ett inverst och illa-ställt matematiskt problem utan unik lösning, med målet att återskapa en bild innan bruskontaminering. Klassiska brusreduktionsmetoder opererar antingen i den spatiala domänen eller i någon transformerad domän. I den spatiala domänen används *linjära* filter (exempelvis medelvärdesfilter eller Wiener-filter), *icke-linjära* filter (exempelvis medianfilter eller bilaterala filter) samt olika *variationsmetoder* (exempelvis regularisering med total variation eller non-local means NLM). Vanliga nackdelar med dessa filter är utsmetning av informativa särdrag såsom kanter och texturer. Genom att transformera den brusiga bilden till en annan domän, exempelvis med hjälp av Fourier-, cosinus- och wavelet-transformer, erhålls en transformerad bild där brus och särdrag har annan karaktäristik och enklare kan separeras från varandra. En vanlig benchmark-metod är BM3D (Block-Matching and 3D filtering) som opererar på 3D-stackade grupper av matchande bildfragment, vanligtvis transformerade till wavelet-domänen [17].

3.3 Avbrusning med djupinlärning

Ett alternativ till de klassiska brusreduktionsmetoderna är metoder baserade på faltande neurala nätverk (CNN: Convolutional Neural Network), där en avbildning från brusiga till brusfria bilder kan erhållas genom att träna nätverket under minimering av en optimeringsfunktion på en träningsmängd innehållandes par av brusiga och brusfria bilder ("ground truth"). Senare exempel som uppnått hög prestanda är DnCNN [18] och FFDNet [19]. Brusfria bilder kan erhållas genom att integrera ljus över längre tid, medelvärdesbilda flera korta exponeringar och/eller samla in data under bättre ljusvillkor. Sådana insamlingar är inte alltid praktiskt genomförbara och metoder har därför även tagits fram där träning sker enbart på brusiga bilder. En av de första metoderna för detta var Noise2Noise (N2N) [20], som tränar ett nätverk att hitta en avbildning mellan två brusiga bilder i ett bildpar mot en och samma scen. Metoden bygger på att det statistiska väntevärdet vid en (L2- eller L1-) minimering är detsamma oavsett om vi tränar mot brusfria eller brusiga bilder så länge scenen i bildparen är densamma eller åtminstone

kvasikonstant och så länge bruset är oberoende realiseringar ur samma statistiska fördelning. N2N har i uppdrag av FOI studerats av LiU för avbrusning av bilder insamlade från mörkerkänslig elektronbombarderande CMOS-kamera (EBCMOS), se exempelresultat från genomförda experiment i Figur 9. [21]

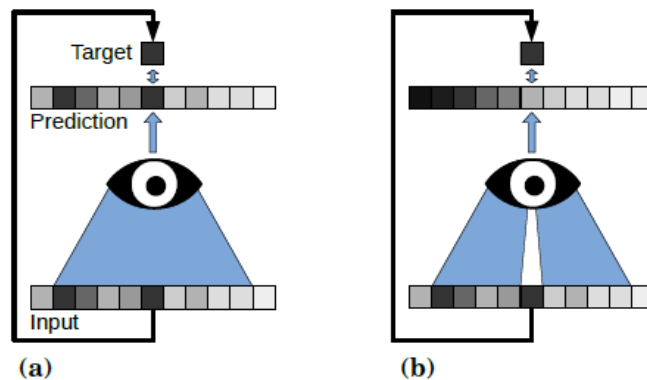


Figur 9 Bildjämförelser i två scener från arbete genomfört av LiU 2018. (övre) brusig originalbild från mörkerkamera. (mitten) N2N-avbrusad bild. (nedre) temporalt medelvärdesbildad bild ("ground truth") [6].

En nackdel med N2N är kravet på ordnade bildpar, vilket kräver insamling med suboptimalt högre bildtakt och/eller mot en kvasistationär scen. En senare utvidgning av konceptet är metoderna Noise2Void (N2V) [21] och Noise2Self (N2S) [22], med träning mot en bilddatamängd bestående av enskilda brusiga bilder, dvs. ej nödvändigtvis ordnade i

bildpar. Metoderna bygger på tre statistiska grundantaganden: 1) att signalen inte är pixelvist oberoende (dvs. att pixlar är predikterbara från sin omgivning), 2) att bruset är betingat pixelvist oberoende givet signalen (saknar pixelkorrelation), samt 3) att bruset har ett nollmedelvärde (en medelvärdesbildning över tid resulterar i den brusfria signalen). Även om metoden inte kan förväntas ge samma prestanda som N2N eller traditionell CNN-avbrusning (som innehåller mer information under träning), är den attraktiv eftersom den är enklare att implementera och mer användbar för rörliga sekvenser mot dynamiska scener.

Både N2V och N2S är självannoterande metoder som bygger på s.k. *blind-spot-nätverk* (se Figur 10), där en pixel i nätverkets s.k. receptiva fält (området i indatarummet som ett särdrag i nätverket påverkas av) medvetet plockas bort och där målet med träningen sedan är att återskapa denna blind-spot. Ett sätt att åstadkomma en blind-spot är att kopiera in ett slumpmässigt pixelvärde från omgivningen. Typiskt förblindas några enstaka procent av alla pixlar i bilden på detta sätt. [21]



Figur 10 I ett konventionellt nätverk (a) predikteras en individuell pixel av ett receptivt fält (patch med pixlar under blå kon) vilket degenererar nätverket vid träning med användning av samma brusiga data både som input och output (nätverket lär sig en trivial identitet). I ett s.k. blind-spot nätverk (b) exkluderas pixeln själv i det receptiva fältet, vilket förhindrar den triviala identitetsinlärningen.

Både N2V och N2S använder sig av en s.k. U-net-arkitektur, med lite olika antal lager och parametersättningar. I övrigt är metoderna mycket lika varandra och ska egentligen ses som två implementationer av en och samma grundmetod. N2V och N2S är kodade i Python och kan laddas ned via github¹. Tabell 1 visar de viktigaste nätverks- och träningsparametrarna.

¹ N2V: <https://github.com/juglab/n2v> (Tensorflow)

N2S: <https://github.com/czbiohub/noise2self> (PyTorch)

Tabell 1 Parametersättningar i de två brusreduceringsmetoderna Noise2Void (N2V) och Noise2Self (N2S).

	N2V	N2S
Nätverk	U-net	U-net
Antal lager	2	4
Faltningskärna	3x3	2x2
Kärnor/lager	32	32/64/128/256
Aktiveringsfunktion	Linjär	Selu
Optimeringsmetod	Adam	Adam
Learning rate	0.004	0.001
Loss-funktion	MSE	MSE

3.4 Experiment

Ett avbrusningsförsök har genomförts med ett dataset insamlat från en mörkerkamera. I experimentet har N2V och N2S jämförts med några traditionella avbrusningsmetoder, med hjälp av några vanligt förekommande referensbaserade och referensfria bildkvalitetsmått.

3.4.1 Träningsdata

Träningsdata utgjordes av ett insamlat dataset från en EBCMOS-kamera (Electron-Bombarded CMOS) från Photonis i 1280x1024 pixlar upplösning (9.7 μm pixelavstånd). Kameran är känslig i våglängdsområdet 400-900 nm, dvs. i det visuella och nära infraröda våglängdsområdet (och tillika det typiska området för bildförstärkare). Kameran var försedd med en 25 mm lins med lågt bländartal (F/0.95) för optimering av ljusinsläpp. Kameran rullade i 30 bilder per sekund och levererade bilder med 10-bitars upplösning. För träningens skull justerades dock dynamiken ned till 8 bitar för att passa in i kodimplementationerna. Träningsdatasetet bestod av totalt ca 2600 bilder från tre olika mätillfällen, med både inom- och utomhusscener, varav 240 användes för test och validering. Bilderna var tagna från flera olika avstånd med en förgrund som bestod av en stillastående och gående människa med olika handhållna objekt (både militära och civila) i olika aspektvinklar. Det bör noteras att bakgrunden ej varierade nämnvärt i sekvenserna och att korrelationen mellan bilder därför får anses vara mycket hög. Några bildexempel ur träningsdatamängden kan ses i Figur 11.



Figur 11 Fyra brusiga bildexempel ur träningsdatabasen.

3.4.2 Avbrusningsmetoder

Avbrusning skedde med sex olika metoder: **A)** temporalt medelvärde av 30 bilder (kunde endast användas i stationära scener där figuren stod still), **B)** medianfiltrering, **C)** Non-local means (NLM), **D)** Block-Matching and 3D filtering (BM3D), **E)** Noise2Void (N2V) och **F)** Noise2Self (N2S).

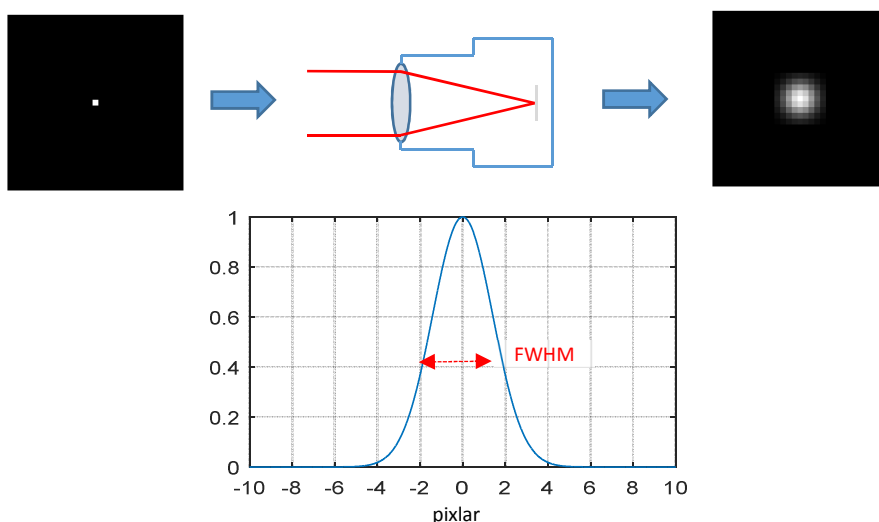
Vid medianfiltreringen ersattes varje pixel av medianvärdet från en lokal 3x3-omgivning. NLM är en medelvärdesbildande metod där varje pixel ersätts med ett viktat icke-lokalt medelvärde (ur hela eller delar av bilden), med viktning efter hur lika pixlar (med omgivning) är varandra. [23] I använd MATLAB-implementering (imnImfilt i Image Processing Toolbox) användes sök- och jämförelsefönster som var 21x21 respektive 5x5 pixlar stora. BM3D förstärker gleshet i en transformdomän och använder s.k. kollaborativ filtrering av Wavelet-transformerade grupper av bildpatchar. [24] Både NLM och BM3D används ofta som referensmetoder pga. sin höga prestanda, men är också relativt beräkningskrävande.

För N2V och N2S användes standardmässiga parametersättningar, i huvudsak enligt Tabell 1. Vid träning med N2V användes 64x64-patchar av indatabilderna, med 1.6% slumpmässigt förblindande pixlar i varje patch och med 64 st sådana patchar per träningsbatch. I träningen med N2S användes något större 256x256-patchar (slumpmässigt utplacerade), med 16 bilder per träningsbatch. För båda metoderna tog träningen ca 2-3 timmar innan konvergens åstadkoms, på en PC (Windows 10) med ett Nvidia GTX 1080 grafikkort. Utvärderingen (avbrusningen) kunde sedan göras med ca 10-15 bilder per sekund på samma hårdvara.

3.4.3 Bildkvalitetsmått

För jämförelse av resultat från de olika avbrusningsmetoderna användes sex olika referensbaserade och referensfria bildkvalitetsmått. De referensbaserade måtten utgjordes av minsta kvadratfelet MSE (*eng.* mean-squared error), kvoten mellan signal- och bruseffekt PSNR (*eng.* peak signal-to-noise ratio) samt den perceptionsbaserade SSIM (structural similarity). Använda referensfria mått var NIQE (Naturalness Image Quality Evaluator), PIQE (Perception-based Image Quality Evaluator) samt BRISQUE (Blind/Referenceless Image Spatial Quality Evaluator). För mer information om dessa se exempelvis MATLABs Image Processing Toolbox, där de alla finns implementerade.

Utöver dessa uppmättes även punktspridningsfunktionen (PSF), vilket beskriver skärpan och upplösningsförmågan hos det avbildande systemet. Detta gjordes genom att mäta kantresponsen ur bilder tagna på kontrasttavlor i scenen (exempel kan ses i nedre högra delen av Figur 11).



Figur 12 Punktspridningsfunktionen (PSF) beskriver impulssvaret hos det avbildande systemet, dvs. hur en väldefinierad punkt i scenen avbildas till en något mer utsmetad fläck i bilden. PSF mätts ofta genom FWHM-värdet (fulla bredden på halva höjden) i antal pixlar.

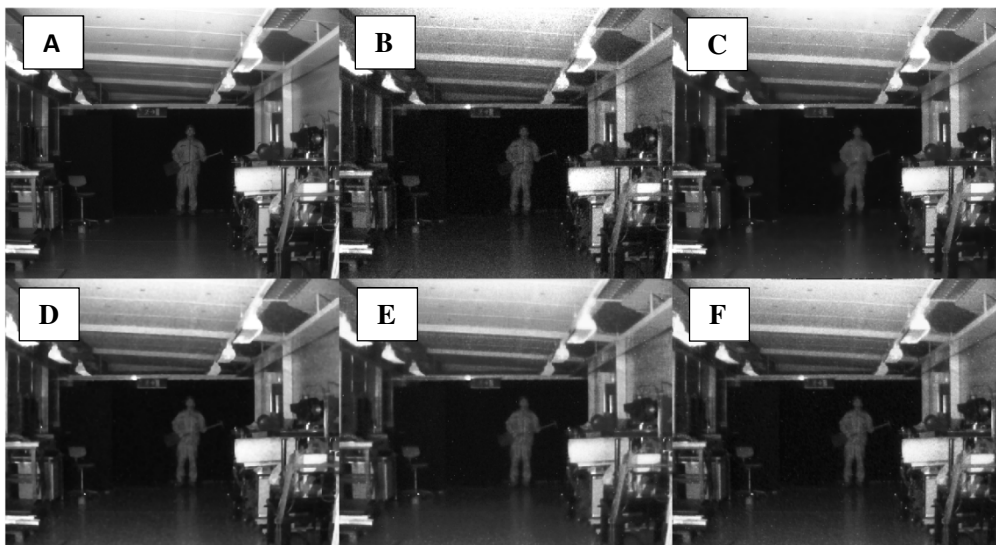
3.5 Resultat

Resultaten från avbrusningsförsöken redovisas här i form av bildexempel och resultattabeller med nämnda bildkvalitetsmått.

3.5.1 Bildexempel

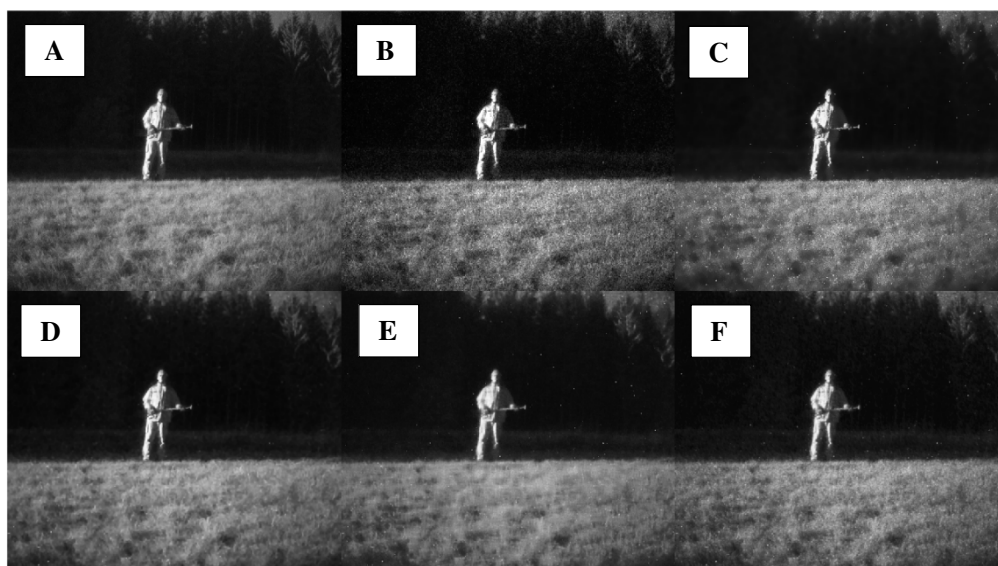
I Figur 13-Figur 16 nedan visas några bildexempel efter avbrusning med de använda metoderna. Alla metoderna reducerar brus, jämför med originalbilderna i Figur 11, men med lite olika slutresultat och karaktäristik. I dessa jämförelser är medelvärdesbildning (A) att betrakta som referensbilder för ett önskvärt slutresultat ("ground truth").

Av de övriga upplevs generellt BM3D (D) bäst i brusreduceringsavseende, tätt följd av N2S (E) och N2V (F) som är relativt likvärdiga. Både medianfiltrering (B) och NLM (C) har relativt tydligt kvarstående brus, men medianfiltreringen är klart skonsammare mot texturer och detaljer än vad NLM är. Vad gäller skärpa upplevs median- och BM3D-filtrerade bilder skarpare än övriga metoder.



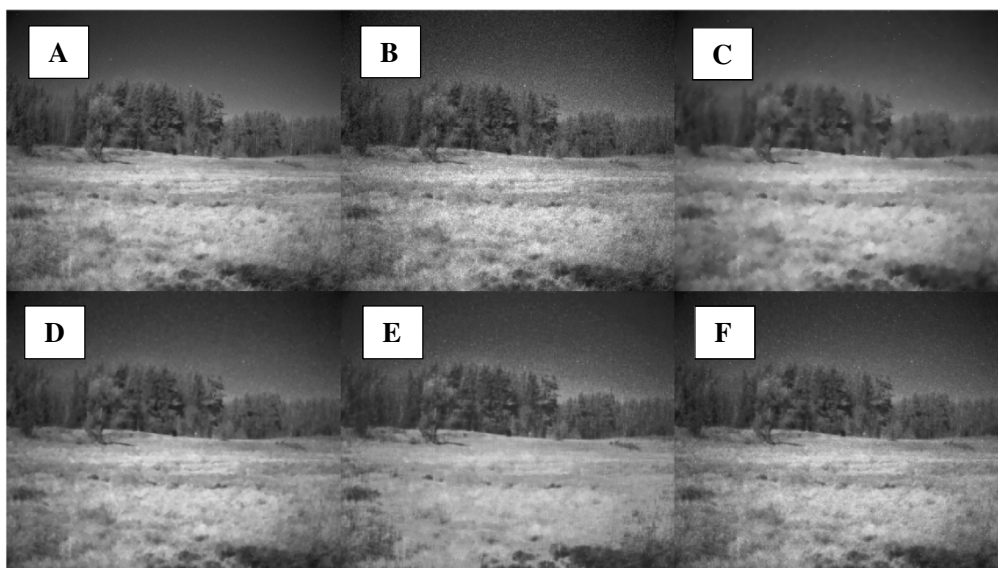
Figur 13 Avbrusningsresultat från inomhusscen.

(A: medelvärde över 30 bilder, B: medianfilter, C: NLM, D: BM3D, E: N2V och F: N2S)



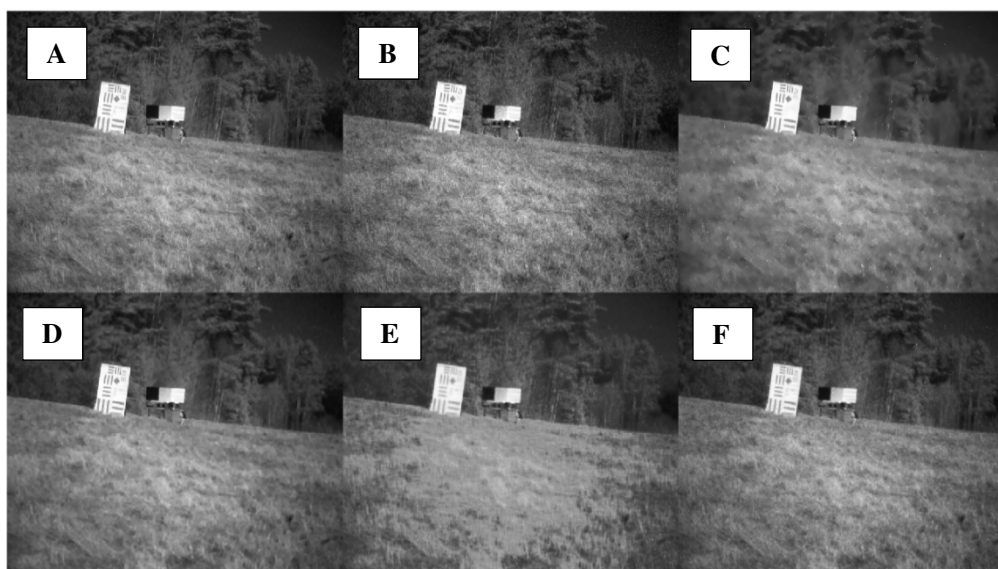
Figur 14 Avbrusningsresultat från utomhusscen 1.

(A: medelvärde över 30 bilder, B: medianfilter, C: NLM, D: BM3D, E: N2V och F: N2S)



Figur 15 Avbrusningsresultat från utomhusscen 2.

(A: medelvärde över 30 bilder, B: medianfilter, C: NLM, D: BM3D, E: N2V och F: N2S)



Figur 16 Avbrusningsresultat från scen med referenstagor.

(A: medelvärde över 30 bilder, B: medianfilter, C: NLM, D: BM3D, E: N2V och F: N2S)

3.5.2 Bildkvalitetsmått

En jämförelse av de utvalda referensbaserade och referensfria bildkvalitetsmåten kan ses i Tabell 2. Som referens användes här den temporala medelvärdesbildningen (metod **A**). Utöver dessa uppmättes även skärpan i form av fullbredden på halva höjden hos punktspridningsfunktionen (PSF_{FWHM}). Bäst resultat är fetmarkerade i respektive kategori. Det kan ses att BM3D presterar bäst för de referensbaserade måten, tätt följd av N2V och N2S. För de referensfria måten är resultaten lite mer varierade, men både N2V och N2S står sig väl i jämförelse med BM3D och NLM. Vad gäller bibehållande av skärpa är NLM och BM3D jämförbara med medianfiltrering (ca 0.5-0.7 pixlar försämring i förhållande till rådata), medan både N2V och N2S introducerar en upplösningsförsämring som är ca dubbelt så stor.

Tabell 2 Jämförelse av bildkvalitetsmått (gula: referensbaserade, blå: referensfria) och PSF.

Metod	MSE	PSNR	SSIM	NIQE	PIQE	BRISQUE	PSF _{FWHM}
Rådata	955 ± 734	19,8 ± 3,9	0,26 ± 0,12	15,3 ± 2,0	69,5 ± 5,5	45,3 ± 2,1	2,3 ± 0,7
Medelvärde (A)	-	-	-	9,5 ± 1,5	39,4 ± 14,9	34,1 ± 6,6	2,7 ± 0,1
Median (B)	285 ± 251	25,5 ± 4,4	0,46 ± 0,16	4,9 ± 0,8	20,3 ± 8,8	31,1 ± 5,0	3,0 ± 0,7
NLM (C)	171 ± 124	27,1 ± 3,5	0,63 ± 0,13	6,8 ± 0,6	25,8 ± 12,3	25,0 ± 10,0	2,9 ± 0,6
BM3D (D)	78 ± 52	30,4 ± 3,5	0,64 ± 0,14	5,8 ± 0,7	72,6 ± 14,1	56,2 ± 3,6	3,0 ± 0,4
N2V (E)	110 ± 53	28,2 ± 2,2	0,61 ± 0,14	9,6 ± 2,2	10,2 ± 4,7	33,0 ± 2,9	3,6 ± 0,5
N2S (F)	99 ± 67	29,4 ± 3,4	0,64 ± 0,14	4,4 ± 0,4	16,2 ± 8,8	33,9 ± 3,6	3,6 ± 0,5

3.6 Slutsatser och diskussion

Resultaten visar att N2V/N2S står sig väl vad gäller brusreducering, i förhållande till BM3D och NLM som båda anses som "state-of-the-art" inom traditionell avbrusning. Avbrusningen sker dock inte utan kompromisser. NLM presterar tydligt sämre vad gäller bevarande av texturer, fortfarande något oklart varför. Upplösningförsämringen om ca 1-1.5 pixlar (i termer av PSF_{FWHM}) hos både N2V och N2S är också lite väl stor och bilder avbrusade med metoderna upplevs ofta som något mjuka och oskarpa. Aktuella parametersättningar, både för de mer traditionella metoderna (främst NLM och BM3D) samt för N2V/N2S, kan sannolikt förbättras och optimeras mot det aktuella datasetet och den aktuella kameran än vad som hunnits med i detta försök. För NLM handlar det om att välja sök- och jämförelsefönster av rätt storlek. För BM3D handlar det bland annat om att estimeras spektraltäthet hos bruset, vilket i den aktuella användningen förenklades till att använda standardavvikelse i bild (vilket innebär ett antagande om vitt brus). För N2V och N2S handlar det om nätverks- och träningsrelaterade parametrar så som listade i Tabell 1 (dvs. antal lager, storlek på faltningskärnor, antal kärnor per lager, etc.).

I samband med presenterade slutsatser bör det också starkt påpekas att bildmaterialet som använts egentligen är för litet, både sett ur mängden bilder samt i dess variation. Ett större bildunderlag är önskvärt både ur träningsynpunkt samt för erhållande av statistiskt säkerställda jämförelseresultat. Insamling av bilder under olika ljusbetingelser och i olika bakgrundsmiljöer är dock ett företag som kräver stor arbetsinsats, under arbetstid som ofta är att betrakta som relativt obekvämt.

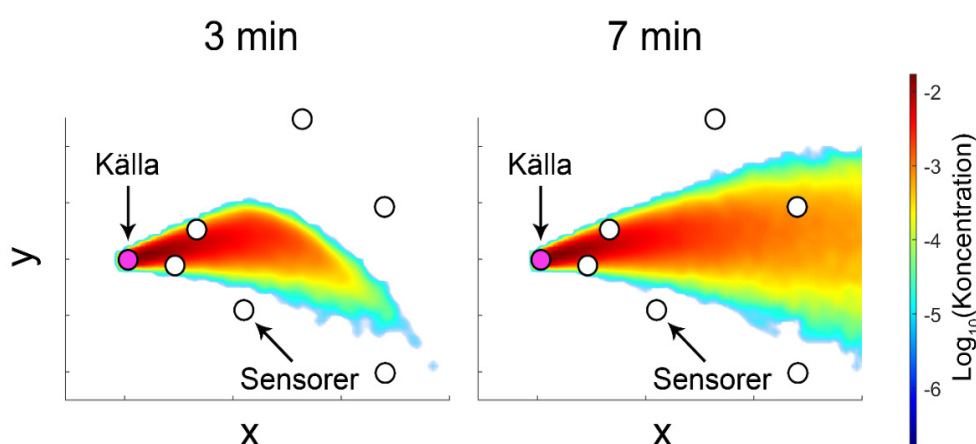
Avslutningsvis kan det nämnas att metodiken bakom N2V och N2S utvecklats vidare sedan de första publikationerna. En senare förbättring av N2V presenterar en utökad och modifierad träningsmetod kallad Probabilistic Noise2Void (PN2V) [21], som introducerar en godtycklig brusmodell som kan estimeras ur bildmaterialet. Författarna visar att metoden kan konkurrera med traditionell CNN-baserad träning som använder sig av brusfria referensbilder (så som DnCNN, m.fl.) samt med träningsmetoder baserade på brusiga bildpar (N2N).

4 Maskininlärning för invers modellering av atmosfärisk spridning

4.1 Bakgrund

Ämnen kan spridas från många typer av källor genom att transporteras av strömningar i den omgivande luften. Detta kan ske på små skalor inomhus, lokalt inom en stad, på längre distanser nationellt och även globalt i vissa fall. Beroende på ämnestyp, och mängd därav, kan utsläppet orsaka varierande skadeeffekt på människor, samhälle eller miljö. Det finns ett samhällsintresse av att förstå och inneha en förmåga att hantera dessa frågor. En viktig komponent för att uppnå den kunskapen och förmågan är modellering och simulering av det fysikaliska spridningsförloppet.

Atmosfärisk spridning av aerosoler och gaser är en komplex process som ofta baseras på en Langevin-ekvation. Det stokastiska inslaget innebär att det är omöjligt att förutse ett deterministiskt förlopp vilket betyder att en simulering ger upphov till en realisering av en sannolikhetsfördelning som är okänd *a priori*. De drivande strömningarna i lufrummet beror på vilken skala som hanteras men kan generellt delas upp i ett konvektionsfält samt ej upplöst turbulens. Det är den senare termen som beskrivs som en stokastisk term i Langevin-ekvationen i spridningsmodellen. För en inert substans kan koncentrationsfält simuleras utifrån modellen givet källterm, vindfält och turbulensbeskrivning. Figur 17 visar resultat från en simulering med FOI:s spridningsmodell LPELLO som är utvecklad för att hantera atmosfärisk spridning på den lokala skalan, se [25, 26].



Figur 17. Exempel på simuleringsresultat av atmosfärisk spridning givet en källa i origo. Den vänstra panelen visar plymens utbredning efter tre minuter medan den högra panelen visar plymen i sitt jämviktsläge, givet en konstant källstyrka och konstant meteorologi, som har uppstått efter sju minuter. Sex stycken slumpmässigt utplacerade sensorer mäter punktvisa koncentrationsvärden. Två av dem befinner sig inom plymen i vänstra panelen och en tredje sensor inkluderas i ett senare skede av förloppet vilket syns i den högra panelen.

Koncentrationsnivåer kan mätas med olika typer av sensorer där punktmätningar i rummet är vanligast. Sådana instrument samlar antingen ett luftprov över en viss tid för analys eller så mäter den upp en kontinuerlig koncentrationskurva. Sensorer används ofta för att tidigt uppfatta möjliga hot och ge information till beslutsfattare.

Antag att ett skyddsvärt objekt ska skyddas och att ett antal sensorer är utplacerade i närområdet för att ge information om potentiellt farliga utsläpp. En händelse äger rum där vissa sensorer registrerar förhöjda värden utan någon uppenbar anledning. Problemet för ansvariga för det skyddsvärda objektet kan då formuleras som: *givet ett*

visst antal sensorsignaler, var är källan och hur stark är den? Detta är ett delområde inom spridningsberäkningar som kallas inversmodellering.

4.2 Invers modellering

Inversa atmosfäriska spridningsproblem löses oftast med en *adjungerade spridningsmodeller*, se [27, 28], vilket innebär att en modellplym anger utfallet av en given *mätning* (position och tid), för ett stort antal olika källpositioner och källtider. Detta till skillnad från konventionella spridningsmodeller (även kallade framåtspridningsmodeller), som anger utfallet av en given *källa* (position och tid) för ett stort antal olika mätpositioner och mättidpunkter. För att skatta källparametrar måste en lämplig *källhypotes* formuleras, dvs. källans utseende i rum och tid måste ansättas. I källansatsen ingår parametrar i en lämplig *källparameterrymd*. Det inversa spridningsproblemet består av att bestämma en punkt i källparameterrymden vars modellresultat ger bäst överensstämmelse med observerade mätdata. Överensstämmelsen kan mätas på olika sätt. Grovt sett har två metoder använt: optimerings (regressions)-metoder, se t.ex. [29, 30], samt statistiska (Bayesiska) metoder, se t.ex. [31]. Noga räknat så överlappar metoderna varandra då många statistiska metoder mynnar ut i optimeringsproblem, till exempel *maximum likelihood*-skattningar. Den Bayesiska tolkningen innebär att man skattar en *sannolikhetsfördelning* på källparameterrymden, vilken har fördelen att ge en uppfattning om *osäkerheten* i källskattningen. Då källparameterrymden (åtminstone för tidsberoende källor) är högdimensionell så krävs effektiva Monte-Carlometoder för detta, se t.ex. [32]. Optimeringsmetoder kan tyckas generellare, men kan i själva verket tolkas som en *maximum likelihood* – skattning av en Bayesisk modell genom lämplig transformation av optimeringsmodellens målfunktion.

Om man beaktar källskattningsproblemet enbart som ett regressionsproblem, dvs. minimerar ett mått på diskrepansen mellan observerade data och motsvarande modelldata för givna källparametrar, så blir lösningen i regel mycket känslig för variationer i indata, dvs. problemet är *illa ställt*. För att erhålla en robust lösning krävs någon form av *regularisering*, vilket innebär att man viktar samman önskvärda *a priori*-egenskaper hos lösningen med ovan nämnda diskrepans. I optimeringsfallet innebär detta att man adderar regulariseringstermer till målfunktionen som straffar brott mot de önskade *a priori*-egenskaperna. I det Bayesiska fallet innebär det att man väljer en lämplig *a priori* sannolikhetsfördelning på källparameterrymden. Hur detta görs styrs av ett antal *regulariseringsparametrar*. Inom invers spridningsmodellering används väljs oftast parametrarna baserat på en subjektiv bedömning av lösningens utseende, vilket naturligtvis är en brist. I stället bör någon form av systematisk *parametervalsmetod* användas. Grovt sett kan dessa metoder delas in i tre grupper, se:

- A priori parametervalsmetoder baserade på brusnivå och släthetsvillkor på lösningen, jfr. [33].
- A posteriori parametervalsmetoder baserade på brusnivå och indata.
- A posteriori parametervalsmetoder baserade enbart på indata, s. k. *heuristiska parametervalsmetoder*.

En översikt av vanliga parametervalsmetoder finns i [34]. För ansatser till användning av maskininlärning för parameterval, se [35].

4.3 Maskininlärning

Två helt olika sätt att angripa detta problem är identifierade. Maskininlärning skulle kunna användas för att optimera parametrarna i metoden med adjungerande fält som

beskrivs ovan. Ett alternativ är att helt bortse från den metodiken samt andra fysikaliska modeller, och istället skapa ett neuralt nätverk som på ett mer abstrakt sätt kopplar relevant indata till en sannolikhetsfördelning av källans egenskaper. Även om båda metoderna är intressanta har den senare metoden valts inom detta projekt. Motivering för detta val är inte att denna metod nödvändigtvis är mer lovande, utan för att det är intressant att testa en metod som skiljer sig på en grundläggande nivå från den första, sedan tidigare mer utforskade, metoden.

4.3.1 Metodbeskrivning

Atmosfärisk spridning är beroende av ett stort antal parametrar och kan modelleras på många olika sätt genom att inkludera olika delar av de fysikaliska processerna som äger rum. I denna studie exkluderas all tidsvariation för att renodla problemet så långt det är möjligt. Det innebär att problemet studeras när plymen uppnått jämviktsläge vilket här definieras som att sannolikhetsfördelningen för koncentrationsfälten inte ändras med tiden. Detta är inte helt synonymt med ett statiskt koncentrationsfält eftersom fluktuationer som följd av de stokastiska inslagen i partikelmodellen LPELLO fortfarande finns närvarande. Sensorsignaler är normalt sett brusiga vilket beror på störningar från bakgrunden samt osäkerheter i själva sensorn. Dessa faktorer inkluderas inte i denna analys utan vi antar att miljön och sensorerna är ideala. Vi låter spridningsmodellens stokastiska variationer, i kombination med det numeriska bruset som uppstår i renderingen av koncentrationsfältet (orsakat av ändligt antal modellpartiklar) representera all variation för plymen i jämviktsläge. Helt statisk plym skulle kunna uppnås genom att tidsmedelvärda bort detta brus. Men fluktuationer bedöms innehålla ett mervärde för att variera träningsdatat och behålls därför intakt. Sammanfattningsvis, momentana koncentrationsfält för plymen används efter att den har uppnått jämviktsläge, se högra panelen i Figur 17.

4.3.2 Generering av träningsdata

4.3.2.1 Simulering av en plym

För att sätta upp en träningsmiljö för maskininlärning krävs vissa antaganden och begränsningar. Vi antar att vi har en källa med konstant källstyrka. Källan har en spatial utbredning som kan antas vara punktformig och är placerad strax över marken. Spridning sker på lokal skala vilket gör att vi kan anta analytisk och horisontellt isotropisk meteorologi [36]. Endast stabil meteorologisk skiktning används (Pasquill klass E). Vi antar en platt mark utan hinder med en låg ytråhet. Alla sensorer placeras på samma höjd som källan. Vindriktningen är känd a priori vilket gör att, i kombination med horisontellt isotropisk geometri och meteorologi, systemet alltid kan roteras därefter. Vindriktningen kan därmed hållas konstant mellan alla fall. Detaljerad information om vilka simuleringsparametrar som användes ges i Tabell 3. Spridningssimuleringar genomfördes med FOI:s modell LPELLO [25, 26].

Tabell 3 Specifikation av de viktigaste parametrarna som används vid simulering av koncentrationsfältet med LPELO.2

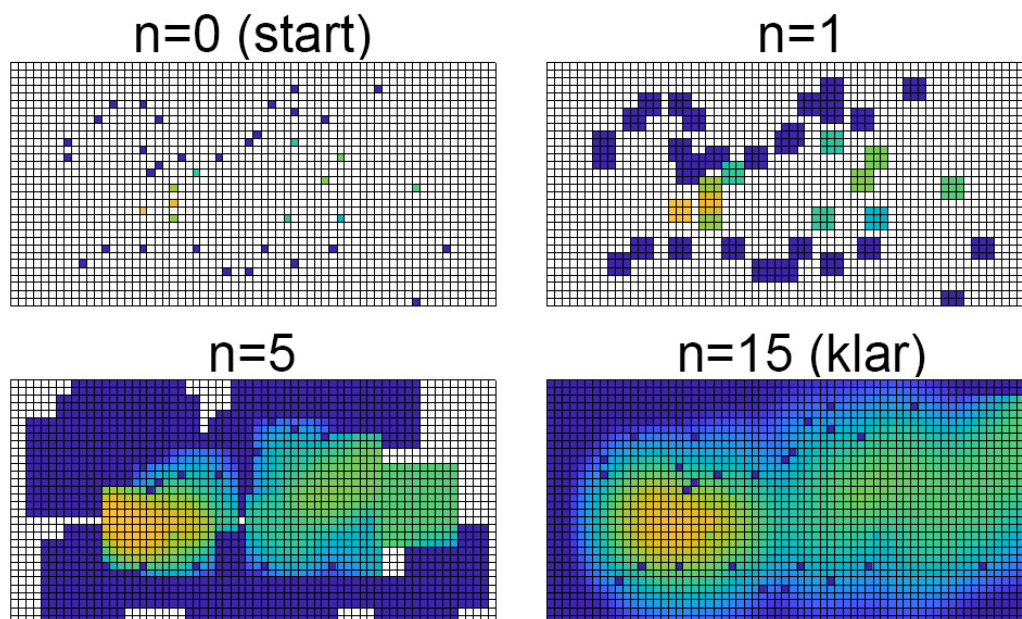
PARAMETRAR	VÄRDE
Källans utbredning	1.0 m
Källstyrka	1.0 kg/s
Källans position	(0.0, 0.0, 2.0) m
Sensorernas vertikala position	2.0 m
Antal modellpartiklar	1.000 per sekund
Ytråhet	0.01 m
Atmosfärisk stabilitet	E
Vindhastighet vid höjden 10 m	3.0 m/s
Utsläppsintervall	600 s
Cellstorlek	10 m

4.3.2.2 Komplettering av rasterbilder

Koncentrationsplymer kan betraktas som stillbilder och användas för att konstruera indata till det neurala nätverket. Genom att dela in den spatiala domänen i ett rutnät skapas en rasterbild. Varje cell i rutnätet motsvarar en pixel i bilden, där koncentrationsvärden i varje cell motsvarar färgen på pixlarna (mer korrekt är egentligen att se koncentrationen som en nyans i en bild i gråskala). Majoriteten av alla pixlar i indatabilden är okända medan punktmätningar från sensorer ger information om koncentrationsvärdet i de rutnätsceller de är placerade i. Förhållandet mellan kända och okända värden är sådant att bara cirka 1% av pixelvärdena är kända. Den låga andelen kända pixelvärden kan upplevas som nedslående men man får komma ihåg att det finns en stark korrelation i pixelvärdena så utifrån problemets struktur finns det mer dold information i dessa glesa rasterbilder. Det uppstår tekniska svårigheter med att hantera bilder med en majoritet okända pixelvärden. För att hantera detta infördes en metod för att fylla hela bilden med data. Med andra ord, varje pixel med okänt värde tillsätts ett värde baserat på den tillgängliga informationen. De faktiska värdena är okända men en metod som baseras på den inneboende korrelationen mellan närliggande punkter implementerades. Metoden använder sig av en iterativ process enligt:

1. De pixlar som har värden från sensorer låses och ändras aldrig.
2. Alla övriga pixlar tillsätts ett värde som motsvarar medlet på samtliga omgivande celler som har värden.
3. Repetera steg 2 tills bilden är fylld.

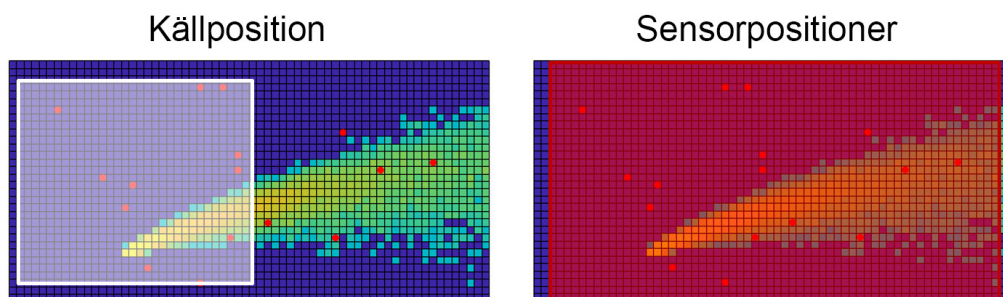
Metoden innebär att pixlar utan värden som angränsar till kända pixlar ansätts värden i första iterationen. I nästa iteration ansätts pixlar som angränsar även till dessa pixlar. Notera att de pixlar som redan tilldelats värden också uppdateras vid varje iteration. Bilden fylls stegvis upp tills den är komplett. Figur 18 visar hur detta förlopp ser ut för ett exempel.



Figur 18 Ett exempel på hur den iterativa metoden för att fylla rasterbilderna fungerar. Uppe till vänster visas startdata med kända pixelvärden där sensorer var placerade. Efter en iteration ($n=1$) har angränsande pixlar ansatts medelvärden från alla angränsande pixlar med värden, uppe till höger. När ytterligare fyra iterationer har genomförts erhålls panelen nere till vänster. I detta fall krävdes det 15 iterationer för att fylla hela bilden och slutresultatet visas nere till höger. Figuren visar koncentrationvärden i logaritmisk skala.

4.3.2.3 Permutering av träningsdata

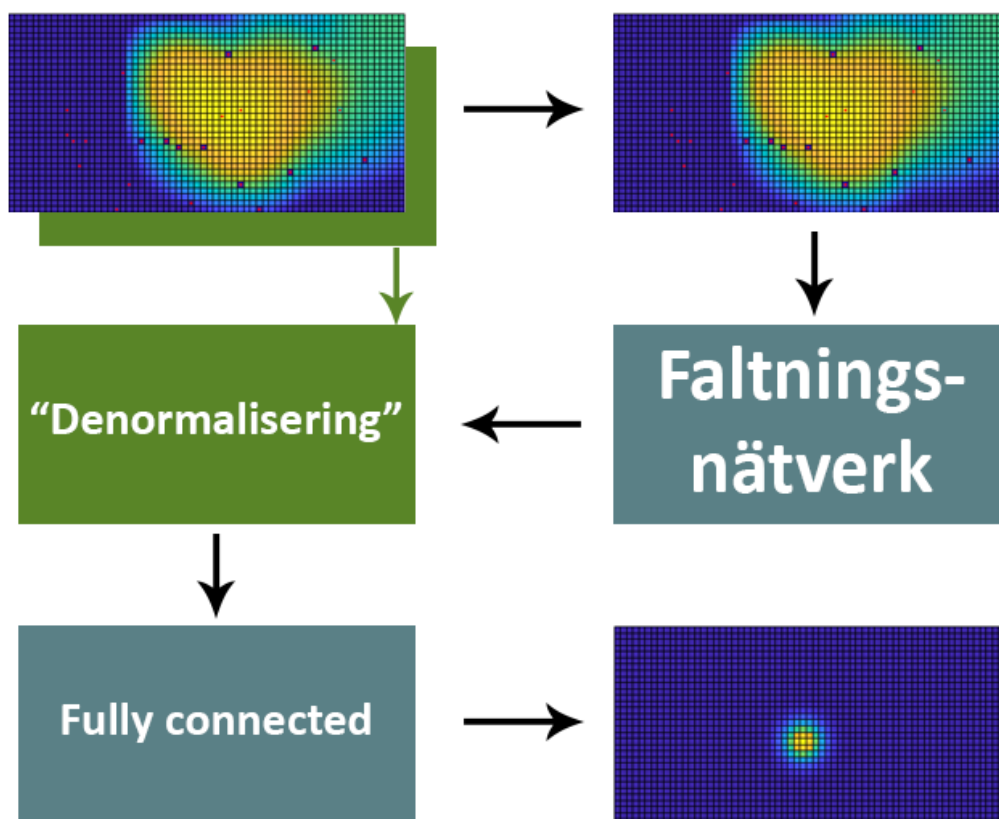
En spridningssimulering tar några minuter att genomföra men skiljer sig endast åt genom de stokastiska fluktuationer som härrör från modellen samt den numeriska renderingen. Nya tränings exempel kan konstrueras från en simulerad plym genom att byta ut antalet sensorer och deras positioner samt genom att variera källans position. De tillåtna områdena för källan och sensorerna visas i Figur 19. Uppsättningen av sensorer har ett mycket omfattande antal möjliga permutationer vilket innebär att en stor domän med syntetisk annoterad träningsdata kan skapas för träning. Antal sensorer tilläts variera inom intervallet 15-25. Fyra olika simulerade plymer användes vilka tillsammans med variationer av källpositioner och sensorer gav upphov till totalt 10 000 fall. Av dessa användes 80% till träning och 20% till validering.



Figur 19 Positionen på källan samt positionerna på sensorerna slumpades med en likformig sannolikhetsfunktion över den vita respektive röda arean.

4.3.2.4 Maskininlärning

En passande metod för denna typ av problem är faltningsnätverk (CNN). Utöver faltningsnätverket inkluderas några ytterligare steg som illustreras i Figur 20. Tillsammans utgör de ett system för maskininlärning anpassad för denna studie i inversmodellering av atmosfärisk spridning.



Figur 20 En översiktlig beskrivning av systemet för maskininlärningen. I centrum finner vi ett faltningsnätverk som ofta används för bildanalyser.

Figur 20 visar överskådligt de viktigaste komponenterna i systemet och hur arbetsflödet ser ut. Flödet startar uppe till vänster med en bild som indatas som innehåller två kanaler. Den första kanalen innehåller förbehandlad koncentrationsdata. Den andra kanalen innehåller metadata som vindhastighet och källstyrka. Dessa parametrar användes inte i denna studie men är inkluderade för att ge möjligheten till vidareutveckling. Den första kanalen med koncentrationsdata behålls och skickas in i faltningsnätverket. Detta innehåller tre nivåer av faltning. Vid varje nivå sker även en normalisering och en ReLu tillämpas följt av MaxPooling. Systemet är symmetriskt på det sätt att omvänd behandling med samma storlek på faltningskärnor används för att avfalta systemet tillbaka till en målbild i samma format som indatabilden. Här finns en möjlighet att tillämpa den informationen som extraherades i första steget. I denna studie händer det ingenting i detta steg. Sedan följer ett fullt kopplat lager. Som utdata producerar nätverket en ny rasterbild där varje pixel tolkas som sannolikheten för källpositionen. Avslutningsvis hittar vi ett regressionslager som jämför systemets resultat med den kända källpositionen. Den faktiska källpositionen utgörs av endast en pixel. En heuristik metod tillämpades för att underlätta träningen av nätverket. Istället för att målbilden endast ska utgöras av en enda korrekt pixel ansätts en Gaussisk sannolikhetstäthet över den kända

källpositionen. Detta innebär att systemet får en mjuk gradient att anpassa systemets parametrar mot.

De funktionella komponenterna i varje träningsexempel är alltså:

Indata I = Bild på koncentrationsfältet där pixlarna med okända värden har tilldelats värden med den iterativa processen som har beskrivits ovan. I en andra kanal kan metadata inkluderas.

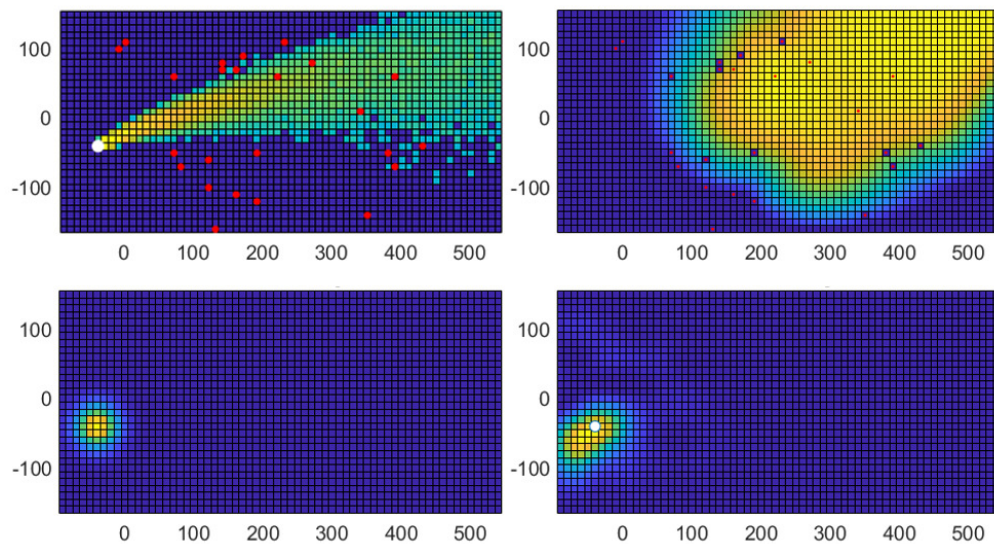
M = Målbild vilket är en bild med Gaussisk fördelning centrerad över källans position.

Utdata S = Rasterbild med sannolikhetstäthet för position i varje cell som ska motsvara M.

Systemet implementerades med hjälp av MATLAB:s Deep Learning Toolbox och som lösare användes sgdm (stochastic gradient descent with momentum). Träningen tog i storleksordningen timmar att genomföra. Med ovanstående beskrivna metodik konvergerade lösningen på ett stabilt sätt.

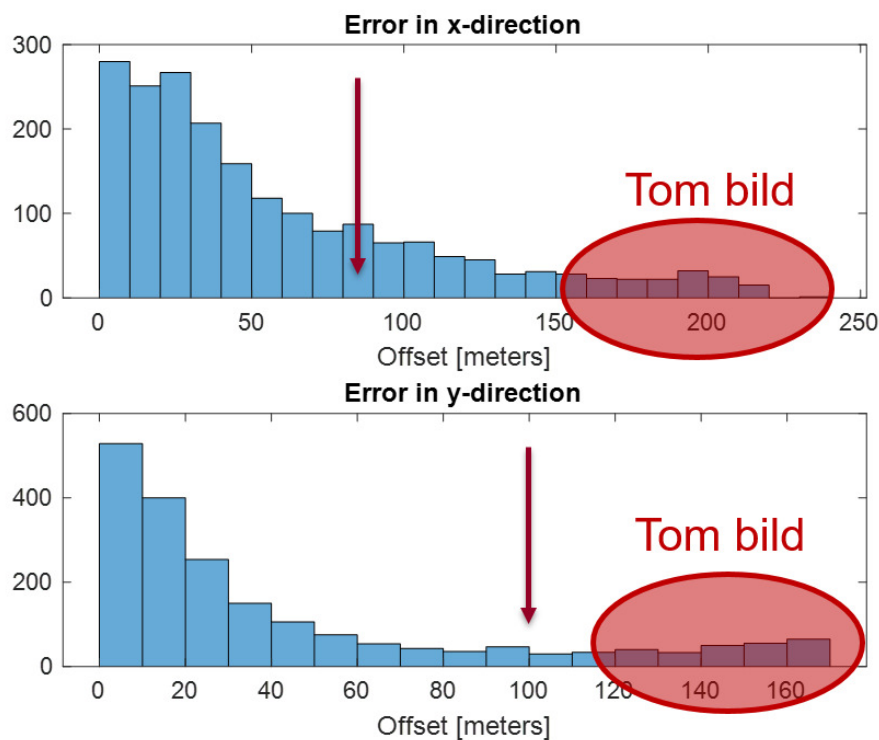
4.4 Resultat

Den generade uppsättningen träningsdata innehöll 2 000 valideringsfall. Ett exempel visas i Figur 21. Uppe till vänster visas plymen och sensorerna. När värdena från sensorerna använts i den iterativa processen för att fylla ut rasterbilden erhålls bilden uppe till höger. Denna bild utgör också indata till det neurala nätverket. Nere till vänster syns målbilden, alltså en Gaussisk fördelning över källans position. Nere till höger visas nätverkets resulterande utbild. Markerat med en vit punkt är källans egentliga position. I detta fall lyckades systemet att hitta källan riktigt bra.



Figur 21 Ett exempel som visar hur data ser ut i olika steg i processen. Ursprunglig koncentrationsplym visas uppe till vänster. Denna information är aldrig känd av systemet. De röda cirklarna visar på de utslumpade sensorerna som tillhandahåller den enda kända informationen om koncentrationsvärdena. Efter utfyllnadsprocessen erhålls panelen uppe till höger vilket fungerar som indata till systemet. Målbilden (facit) visas nere till vänster, det är denna bild systemet är tränat att hitta. Systemets resultat för detta valideringsfall visas nere till höger. Koncentrationsvärdena är plottade i logaritmisk skala.

För att kvantifiera resultaten beräknades en distinkt källposition ut från systemets utbild över sannolikhetsfördelningen av källpositionen. Källpositionen beräknades som väntevärdet på sannolikhetsfördelningen. Denna position jämfördes i varje valideringsfall med den kända källpositionen. Felet i både x- och y-riktningen erhöles då för varje enskilt fall. Den statistiska fördelningen av dessa fel visas i Figur 22. Förväntat resultat om källpositionen istället slumpas fram inom tillåtet område är 85 m i x-led och 100 m i y-led. Dessa värden är markerade med röda pilar i figuren. Det är tydligt att systemet klarar av att förutspå källans position avsevärt bättre än rent stokastiska gissningar. Vi ser även att källans position uppskattas bättre i y-led än i x-led. Detta är en följd av att vindriktningen främst ligger i x-riktningen och det är vanligt även när andra inversa metoder används att källans position i vindriktningen är svårare att hitta än i den ortogonala riktningen. Notera att i vissa fall blev indatabilden helt tom. Detta händer då källans position ligger väldigt högt eller lågt i kombination med att ingen sensor hamnade i motsvarande läge. Det är framförallt i dessa fall som systemet inte klarar av att finna källans position alls och är markerad med "Tom bild" i Figur 22. Dessa fall utgör besvärande situationer för systemet då inbilden är helt blank.



Figur 22 Histogram över felen i positionsuppskattningen i de båda riktningarna. Pilarna visar var felen i snitt skulle befinna sig om källpositionen skulle slumpas fram inom det tillåtna vita området som visas i Figur 19. Områdena med största felen är märkta med "Tom bild" och domineras av fall där inga sensorer var placerade inom plymen. Bilderna som matas in i systemet är då helt tomma och felet i källans position kan bli stort.

4.5 Diskussion

Det finns ett stort intresse att lyckas beräkna källpositioner utifrån sensordata snabbt, korrekt och robust. Detta är ett svårt problem som typiskt angrips med matematisk sannolikhetslära. I denna studie har en annan metod testats. Problemet har ställts upp som ett bildbehandlingsproblem och en vältestad maskininlärningsprocess, i form av ett falttningsnätverk, har applicerats. Studien var mycket begränsad i omfattning vilket innebar att många förenklingar applicerades. Resultaten visade att detta angreppssätt fungerar på så sätt att systemet klarar av att hitta källans position betydligt bättre än om en position slumpas fram inom tillåtet område. Huruvida precisionen kan förbättras

genom mer träning, förändrat faltningsnätverk eller andra metoder att behandla okända pixelvärden är oklart. Variationer i källstyrka och de meteorologiska förhållandena vore också önskvärda att inkludera. Det fanns tyvärr inte utrymme att reda ut de frågorna inom ramen för detta mini-projekt. Det bör också noteras att maskininlärning kan användas på denna typ av frågeställningar utan att formulera problemet som en fråga om bildtolkning.

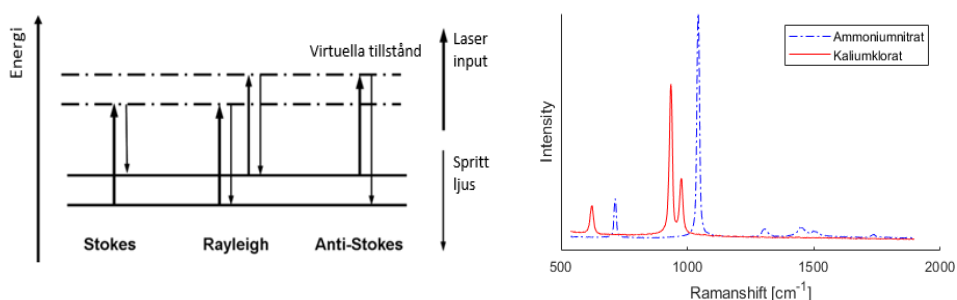
5 Deep Learning för avkodning av hyperspektrala Raman-bilder insamlade med Compressed Sensing

5.1 Bakgrund

Improviserade bomber eller IED:er (Improvised Explosive Device) är ett vanligt och ökande hot i samhället. För att i fält säkert kunna hitta dessa är det önskvärt att kunna ha en detektionsutrustning som inte bara kan hitta och identifiera IED:erna utan även klarar av detta på ett säkert avstånd. Laserbaserad avståndsdetektion t.ex. med Ramanspektroskopi är en teknik som har potential att klara detta. Genom att belysa explosivämnet eller explosivämnesspår med laserljus och samla in det spridda ljuset kan dessa upptäckas och identifieras. Idag har teknikutvecklingen kommit så långt att många ämnen kan detekteras i mycket små mängder. Det finns dock ett stort behov av att korta ner mättiderna för upptäckt och identifiering men även minska känsligheten för eventuellt störande bakgrund. För att åstadkomma detta har FOI använt spåringsdetektorer för explosivämnesspår, baserad på multispektral avbildande Ramanspektroskopi [46], och även hyperspektral avbildande Ramanspektroskopi [47] baserad på CS-teknik (Compressive Sensing)[48]. Även om mättiderna är i sekundnivå så är dagens CS-algoritmer långsamma med beräkningstider på flera minuter. Detta gör att även om det är mätteknisk möjligt att mäta i real-tid och täcka större ytor så är det inte praktiskt möjligt. Detta delprojekt har undersökt möjligheten att utnyttja maskininlärning och speciellt CNN (Convolutional Neural Network) för att väsentligt snabba upp avkodningen av mätdata.

5.2 Ramanspektroskopi

Även om teorin för Ramanspridning [49] är komplex så går principen för det att beskrivas genom en förenklad modell. När ljus infaller på en molekyl kan det spridas genom både elastiskt och oelastiskt spridning. Den elastiska spridningen är energibevarande tex. Rayleighspridning, detta betyder att det spridda ljuset har samma våglängd som den inkommande och innehåller således liten eller ingen molekylspecifik information. Den oelastiska spridningen där energin på fotonerna (dess våglängd) ändras är däremot av större intresse. Ramanspridning är en oelastiskspridningsprocess där en liten del av de inkommande fotonerna får energi (Anti-Stokes) eller tappar energi (Stokes) till molekylerna de interagerar med och därmed ändras våglängden på det spridda ljuset, på detta sätt kan vibrationsnivåerna i molekylerna undersökas vilket tillsammans ger ett molekylspecifikt fingeravtryck av molekylerna som undersöks.

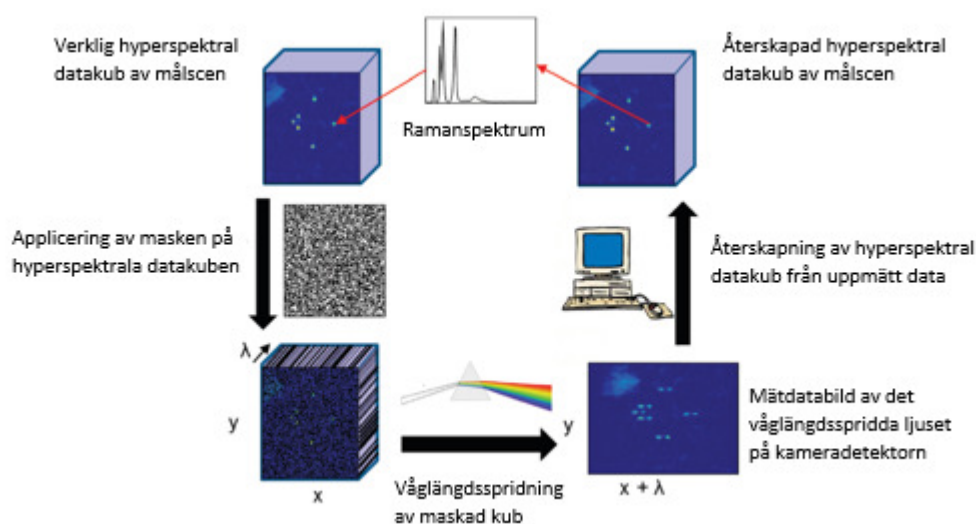


Figur 23 Vänster: Ramanspridning i en molekyl. Höger: Exempel på Ramanspektra.

I den vänstra delen av Figur 23 ses dessa tre ovan nämnda spridningsprocesser. Den högra bilden visar exempel på Ramanspektra från ammoniumnitrat och kaliumklorat, som ses har ammoniumnitrat två distinkta Ramantoppar i sitt spektrum medan kaliumklorat har tre.

5.3 Coded Aperture Snapshot Spectral Imaging - CASSI

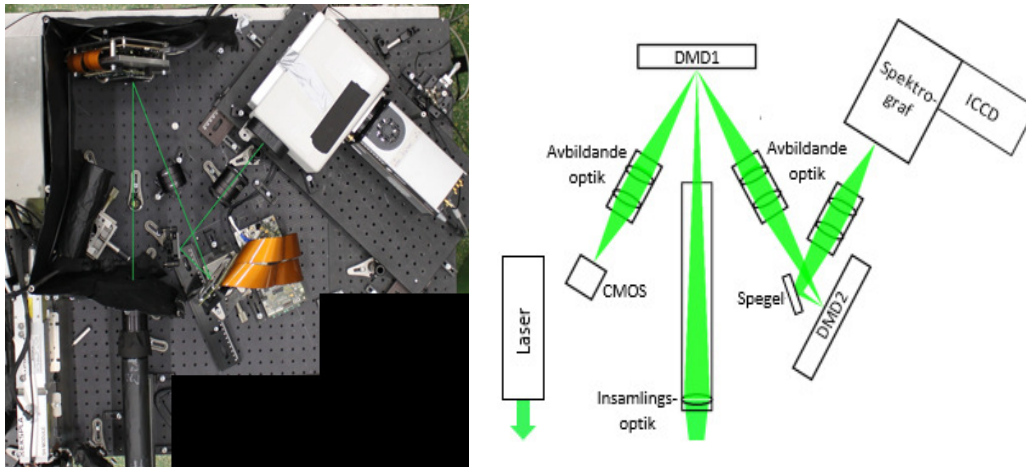
Att mäta en 3D datakub (två rumsliga, en spektral dimension) med hyperspektraldata när tillgängliga detektorer är som mest 2D är en komplicerad uppgift. Datainsamlingen blir därför en kompromiss mellan spektral upplösning, rumslig upplösning, ljusinsamling, och mättid. Genom att använda sig av mättekniker såsom CASSI (Coded Aperture Snapshot Spectral Imaging) [46, 50, 51], är det möjligt att mixa rumslig och spektral data vid mättillfället, och sedan från dessa mixade data, återskapa hela den ursprungliga hyperspektrala datakuben genom CS-algoritmer såsom TwIST (Two-Step Iterative Shrinkage/Thresholding) [52] eller GPSR (Gradient Projection for Sparse Reconstruction) [53].



Figur 24 Principiell bild av CASSI-tekniken.

CASSI-tekniken ses principiellt beskriven i Figur 24. Den verkliga hyperspektrala datakuben av målsценen (vänster upp) avbildas på en mask. Masken består av ett binärt mönster som antingen släpper igenom eller blockerar ljuset som faller in på den. De delar av bilden som passerar masken våglängds-uppdelas (tex av en spektrometer), som i sin tur kan avläsas av en 2D kameradetektor. Bilden som avläses på kameradetektorn består av överlagrad rums- och spektral information (höger nere), och kan sedan avkodas med en CS-algoritm för att återskapa den hyperspektrala datakuben av målsценen. För att ytterligare förbättra återskapningen kan man mäta flera bilder efter varandra med olika masker vid varje mätning[54], det både förbättrar kvaliteten på återskapningen men även hastigheten för CS-algoritmen att konvergera.

5.4 Experiment



Figur 25 Vänster: Fotografi över experimentupställningen. Höger: Schematisk bild av CASSI-uppställningen.

Ett foto av experimentuppställningen kan ses till vänster i Figur 25, och i den högra delen ses en schematisk bild med huvudkomponenterna utmärkta. En pulsad laser belyser provytan varvid det spridda ljuset samlas in genom insamlingsoptiken och avbildas på DMD1. Insamlingslinsen är 2" i diameter och direkt framför denna sitter också ett laserfilter som blockerar det direktspridda laserljuset. En DMD (Digital Mirror Device) består av många små speglar som kan ställas i två olika vinklar, där ena läget riktar ljuset mot en CMOS-kamera så en vanlig bild över provytan kan ses, och där andra läget riktar ljuset mot en andra DMD (DMD2). Denna andra DMD (DMD2) används för att kompensera för det sneda bildplan som DMD1 skapar.



Figur 26 Exempel på binär mask som visas på DMD1 med 30x30 element i upplösning.

På DMD1 skapas den kodade masken, se exempel i Figur 26. Efter DMD2 avbildas det insamlade ljuset på ingångsplanet av en avbildande spektrograf. I spektrografen våglängduppdelas ljuset för att sedan mätas med en ljusförstärkt CCD-kamera. På detta sätt får man en bild av målet för varje enskild våglängd överlagrat på detektorn, och tack vare den kodade masken på DMD1 går det att separera och återskapa fullständig bild och spektral information genom CS-algoritmer.

5.5 Maskininlärning och Deep Learning

Maskininlärning (ML) och Deep Learning (DL) används huvudsakligen för att extrahera information från sensordata där detektion och klassificering av bildinnehåll är frekvent förekommande tillämpning[55]. På senare tid har DL även tillämpats på inverterade problem där man försöker återskapa ursprungsdata utifrån en uppsättning transformerade data. Två exempel på ett inverterat problem är avbrusning där man försöker återskapa brusfria bilder från en brusig bild och superupplösning där man vill återskapa en högupplöst bild utifrån en lågupplöst nedsamplad bild. Inversa problem är vanligen illaställda och kan inte lösas utan att introducera något (regulariserande) bivillkor. Compressed Sensing (CS) är ett inverterat problem som är illaställt där man utifrån ett fåtal glesa/komprimerade mätningar vill rekonstruera en högupplösning bild eller Ramanspektrum.

Inom maskininlärning är DL ett neuralt nätverk med flera dolda lager. Inom bildanalys är DL ett faltande neuralt nätverk (eng. Convolutional Neural Net (CNN)) med flera dolda lager. Ett faltande neuralt nätverk är ett neuralt nätverk med en specifikstruktur där faltande lager kombineras med vanliga så kallade fullt kopplade lager (engelska fully connected layer). Ett faltande nätverk har vanligen följande struktur: ett faltande lager följs av ett nedsamlingslager (eng. pooling layer) och där efter ett aktiveringslager. Ett CNN tränas genom att justera nätverkets vikter och då primärt vikterna i de olika faltningskärnorna. Ett CNN kan användas för detektion och klassificering men också för att generera högupplösliga bilder från en ett fåtal parametrar. Det är denna egenskap som är central för användning inom CS där man vill återskapa högupplösliga data från ett fåtal mätningar[37, 56, 57]. Denna typ av ansatser brukar benämnas avkodnings-CNN (eng. decoding CNN) eftersom man vill återskapa data från en komprimerad representation. Målsättningen är att återskapa data genom att avkoda den komprimerade representationen.

CS inom Raman CASSI kan ses som ett regressionsproblem där man vill återskapa Raman spektrum utifrån ett antal uppsättningar CS mätningar. Presentationen nedan är förenklad då man inte återskapar enskilda spektrum från CS-mätningar utan man istället rekonstruerar en vektor med spektrum simultant. Låt x vara ett Raman spektrum, y vara en (simulerad) CS mätning från ett CASSI-Raman systemet och låt f vara överföringsfunktionen från Raman spektrum till CS-mätning:

$$y = f(x)$$

Givet att överföringsfunktionen (diskuteras i detalj i avsnittet Datagenerering) är känd så kan man beräkna CS-mätningen y för ett givet Raman spektrum x . Målet för DL är att bestämma ett CNN som kan återskapa x utifrån CS-mätningarna y , man vill bestämma en funktion

$$\hat{x} = c(y; w)$$

där c är ett CNN som avkodar CS-representationen och rekonstruerar ett Ramanspektrum $\hat{x}$.

$c(y; w)$ är ett CNN som rekonstruerar x utifrån CS-mätningen y dvs. en avbildning från mätningrummet till datarummet och w utgör parametrarna i nätverket. Rekonstruktionsfelet för $c(y; w)$ definieras som

$$\|x - \hat{x}\|_1.$$

Rekonstruktionsfelet uttrycker hur väl $c(y; w)$ nätverket återskapar mål-Ramanspektrumet x från CS-mätningarna givet en uppsättning parametrar w . L_1 -normen används vanligen men även andra normer är frekvent förekommande. Givet överföringsfunktionen f och rekonstruktionen $\hat{x}$ kan överföringsfunktionen appliceras på det rekonstruerade Ramanspektrumet

$$\hat{y} = f(\hat{x}).$$

Där $\hat{y}$ är en CS-mätning för det rekonstruerade Ramanspektrumet $\hat{x}$. Då y är känt sedan tidigare vill vi även minimera

$$\|y - \hat{y}\|_1$$

som brukar kallas återprojektionsfelet. Målsättningen är att bestämma ett CNN $c(y; w)$ som minimerar rekonstruktionsfelet och återprojektionsfelet. Målfunktionen, som också benämns förlustfunktion eller loss-funktion, som skall minimeras med avseende på parametrarna w i nätverket blir

$$E(y) = \|x - \hat{x}\|_1 + \lambda \|y - \hat{y}\|_1$$

där λ är en parameter som möjliggör viktning mellan rekonstruktionsfelet och återprojektionsfelet. $\hat{x}$ och $\hat{y}$ beror på CNN nätet $c(y; w)$ och man vill bestämma parametrarna w för $c(y; w)$ så att målfunktionen minimeras över en träningsmängd. Parametrarna w i $c(y; w)$ optimeras med gradientsökning med adaptiv steglängd, där i detta fall optimeringsmetoden Adam [58] har använts.

Givet en fix struktur för nätverket $c(y; w)$ och en träningsmängd så kan en optimal (lokalt minimum) uppsättning vikter w bestämmas med gradientbaserad optimering enligt beskrivningen ovan.

$c(y; w)$ är dock en mycket komplex struktur som måste bestämmas manuellt och dessa val kommer att vara helt avgörande för rekonstruktionsresultaten. Några centrala parametrar som måste bestämmas och som kommer få mycket stor påverkan på rekonstruktionsresultatet är [59]:

- Transfer learning – är det möjligt att utnyttja förtränade nätverk och strukturer för att uppnå goda resultat.
- Antal lager, typ av lager och antal noder i varje lager.
- Antal faltande lager, storlek på faltningsskärnorna och nedsamlingsmetod.
- Kopplingsstruktur mellan lager i nätverket – gleskopplat, fullkopplat eller "skipping"-kopplingar.
- Aktiveringsfunktion i olika de olika lagerna.

I vissa fall finns väletablerade riktlinjer för hur man bör bestämma nätverksarkitektur, detta gäller dock primärt för detektions- och klassificeringsproblem. För inverterade problem och speciellt CS-problem är tidigare erfarenheter mer begränsat och man måste istället experimentellt utvärdera olika parametrar för att hitta en lämplig konfiguration. Explicit diskussion av nätverkskonfiguration diskuteras i avsnitt Nätverksarkitektur.

5.6 Metodbeskrivning

5.6.1 Datagenerering

Inom DL är ett av de största problemen att införskaffa eller simulera stora mängder av träningsdata. Den data som används i träningen av ett DL-nätverk ska helst vara införskaffade med den hårdvara som nätverket ska tränas för. Problemet med "verklig" data är mängden som behöver samlas in, variationen i den insamlade data samt annotering av data, är tidskrävande och kostsamt. Alternativet är att simulera data, detta öppnar upp möjligheten att generera mycket data, med stor variation och som är annoterad. Nackdelarna med simulerade data är att det är svårt att modellera ett existerande system inklusive brus och distorsioner på ett tillfredställande sätt.

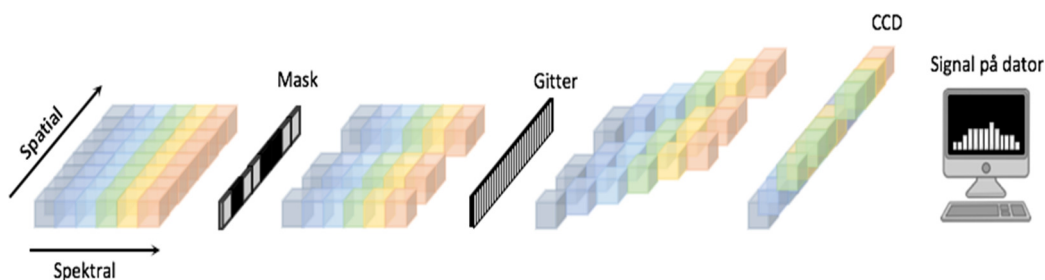
Fördelarna med att använda en förenklad simulering i detta tidiga skede av forskningen överväger mot att använda insamlade data och även en komplex simulerad modell av det fysiska systemet. Det är en stor utmaning hur man ska applicera DL-modeller tränade på simulerade träningsdata på data från vårt utvecklade CASSI-systemet.

Därför valdes som första steg i forskningen att generera en förenklad modell av systemet där den mer teoretiska inverteringen (rekonstruktionen) står i fokus.

För att skapa träningsdata genererades en förenklad modell av CASSI-systemet. De förenklingar som gjordes var:

- Ingen "cross-talk" mellan närliggande rader, vilket gör att en rad i taget går att rekonstruera. Detta går att realisera i verkligheten genom att stänga varannan rad i masken, detta leder till att dubbelt så många mätningar måste göras. Fördelarna med att separera varje rad för sig är att det minskar storleken på det neurala nätverket och även minskar tiden att träna nätverket.
- Konstant dispersion dvs. avbildningen för en rad är konstant oavsett våglängd.
- Ingen optisk aberration.
- Spatial(rumslig) upplösning 64 x 64 punkter.
- Spektral upplösning 340 våglängder.
- Fyra stycken olika masker dvs fyra mätningar per scen.

Detta resulterar i en CASSI-modell som för en rad för i den hyperspektrala kuben kan illustreras enligt Figur 27.



Figur 27 Bild som illustrerar ljusets väg genom en rad i CASSI systemet

I Figur 27, ses hur ljuset, för olika våglängder, passerar genom det simulerade CASSI systemet. Den hyperspektrala kuben för en specifik rad (längst till vänster) träffar masken varvid delar av ljuset blockeras. Därefter möter den delvis blockerade kuben ett gitter och förskjuts en pixel för varje våglängd. Detta ljus träffar detektor (CCD) som spatialt och spektralt mixade data. Eftersom detektorn är monokrom mäter den styrkan på ljuset oberoende av våglängd och den detekterade signalen blir likt bilden på datorn längst till höger. Detta representerar mätdata som senare används för att återskapa den hyperspektrala datakuben för denna specifika rad.

Genereringen av träningsdata var styrd av en algoritm som på slump lade ut Ramanspektra på en 64x64 bild. De spatiala substanserna hade formen av rektanglar eller ellipser som roterades slumpartat. Verktöget skapade både en hyperspektral bild och en mätning av hyperspektrala bilden. Med hjälp av verktöget kunde ett antal parametrar styras för att få önskade träningsdata:

- Antal substanser i spatiala bilden.
- Storleken på substanserna.
- Vilka substanser som används.

- Magnitud på Ramanspektrat
- Magnitud additivt bakgrundsbrus på hyperspektrala referens-bilden.
- Magnitud additivt brus på mätningen.

5.6.2 Nätverksarkitektur

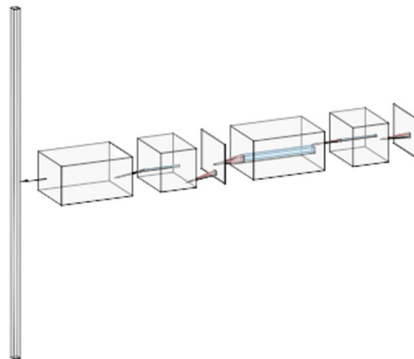
Eftersom DL inte har applicerats på CASSI-Raman till författarnas kännedom söktes DL referens-punkter inom ett liknande men annat underområde, nämligen Compressive Imaging (CI). Inom CI har både rena FC- och CNN- nätverk [37, 60, 61] och generativa modeller [56, 60] använts framgångsrikt för att invertera en-pixel-kameramätningar. Gemensamt för alla dessa nätverk är att som första steg ta in mätningarna Y i ett FC-lager och transformera antalet noder till en önskvärd mängd som sedan går in till ett CNN med flera lager som ger ut den rekonstruerade bilden. Från dessa nätverk som inspiration skapades tre typer av nätverk.

Indata till alla nätverks konfigurationer är en

$$n_0 = \text{mätningar} \times (\text{spatial} + \text{spektral upplösning} - 1) = 4 \times (64 + 340 - 1) = 1612$$

lång 1D vektor där spatiala och spektrala data har modulerats ihop. Utdata från alla nätverkets konfigurationer är $\text{spatial upplösning} \times \text{spektral upplösning} = 64 \times 340$ element stor 2D matris där kolumnerna representerar den spatiala positionen och raderna representerar respektive spatiala kolumnens spektrala data. Även om utdata från nätverket är en 2D matris bör den betraktas som en $n_1 = 64 \times 340 = 21760$ lång 1D vektor där varje spatial position innehåller en 340 dimensionell spektral signatur för positionen. Utdata från nätverken är således en rekonstruerad rad där varje position i vektorn innehåller en spektral signatur.

Nätverksarkitektur 1 - FC(1 lager)→CNN(5 lager)



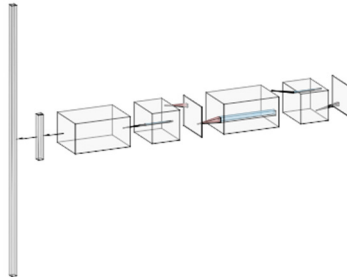
Figur 28 Arkitekturen för CNN-nätverk 1.

Nätverksarkitektur 1 är samma nätverk som [62], som anpassats till datan. Då indata är en endimensionell vektor innehållande mätvärden, är det naturligt med ett första lager som är fullkopplat (FC). Ett potentiellt problem är att FC lagret blir relativt stort (35M parametrar) i förhållande till CNN-delen (~1M parametrar). Allt för många parametrar i ett nätverk ökar risken för överträning och nätverket tar längre tid att träna. Vidare kommer FC-delen att dominera nätverket i förhållande till CNN-delen av nätverket.

I Figur 28 ses en visualisering av arkitektur 1. Indatat till modellen är n_0 datapunkter. Första lagret ett fullkopplat lager som har n_0 noder som input och n_1 noder output. Outputen från de fullkopplade nätet går in i efterföljande CNN. Alla lager har batch-normalisering och ReLu-aktivering efter lagret förutom det sista lagret. I de första och fjärde faltningsslagen är filterkärnans storlek 5×5 och filterdjup på 128, de andra och femte lagren är filterkärnans

storlek 1x1 och filterdjup på 64 och tredje och sjätte lagret är filterkärnans storlek 3x3 och endast en filterkärna. Detta resulterar i ~35M parametrar.

Nätverksarkitektur 2 - FC(2 lager)→CNN(5 lager)

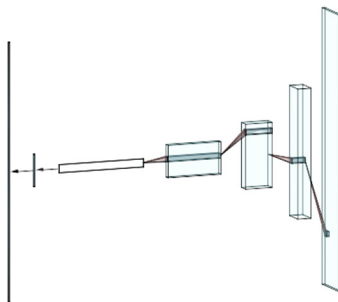


Figur 29 Arkitekturen för CNN-nätverk 2.

Hypotetiskt kan indata som är $\frac{n_0}{n_1} \approx 7\%$ undersamplat komprimeras ytterligare. Det finns två motiveringar till det, dels rent teoretiskt även om det finns Ramanspektra på hela raden visar sig Ramaninformationen endast i en fraktion av de 340 banden som vi vill rekonstruera, vidare överlagrar mätdata i sig (4 mätningar) information från varandra. Det andra argumentet som är mer empiriskt visar att generativa nätverk som GAN [56, 60] kan rekonstruera bilder med mer information och storlek i paritet med n_1 , från en 1D vektor som input till nätverket med en storlek på 100 variabler.

För att minska antalet parametrar i FC-delen i arkitektur 1 läggs därför ett FC-lager till, med ändringar att det första FC samplar ner input n_0 till ~200 noder och det andra FC lagret samplar upp till önskat antal noder i CNN-delen. Denna metod reducerar parametrarna i FC-delen från ca 35M till ca 4-10M beroende på antal noder i nedsamlingssteget och storlek på input i CNN-delen. De efterföljande CNN-lagren är identiska med arkitektur 1. I Figur 29 ses en visualisering av arkitektur 2, där skillnaden mot arkitektur 1 är ett extra FC-lager som fungerar som nedsampling.

Nätverksarkitektur 3 - FC(2 lager)→CTNN(4 lager)



Figur 30 Arkitekturen för CNN-nätverk 3.

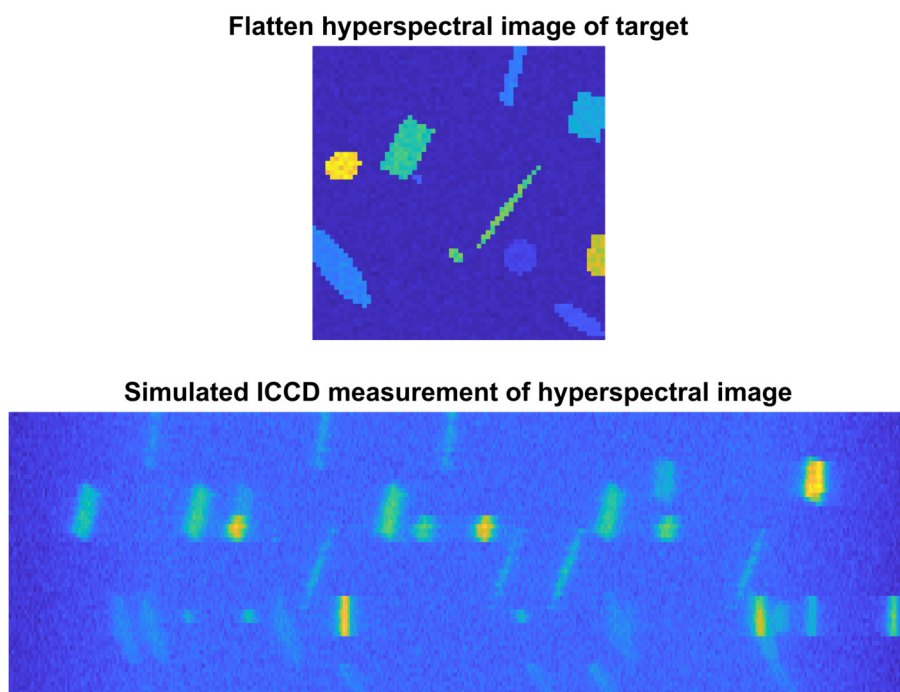
Den sista nätverks-arkitekturen som skapades hämtade inspiration från [56, 60] som framgångsrikt använde sig av [63] för att rekonstruera bilder. Avkodaren i [64] rekonstruerar bilden från en 1D vektor använder 2D-avfaltung för att sampla upp en 2D bild till önskad storlek. Denna typ av avkodare passar in på hur indata ser ut och även hur utdata kan tolkas.

Nätverksarkitekturen på det tredje nätet har samma typ av fullkopplade nätverk som nätverk 2 men outputen från det sista fullkopplade nätet är $256 \times 21 \times 4 = 21504$ noder

som går in i efterföljande avfaltande nät där noderna är transformerade till storleken (21,4,256). Alla lager har en filterkärna på 3x3 och stride på (2,2) samt batch-normalisering och ReLu aktivering efter alla lagren förutom det sista lagret. Med dessa inställningar dubblas den spatiala upplösningen mellan varje avfaltande lager, för varje nytt lager halveras antalet filterkärnor förutom i sista avfaltande lagret som har en filterkärna. Mellan avfaltningsslager tre och fyra läggs en rad till (eng. zero pad) för att output storleken ska stämma. I Figur 30 ses en visualisering av nätverk 3 som har ~7M parametrar.

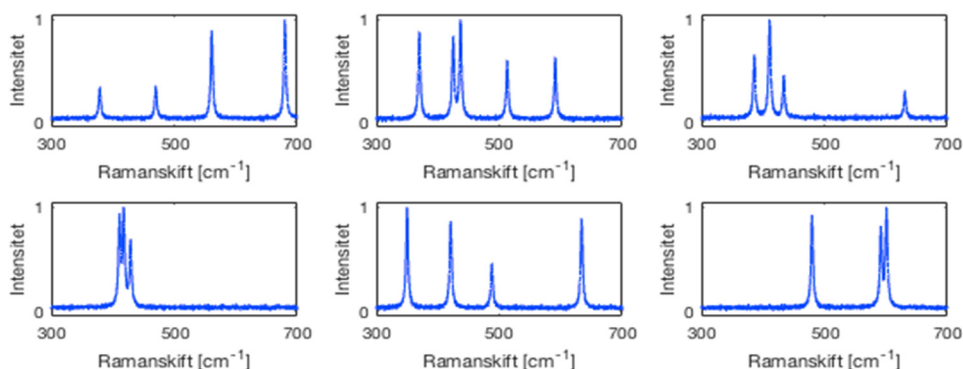
5.7 Resultat

Vid generering av målytan slumpades ett antal rektanglar och ellipser ut som mål och placerades på målytan. Se Figur 31 för exempel på en målyta med de olika målen utplacerade och simulerad mätning.



Figur 31 Exempel på mål placerade på målytan (övre) och simulerad ICCD-mätning av målytan (undre).

Dessa olika mål tilldelades sedan olika Ramanspektrum. Ramanspektra slumpades fram genom att låta ett spektrum bestå av 3-6 Ramantoppar med Ramanskift $300-700\text{ cm}^{-1}$ och därefter lades ett vitt bakgrundsbrus på. Senare vid simulerade mätningen adderades även ett vitt mätbrus. I Figur 32 ses exempel på Ramanspektra som genererades med bakgrundsbrus adderat.



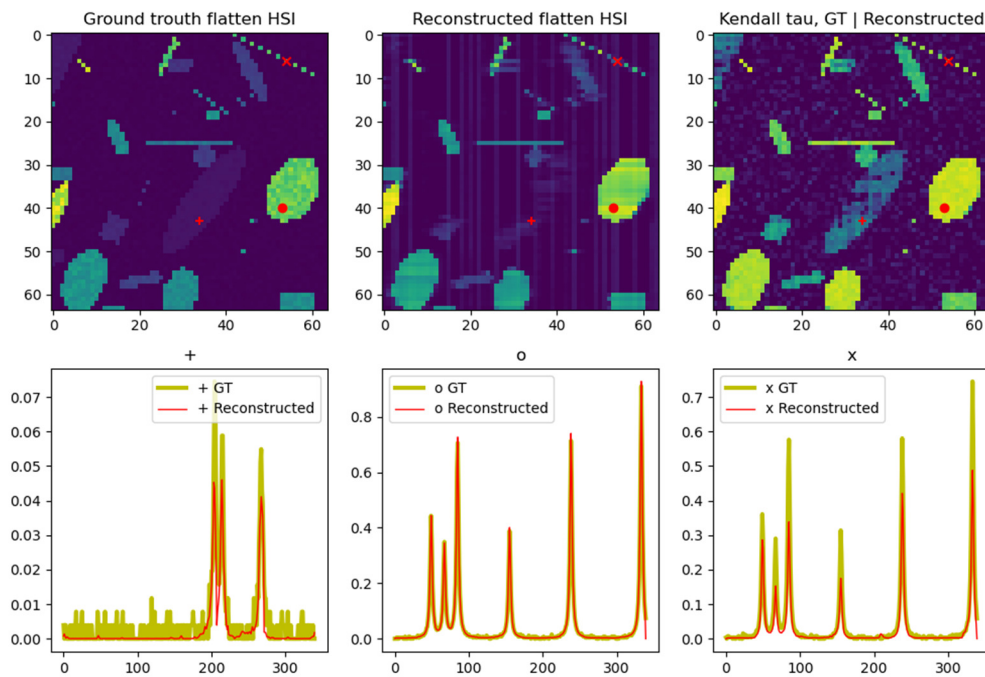
Figur 32 Exempel på Ramanspektra med pålagt brus.

5.8 Träning och rekonstruering av få Ramanspektra

I första testet användes endast 16 olika Ramanspektra i både test- och valideringsdataseten. Bakgrundsbruset i hyperspektrala bilden låg på 0.5% och 1% i simulerade mätningen. Alla tre nätverksarkitekturer konvergerade snabbt och nätverksarkitektur (3) visade minst fel i valideringen. Vid visualisering av de rekonstruerade hyperspektrala bilderna kunde man se att arkitektur (1) och (2) hade en kraftig DC-offset i Ramanspektra som var konstant över den spatiala domänen. De spektrum med högre intensitet i hyperspektrala bilden än DC-offseten i den rekonstruerade bilden kunde topparna rekonstrueras korrekt, medans spektra med lägre magnitud än DC-offseten dränktes. Arkitektur (3) visade på samma tendenser men med en mycket lägre DC-nivå och spektra med två gånger högre magnitud än bruset kunde rekonstrueras i sin helhet, därav är följande resultat givet arkitektur (3).

I valideringssetet rekonstruerades alla Ramanspektra korrekt, med relativt korrekt magnitud och på rätt plats spatialt förutom när SNR blev för lågt. Modellen kunde rekonstruera Ramanspektra som endast syntes i en pixel spatialt, men rekonstruktionen var lite utsmetad till grannpixlarna. Modellen kunde även reducera det additiva bruset som lagts till. Magnitudbild, Kendall Tau och enskilda spektra ses i Figur 33.

När modellen såg ett Ramanspektra som inte hade varit med i testdatat kunde två rekonstruktionsfel inträffa, om en topp i det okända spektra delade topp med något av de kända spektra rekonstruerades den delen av spektra men inget mer, alltså en partiell rekonstruering av del delen av spektra modellen kände till. Det andra rekonstruktionsfelet som kunde inträffa var att en topp i det okända spektra delade ungefärligt en eller flera toppar med ett känt spektra, då rekonstruerades hela substansen som det närmast kända spektra, alltså en felklassificering.



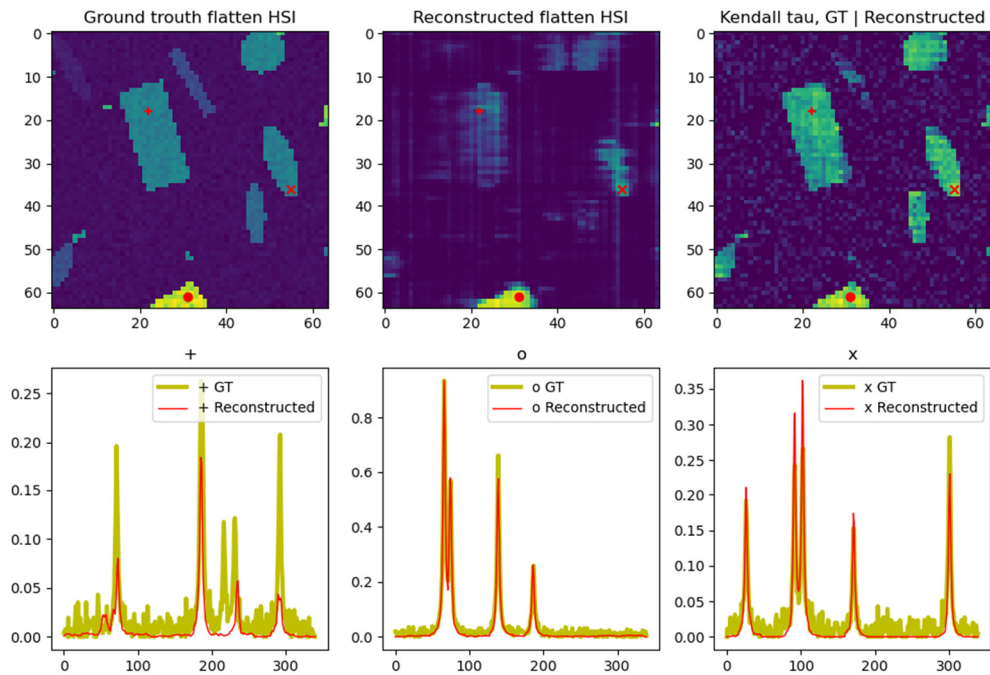
Figur 33 Övre raden: t.v. Referensbild och center Rekonstruerad bild (magnituden av Ramanspektra), t.h. Kendall Tau mellan referens och rekonstruerade kuberna. Undre: Spektra från enskilda spatiala pixlar, referens i gult och rekonstruerad i rött.

5.9 Träning och rekonstruering med brett antal Ramanspektra

Andra testet bestod av 16 olika förstahands Ramanspektra (FR), där fyra av dessa användes i varje bild, 128 olika andrahands Ramanspektra (AR) där 4 av dem fanns i varje bild och 8 helt slumpade Ramanspektra (SR) i varje bild (Notera möjligt att komma med i en bild). Detta innebär att i valideringssetet var hälften av spektra inte med i träningsseten (nätverket har aldrig sett de Ramanspektra). Både bakgrundsbruset och mätbruset ökades till 1% respektive 3.5% vilket anses som en mer realistisk och svårare brusmodell för nätverket att träna på. Denna uppsättning var gjord dels för att göra en mer robust och generell rekonstruktion men också för att se vad som skulle hända när nätverket rekonstruerar ett helt okänt spektrum men ändå behålla fokus på "viktiga spektra" (FR). Notera även att många av spektra i testdatat med hög sannolikhet endast kommer visa sig för modellen en eller ett fåtal gånger.

Detta test visade alla arkitekturerna samma resultat för DC-nivåerna som i första testet, därav presenteras resultatet för arkitektur (3). Likt första testet kunde denna modell rekonstruera FR givet en adekvat SNR. AR rekonstruerades antingen till fullo eller partiellt, medan SR med högre sannolikhet rekonstruerades endast partiellt eller ingenting alls. I en mindre del av valideringen i spektra från SR med låg SNR syntes i rekonstruktionen i vissa fall toppar eller par av toppar som inte stämde överens med referensdatat, till skillnad från första testet låg dessa felaktiga toppar relativt nära referenstoppen och det är oklart hur stor del de relativt kraftiga bruserna kan generera "spöktopp" i rekonstruerade spektra.

I Figur 34 där en bild från valideringssetet har rekonstruerats innehåller endast två substanser kända spektra från FR och AR (noterat med "o" och "x") och resten från SR (fanns inte med i träningen). I den rekonstruerade bildens magnitud syns det att SR-substanser blir undertryckta, medan Kendall Tau visar att det finns korrelation.



Figur 34 Övre raden: t.v. Referensbild och center Rekonstruerad bild (magnituden av Ramanspektra), t.h. Kendall Tau mellan referens och rekonstruerade kuben. Undre: Spektra från enskilda spatiala pixlar, referens i gult och rekonstruerad i rött.

5.10 Slutsatser och diskussion

Inom projektet har vi visat på att det är fullt möjligt att använda sig av DL som rekonstruktionsalgorithm för CASSI Raman. Genom att använda sig av DL är största vinsten att det möjliggör rekonstruktion i realtid – rekonstruktionstiden för hyperspektralkub går från minuter till millisekunder. I rapporten har vi visat att ett relativt litet och därav snabbt tränat CNN återskapar Ramanspektra med god spatial och spektral noggrannhet och precision. Genom att bredda antalet spektra nätverket ser i träningen har vi visat att nätverket klarar av en mer generell rekonstruktion.

Referenser

- [1] D. Gustafsson, T. Gustavi, L. Ahlinder, D. Bergström, O. Björnham, C. Brännlund, P. Brännström, N. Granström, J. Ingemarsdotter, M. Nordberg, L. Persson, P. Svenmarck, och C. Svensson, "Försvarsorienterade tillämpningar av maskininlärning - Nya ansatser och tillämpningar", FOI-D--0945--SE, 2020.
- [2] W. Taylor, E. Camilleri, D. L. Craft, G. Korza, M. R. Granados, J. Peterson, R. Szczepaniak, S. K. Weller, R. Moeller, T. Douki, W. W. K. Mok, och P. Setlow, "DNA Damage Kills Bacterial Spores and Cells Exposed to 222-Nanometer UV Radiation", *Appl Environ Microbiol*, Vol. 86, Nr. 8, Apr 1 2020.
- [3] T. Broman, M. Forsman, S. Nord, och J. Thelaus, "A Matter of Life and Death. Molekylära metoder för att avgöra om bakterier, virus och protozoer lever eller är döda", FOI-D--0862--SE, 2019.
- [4] J. B. Emerson, R. I. Adams, C. M. B. Román, B. Brooks, D. A. Coil, K. Dahlhausen, H. H. Ganz, E. M. Hartmann, T. Hsu, N. B. Justice, I. G. Paulino-Lima, J. C. Luongo, D. S. Lympelopoulou, C. Gomez-Silvan, B. Rothschild-Mancinelli, M. Balk, C. Huttenhower, A. Nocker, P. Vaishampayan, och L. J. Rothschild, "Schrödinger's microbes: Tools for distinguishing the living from the dead in microbial ecosystems", *Microbiome*, Vol. 5, Nr. 1, sidor 86, 2017/08/16 2017.
- [5] L. Ahlinder, P. O. Andersson, och A.-L. Johansson, "Levande och döda: Jämförelse av raman- och SWIR-spektra från levande och avdödade bakterier och sporer", FOI-R--4364--SE, 2016.
- [6] M. Manley, "Near-infrared spectroscopy and hyperspectral imaging: non-destructive analysis of biological materials", *Chemical Society reviews*, Vol. 43, Nr. 24, sidor 8200-8214, 2014.
- [7] J. Dubois, E. Neil Lewis, F. S. Fry, och E. M. Calvey, "Bacterial identification by near-infrared chemical imaging of food-specific cards", *Food Microbiology*, Vol. 22, Nr. 6, sidor 577-583, 2005/12/01/ 2005.
- [8] J. S. Shenk, J. J. Workman, och M. O. Westerhaus, "Application of NIR spectroscopy to agricultural products. In 'Handbook of Near-infrared Analysis'.(E)", 1992.
- [9] R. Tkachuk, "Analysis of whole grains by near-infrared reflectance", *Infrared technology in the agricultural and food industries*, 1987.
- [10] A. Bellet, A. Habrard, och M. Sebban, "A survey on metric learning for feature vectors and structured data", *arXiv preprint arXiv:1306.6709*, 2013.
- [11] Webb-sida: *R: A language and environment for statistical computing ver. 3.3.3*, ([Online]. Available: <https://www.r-project.org/>).
- [12] C. Beleites och S. Valter, "hyperSpec: a package to handle hyperspectral data sets in R ver. 0.98-2016118," *Journal of statistical software*, 2016.
- [13] Webb-sida: *An Introduction to the prospectr package*.
- [14] W. N. Venables och B. D. Ripley, "Modern Applied Statistics with S", Fourth:e upplagan, Springer, 2002.
- [15] Y. Tang, G. Tao, och X. Nan, "dml: Distance metric learning in R version 1.1.0", 2015.
- [16] A. Bar-Hillel, T. Hertz, N. Shental, och D. Weinshall, "Learning a Mahalanobis Metric from Equivalence Constraints", *J. Mach. Learn. Res.*, Vol. 6, sidor 937-965, 2005.
- [17] L. Fan, F. Zhang, H. Fan, och C. Zhang, "Brief review of image denoising techniques", *Visual Computing for Industry, Biomedicine, and Art*, Vol. 2, Nr. 1, sidor 7, 2019/07/08 2019.
- [18] K. Zhang, W. Zuo, Y. Chen, D. Meng, och L. Zhang, "Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising", *IEEE transactions on image processing : a publication of the IEEE Signal Processing Society*, Vol. 26, Nr. 7, sidor 3142-3155, 2017.
- [19] K. Zhang, W. Zuo, och L. Zhang, "FFDNet: Toward a Fast and Flexible Solution for CNN-Based Image Denoising", *IEEE Transactions on Image Processing*, Vol. 27, Nr. 9, sidor 4608-4622, 2018.

- [20] J. Lehtinen, J. Munkberg, J. Hasselgren, S. Laine, T. Karras, M. Aittala, och T. Aila, "Noise2Noise: Learning Image Restoration without Clean Data", *Proceedings of the 35th International Conference on Machine Learning*, Vol. 80, sidor 2965--2974, 2018.
- [21] A. Krull, T. Buchholz, och F. Jug, "Noise2Void - Learning Denoising From Single Noisy Images", i *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2124-2132, 2019.
- [22] J. Batson och L. Royer, "Noise2self: Blind denoising by self-supervision", *arXiv preprint arXiv:1901.11365*, 2019.
- [23] A. Buades, B. Coll, och J.-M. Morel, "Non-local means denoising", *Image Processing On Line*, Vol. 1, sidor 208-212, 2011.
- [24] K. Dabov, A. Foi, V. Katkovnik, och K. Egiazarian, "Image denoising by sparse 3-D transform-domain collaborative filtering", *IEEE Transactions on image processing*, Vol. 16, Nr. 8, sidor 2080-2095, 2007.
- [25] F. Schönfeldt, "A Langevin Equation Dispersion Model for the Stably Stratified Planetary Boundary Layer", Swedish Defence Research Agency., 1997.
- [26] S. Sehlstedt, "A Langevin Equation Dispersion Model for the Unstably Stratified Boundary Layer", Swedish Defence Research Agency, 2000.
- [27] I. G. Enting, *"Inverse Problems in Atmospheric Constituent Transport"*, Cambridge University Press, 2002.
- [28] N. Brännström och L. Å. Persson, "A measure theoretic approach to linear inverse atmospheric dispersion problems", *Inverse Problems*, Vol. 31, Nr. 2, sidor 025009, 2015/01/30 2015.
- [29] E. Walter och L. Pronzato, *"Identification of Parametric Models from Experimental Data"*, Springer, 1997.
- [30] F. O'Sullivan, "A Statistical Perspective on Ill-Posed Inverse Problems", *Statist. Sci.*, Vol. 1, Nr. 4, sidor 502-518, 1986/11 1986.
- [31] A. M. Stuart, "Inverse problems: A Bayesian perspective", *Acta Numerica*, Vol. 19, sidor 451-559, 2010.
- [32] A. Doucet, N. d. Freitas, och N. Gordon, *"Sequential Monte Carlo Methods in Practice"*, 2001.
- [33] H. W. Engl, M. Hanke, och A. Neubauer, *"Regularization of Inverse Problems"*, Springer, 1996.
- [34] F. Bauer och M. A. Lukas, "Comparing parameter choice methods for regularization of ill-posed problems", *Mathematics and Computers in Simulation*, Vol. 81, Nr. 9, sidor 1795-1841, 2011/05/01/ 2011.
- [35] E. D. Vito, M. Fornasier, och V. Naumova, "A Machine Learning Approach to Optimal Tikhonov Regularisation I: Affine Manifolds", 2017.
- [36] S. Zilitinkevich, P.-E. Johansson, D. Mironov, och A. Baklanov, "A similarity-theory model for wind profile and resistance law in stably stratified planetary boundary layers", *J Wind Engineering and Industr Aerodynamics*, Vol. 74, sidor 209-218, 1998.
- [37] C. F. Higham, R. Murray-Smith, M. J. Padgett, och M. P. Edgar, "Deep learning for real-time single-pixel video", *Scientific Reports*, Vol. 8, Nr. 1, sidor 2369, 2018/02/05 2018.
- [38] N. Radwell, S. D. Johnson, M. P. Edgar, C. F. Higham, R. Murray-Smith, och M. J. Padgett, "Deep learning optimized single-pixel LiDAR", *Applied Physics Letters*, Vol. 115, Nr. 23, sidor 231101, 2019/12/02 2019.
- [39] H. Palangi, R. K. Ward, och L. Deng, "Distributed Compressive Sensing: A Deep Learning Approach", *IEEE Transactions on Signal Processing*, Vol. 64, sidor 4504-4518, 2016.
- [40] A. Mousavi, A. B. Patel, och R. G. Baraniuk, "A deep learning approach to structured signal recovery", i *2015 53rd Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, 1336-1343, 2015.
- [41] J. Ingemarsdotter, D. Eidenskog, och V. Hedtjärn Swaling, "Vilse i lasagnen? En upptäcktsfärd i den svenska digitaliseringens mångbottnade problemstruktur", FOI-R--4814--SE, 2020.

- [42] T. Gustavi, J. Karlholm, D. Oskarsson, T. Beran, O. Björnham, E. Gudmundson, H. Heden, H. Karlzén, J. Lindblom, J. Nordlöf, I. Rodhe, T. Sommestad, P. Svenmarck, och M. Svensson, "Försvarnsnära tillämpningar av Artificiell Intelligens", FOI-R--4707--SE, 2019.
- [43] Royal Society, "Machine learning: the power and promise of computers that learn by example", <https://royalsociety.org/topics-policy/projects/machine-learning/>, 2017.
- [44] C. Villani, "For a meaningful artificial intelligence: Towards a French and European strategy", <https://www.aiforhumanity.fr/>, 2018.
- [45] Regeringskansliet Näringsdepartementet, "Nationell inriktning för artificiell intelligens", <https://www.regeringen.se/informationsmaterial/2018/05/nationellinriktning-for-artificiell-intelligens>, 2018.
- [46] M. E. Gehm, R. John, D. J. Brady, R. M. Willett, och T. J. Schulz, "Single-shot compressive spectral imaging with a dual-disperser architecture", *Optics Express*, Vol. 15, Nr. 21, sidor 14013-14027, 2007/10/17 2007.
- [47] M. Nordberg och L. Landström, "UV Raman chemical imaging using compressed sensing", *Analyst*, Vol. 144, Nr. 5, sidor 1513-1518, 2019.
- [48] D. L. Donoho, "Compressed sensing", *IEEE Transactions on Information Theory*, Vol. 52, Nr. 4, sidor 1289-1306, 2006.
- [49] C. V. Raman, "A new radiation", *Indian Journal of Physics*, Vol. 2, sidor 387-398, 1928.
- [50] A. Wagadarikar, R. John, R. Willett, och D. Brady, "Single disperser design for coded aperture snapshot spectral imaging", *Applied Optics*, Vol. 47, Nr. 10, sidor B44-B51, 2008/04/01 2008.
- [51] G. R. Arce, D. J. Brady, L. Carin, H. Arguello, och D. S. Kittle, "Compressive Coded Aperture Spectral Imaging: An Introduction", *IEEE Signal Processing Magazine*, Vol. 31, Nr. 1, sidor 105-115, 2014.
- [52] J. M. Bioucas-Dias och M. A. T. Figueiredo, "A New TwIST: Two-Step Iterative Shrinkage/Thresholding Algorithms for Image Restoration", *IEEE Transactions on Image Processing*, Vol. 16, sidor 2992-3004, 2007.
- [53] M. A. T. Figueiredo, R. D. Nowak, och S. J. Wright, "Gradient Projection for Sparse Reconstruction: Application to Compressed Sensing and Other Inverse Problems", *IEEE Journal of Selected Topics in Signal Processing*, Vol. 1, Nr. 4, sidor 586-597, 2007.
- [54] D. Kittle, K. Choi, A. Wagadarikar, och D. J. Brady, "Multiframe image estimation for coded aperture snapshot spectral imagers", *Applied Optics*, Vol. 49, Nr. 36, sidor 6824-6833, 2010/12/20 2010.
- [55] "Deep learning.", MIT Press, 2016.
- [56] Y. Wu, M. Rosca, och T. Lillicrap, "Deep Compressed Sensing", *Proceedings of the 36th International Conference on Machine Learning*, Vol. 97, sidor 6850-6860, 2019.
- [57] X. Lu, W. Dong, P. Wang, G. Shi, och X. Xie, "ConvCSNet: A Convolutional Compressive Sensing Framework Based on Deep Learning", *CoRR*, Vol. abs/1801.10342, 2018.
- [58] D. P. Kingma och J. Ba, "Adam: A Method for Stochastic Optimization", *CoRR*, Vol. abs/1412.6980, 2015.
- [59] I. Goodfellow, Y. Bengio, och A. Courville, "Deep Learning", The MIT Press, 2016.
- [60] A. Bora, A. Jalal, E. Price, och A. G. Dimakis, "Compressed Sensing using Generative Models", i *ICML*, 2017.
- [61] K. Kulkarni, S. Lohit, P. K. Turaga, R. Kerviche, och A. Ashok, "ReconNet: Non-Iterative Reconstruction of Images from Compressively Sensed Random Measurements", *CoRR*, Vol. abs/1601.06892, 2016.
- [62] "ReconNet: Non-Iterative Reconstruction of Images from Compressively Sensed Measurements.", *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, sidor 449, 2016.
- [63] "Unsupervised representation learning with deep convolutional generative adversarial networks", *CoRR*, 2016.

- [64] A. Radford, L. Metz, och S. Chintala, "Unsupervised Representation Learning with Deep Convolutional Generative Adversarial Networks", 2015.
- [65] J. Gu, Y. Yang, och V. Tresp, "Understanding Individual Decisions of CNNs via Contrastive Backpropagation", i *ACCV*, 2018.
- [66] B. K. Iwana, R. Kuroki, och S. Uchida, "Explaining Convolutional Neural Networks using Softmax Gradient Layer-wise Relevance Propagation", *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, sidor 4176-4185, 2019.
- [67] S. Bach, A. Binder, G. Montavon, F. Klauschen, K.-R. Müller, och W. Samek, "On Pixel-Wise Explanations for Non-Linear Classifier Decisions by Layer-Wise Relevance Propagation", *PLoS one*, Vol. 10, Nr. 7, sidor e0130140-e0130140, 2015.
- [68] A. Dhurandhar, P.-Y. Chen, R. Luss, C.-C. Tu, P. Ting, K. Shanmugam, och P. Das, "Explanations based on the missing: towards contrastive explanations with pertinent negatives", *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, sidor 590–601, 2018.
- [69] S. Joshi, O. Koyejo, B. Kim, och J. Ghosh, "xGEMs: Generating Exemplars to Explain Black-Box Models", *ArXiv*, Vol. abs/1806.08867, 2018.
- [70] A. Feghahati, C. Shelton, M. J. Pazzani, och K. Tang, "CDeepEx: Contrastive Deep Explanations", 2018.
- [71] A. Dhurandhar, T. Pedapati, A. Balakrishnan, P.-Y. Chen, K. Shanmugam, och R. Puri, "Model Agnostic Contrastive Explanations for Structured Data", *ArXiv*, Vol. abs/1906.00117, 2019.
- [72] R. Luss, P.-Y. Chen, A. Dhurandhar, P. Sattigeri, K. Shanmugam, och C.-C. Tu, "Generating Contrastive Explanations with Monotonic Attribute Functions", *ArXiv*, Vol. abs/1905.12698, 2019.
- [73] S. Lapuschkin, A. Binder, G. Montavon, K.-R. Müller, och W. Samek, "LRP Toolbox for Artificial Neural Networks 1.2.0 – Manual", http://heatmapping.org/files/lrp_toolbox/documents/manual.pdf, 2016.

FOI är en huvudsakligen uppdragsfinansierad myndighet under Försvarsdepartementet. Kärnverksamheten är forskning, metod- och teknikutveckling till nytta för försvar och säkerhet. Organisationen har cirka 1000 anställda varav ungefär 800 är forskare. Detta gör organisationen till Sveriges största forskningsinstitut. FOI ger kunderna tillgång till ledande expertis inom ett stort antal tillämpningsområden såsom säkerhetspolitiska studier och analyser inom försvar och säkerhet, bedömning av olika typer av hot, system för ledning och hantering av kriser, skydd mot och hantering av farliga ämnen, IT-säkerhet och nya sensorers möjligheter.



FOI
Totalförsvarets forskningsinstitut
164 90 Stockholm

Tel: 08-55 50 30 00
Fax: 08-55 50 31 00

www.foi.se