



Universitetet
i Stavanger

DET TEKNISK-NATURVITENSKAPELIGE FAKULTET

MASTEROPPGAVE

Studieprogram/spesialisering: Signalbehandling	Vårsemesteret, 2011 Åpen / Konfidensiell
Forfatter: Håvard Eikeland (signatur forfatter)
Fagansvarlig: Trygve Eftestøl Veileder: Trygve Eftestøl	
Tittel på masteroppgaven: Deteksjon av brystkompresjoner i hjertestansdata Engelsk tittel: Detection of chest compressions from cardiac arrest data	
Studiepoeng: 30	
Emneord: Hjertestans, Impedansmåling, Brystkompresjoner, Toppdeteksjon	Sidetall: 37 + vedlegg/annet: CD Stavanger, 10.juni 2011

Sammendrag

Denne oppgaven omhandler deteksjon av brystkompresjoner i impedansmålinger gjort over hjertestanspasienters brystkasse. Til å begynne med redegjøres det for tidligere arbeid, grunnleggende teori om innsamling av impedansmålinger, litt om selve impedansmålingene og problemer med dem, som f.eks. støy, som en må ta hensyn til. Videre utledes fire metoder for deteksjon av brystkompresjoner. Det startes med en metode som består av kun toppdeteksjon og terskling. Terskelverdien som blir brukt er svært lav og fjerner bare støy med svært lav amplitude. Denne metoden hadde en del feildeteksjoner. Derfor ble metode 2 utviklet.

I metode 2 ble støy med lavere og høyere frekvens enn kompresjoner fjernet. Dette bidro til å redusere antall feildeteksjoner.

For å få ned antall feildeteksjoner ytterligere, ble glatting prøvd i metode 3. Dette viste seg å være effektivt.

Metode 4 ble laget for å fjerne støy som ikke ble fanget opp i metode 2. Den fjernet også støy basert på størrelsen av grupperinger av deteksjoner, og avstand mellom grupperingene.

Avslutningsvis sammenlignes og diskuteres resultatene de fire metodene oppnådde.

Forord

Denne oppgaven er gitt ved Universitetet i Stavanger og arbeidet med den er i sin helhet utført våren 2011, som siste del av masterstudiet i Kybernetikk/Signalbehandling. Rapporten er skrevet i L^AT_EX.

Jeg vil takke Trygve Eftestøl for god veiledning.

En stor takk går også til medstudent Pål Henning Olsen, som har spart meg for mye tid og frustrasjon med L^AT_EX og Matlab.

Innhold

1	Innledning	1
2	Teori	3
2.1	Tidligere arbeid	3
2.2	Grunnleggende om impedansmåling og -signaler:	3
2.2.1	Måling	3
2.2.2	Signalet	4
2.3	Toppdetektering	6
2.4	Annotering av kompresjoner	6
2.5	Vurdering av metodene	8
2.5.1	Parametre	8
2.5.2	Klassifisering av deteksjoner	8
2.6	Datamaterialet	11
2.7	Forbehandling av data	12
2.7.1	Støy	12
2.7.2	Tidsforskyvelse	13
2.7.3	Andre problemer	14
2.8	Metodene	14
2.8.1	Metode 1: Toppdetektering med terskling	14
2.8.2	Metode 2: Terskling og avstand	15
2.8.3	Metode 3: Med glatting	19
2.8.4	Metode 4: Fjerning av deteksjoner basert på grup- pering og avstand	20
3	Resultater	24

4	Diskusjon	25
4.1	Metode 1: Toppdetektering med terskling	25
4.2	Metode 2: Støyfjerning	25
4.3	Metode 3: Glatting	26
4.4	Metode 4: Fjerning av deteksjoner basert på gruppering og avstand	27
5	Konklusjon	29
A	Resultater for alle datasett	33
A.1	Metode 1	33
A.1.1	Treningsdata	33
A.1.2	Testdata	33
A.2	Metode 2	34
A.2.1	Treningsdata	34
A.2.2	Testdata	34
A.3	Metode 3	35
A.3.1	Treningsdata	35
A.3.2	Testdata	35
A.4	Metode 4	36
A.4.1	Treningsdata	36
A.4.2	Testdata	36
B	Matlabkode	37

1 Innledning

Det er flere årsaker til at personer får hjertestans. Ofte kommer det av at tilførselen av blod til selve hjertemuskelen er forstyrret, et hjerteinfarkt, som over tid kan gi så store skader på hjertemuskulaturen at det stopper. Andre årsaker kan være medfødte defekter på hjertet eller ulykker. Felles for alle former for hjertestans er at hjertet slutter å pumpe blod. Når hjertet slutter å pumpe blod, blir ikke blod pumpet til lungene for å ta opp oksygen og videre ut i kroppen for å tilføre alle deler av kroppen oksygen. Mangel på oksygen kan f.eks. føre til permanente hjerneskader, og i ytterste konsekvens døden. Hjerte-lunge-redning, heretter forkortet HLR, er derfor viktig for å tilføre nytt oksygen til blodet, gjennom innblåsninger, og å få det oksygenberikede blodet fra lungene rundt i kroppen, gjennom brystkompresjoner. Hvordan brystkompresjoner og innblåsninger skal gis har variert opp gjennom tidene, men gjeldende retningslinjer sier at kompresjoner og innblåsninger skal gis i et 30:2 forhold, altså først 30 brystkompresjoner etterfulgt av 2 innblåsninger. Det er også gitt i retningslinjene hvilken frekvens og dybde kompresjoner og innblåsninger skal ha[11, 21].

HLR alene har altså til hensikt å hindre skader grunnet oksygenmangel, og er som oftest ikke tilstrekkelig for oppnå egensirkulasjon, altså at hjertet selv pumper blod igjen. Som regel må det defibrillering til for å oppnå egensirkulasjon[18]. Defibrillering er når en plasserer elektroder på hjertestanspasientens brystkasse og sender et kortvarig støt gjennom brystkassen. Dette gjøres for å stanse eventuelle rytmeforstyrrelser, rytmeforstyrrelser kan føre til at hjertet ikke pumper tilstrekkelig med blod, og gjenopprette normal sinusrytme, det vil si at hjertet pumper som normalt[22, 23].

Selv om det altså er gitt retningslinjer hvordan HLR skal utføres [11] forekommer det variasjoner i utførelsen, også blant trent medisinsk personell[5, 6]. Kvaliteten på HLRen påvirker utfallet for pasienten som har hatt hjertestans. HLR med bedre kvalitet fører til bedre sjanser for at en person skal overleve, og bedre sjanse for å unngå permanente skader[7, 8, 9].

Derfor er det ønskelig å gjøre HLRen best mulig. For å oppnå dette er det utviklet utstyr som overvåker HLRen, og gir tilbakemeldinger om kvaliteten. Med kvalitet menes det her om HLRen er i tråd med gjeldende retningslinjer [11].

For å kunne si noe om kvaliteten må en hente ut informasjon om noen parametre, som f.eks. kompresjonsdybde og kompresjonsfrekvens.

Informasjon om kompresjonsfrekvens blir blant annet brukt til å estimere og redusere HLR-artefaktene i EKG-signaler[16]. Dette for å unngå og påvirke de automatiserte eksterne defibrillatorene som ser på EKG-signalet for å bestemme om det er noen sjokkbare rytmer tilstede[17].

Denne oppgaven omhandler det å detektere brystkompresjoner i impedanssignaler, ved bruk av impedanssignalet alene. Eventuelle kompresjoner vil dukke opp som topper i impedanssignalet. Hovedvekten er derfor lagt på metoder som baserer seg på å finne toppunkter i impedanssignalene.

Opgaven er bygget opp slik at det i kapittel 2 redegjøres for tidligere arbeid, grunnleggende teori som ligger bak de forskjellige metodene som er brukt og en gjennomgang av selve metodene. Alle metodene som presenteres bygger på hverandre, metode 2 vil derfor være en videreutvikling av metode 1 osv. Kapittel 3 består av en enkel presentasjon av resultatene en fikk ved utførelsen av eksperimentene. Kapittel 4 inneholder en diskusjon av resultatene fra kapittel 3. I kapittel 5 finner vi en konklusjonen for arbeidet.

2 Teori

Dette kapittelet gir en oversikt over tidligere arbeid som er gjort, teori som oppgaven bygger på, dataene som er brukt, metodene som er utviklet og brukt og hvordan metodene er evaluert opp mot hverandre. Alt arbeid med metoder og eksperimenter ble utført i Matlab.

2.1 Tidligere arbeid

Som nevnt innledningsvis er det laget utstyr for å overvåke og å gi tilbakemeldinger på HLR som utføres på en pasient. F.eks ble det i sammendraget i [10] foreslått å utvikle defibrillatorer med mulighet til å overvåke HLRen som blir utført. Arbeidet etter og rundt dette har ført til hjelpemidler som

“Q-CPR measurement and feedback tool”[15]. Dette er nettopp en defibrillator som gir tilbakemeldinger på HLR. Den har i tillegg defibrilatorelektrodenes kraftmåler, som måler dybden på kompresjonene og samtidig gir informasjon om frekvensen. Det er allerede også utviklet defibrillatorer som filtrerer bort HLR-artefakter, Zoll R Series ALS/PLUS. Her er dog diagnosen basert på det filtrerte signalet ikke pålitelig nok[16].

Det er også laget en applikasjon til iPhone-telefoner som gir tilbakemelding på kvaliteten på HLR[14]. Disse applikasjonene henter informasjonen den trenger fra akselerometeret i telefonen. Da de fleste mobiltelefoner i dag er utstyrt med slike akselerometer, kan i teorien alle disse telefonene brukes til å gi tilbakemeldinger på HLR-kvalitet. Det var dog sparsommelig med informasjon om nøyaktigheten til oppsettet med iPhone og HLR-applikasjonen på fabrikantens hjemmeside. Det er derimot en spennende tanke at alle til enhver tid kan ha med seg en telefon som kan gjøre dem i stand til å gi HLR i best mulig kvalitet.

2.2 Grunnleggende om impedansmåling og -signaler:

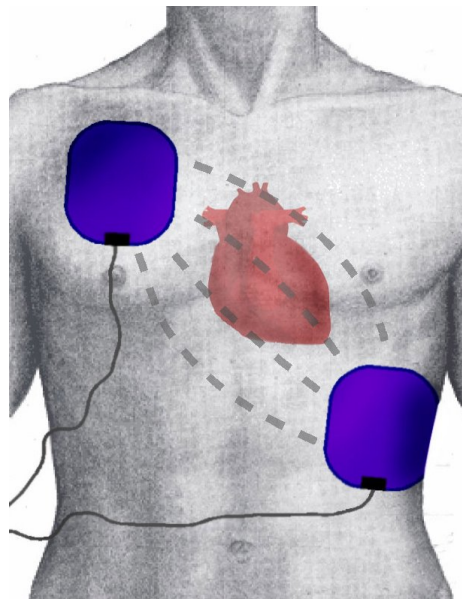
2.2.1 Måling

Impedanssignaler får en ved å utføre impedansmålinger gjennom thorax, brystkassen. Impedansmålinger over thorax foregår, generelt sett, ved at en plasserer to elektroder på pasientens brystkasse. Det sendes så en svak strøm, I , med kjent amplitude mellom disse elektrodene. Spenningen, U , over brystet måles

også av disse elektrodene. Når en så vet hvor mye strøm som går gjennom, og hvor mye spenning som ligger over, thorax kan en ved bruk av Ohms lov finne impedansen, Z :

$$Z = \frac{U}{I}$$

Impedansdataene brukt i denne oppgaven er hentet fra elektrodene på en defibrillator. Typisk plassering av elektrodene til en defibrillator er vist under:



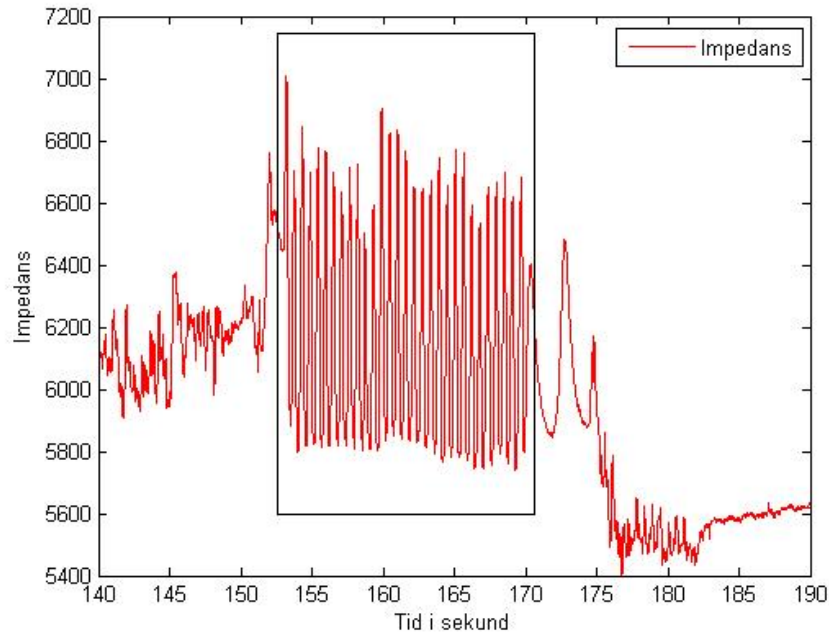
Figur 2.1: Oppsett for impedansmåling[20]

2.2.2 Signalet

Impedansen i brystet vil variere, avhengig av blant annet hvor mye luft det er i lungene og hvor mye blod det er i vevet mellom elektrodene. Følgelig kan impedansmålingen gi informasjon om eventuell sirkulasjon og ventileringer[3, 4]. Impedansen vil dog også variere p.g.a. en rekke andre faktorer, f.eks. vil det å flytte pasienten kunne gi store utslag på en impedansmåling. Det at impedansen påvirkes av en rekke faktorer gjør at et slikt signal kan være svært støybefengt. Slik støy kan vanskeliggjøre detekteringen og i verste fall også umuliggjør deteksjon av kompresjoner, da disse drukner i støy. Dette er noe en må ta hensyn til

når en skal lage metoder for å detektere eventuelle kompresjoner. Støyen kan også ligne på kompresjoner og føre til falske deteksjoner.

Under vises et utsnitt fra en impedansmåling:



Figur 2.2: Plottede impedansdata

Området innenfor boksen i plottet inneholder en serie med kompresjoner.

I figuren over, Figur 2.2, ser vi at fra 140 til 150s har vi en del støy med høy frekvens og lav amplitude. Etter boksen i plottet ser vi at vi har to topper med omtrent samme amplitude som kompresjonene, men disse har større avstand mellom seg, altså lavere frekvens. Denne forskjellen i amplitude og frekvens mellom støy og kompresjoner kan brukes når en skal lage metoder som skal detektere kompresjoner.

I Figur 2.2 ser en også at det er baselinedrift¹, som en må ta hensyn til, i impedanssignalet.

¹Baselinedrift vil si at gjennomsnittsverdien for signalet endres over tid. På Figur 2.2 ser en at gjennomsnittsverdien fra 155s til 170s er ca 6300, mens den fra 180s og utover ligger på ca 5500. Dette er baselinedrift.

2.3 Toppdetektering

Toppdeteksjon går ut på å finne topppunkter i et signal. Måten topper blir funnet på i denne oppgaven er at en enkelt og greit sammenligner et punkt med sine nabopunkter, hvis punktet har større amplitude enn sine naboer er det et toppunkt. Denne testen blir kjørt på alle punktene i målingene, fra første til siste punkt. En finner da alle eventuelle topppunkter i målingen.

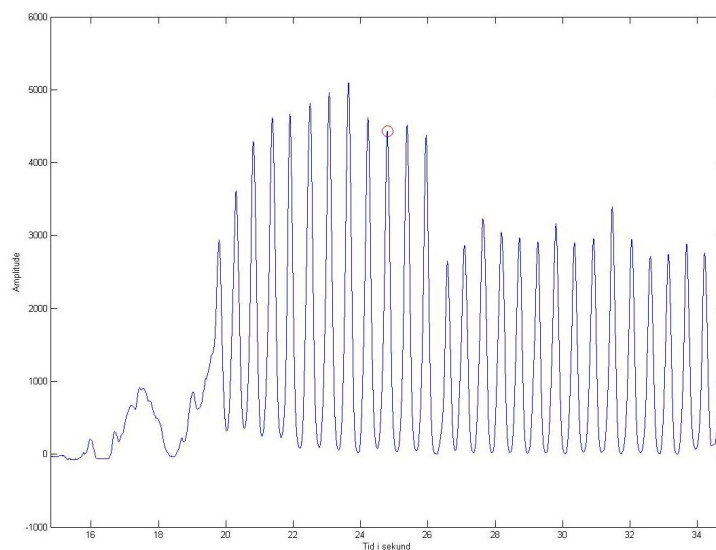
Alle de toppene som blir funnet blir lagret som mulige deteksjoner. De blir lagret på tilsvarende måte som de annoterte kompresjonene, som introduseres i neste delkapittel, kapittel 2.4.

Når det senere i oppgaven beskrives hvordan støytopper fjernes, menes det at de aktuelle deteksjonene, tilhørende støytoppene, blir fjernet fra lista med alle deteksjonene som er gjort for den aktuelle målingen.

2.4 Annotering av kompresjoner

For å kunne måle ytelsen til de forskjellige detekteringsmetodene, trenger en en “fasit” for å kunne avgjøre om deteksjonene de har gjort er korrekte eller ikke. For å lage en slik fasit ble det i denne oppgaven brukt signalet fra en kraftmåler, som måler kraften av eventuelle brystkompresjoner. Signalet fra kraftmåleren er så og si støyfritt, og egnert seg derfor godt til annotering av kompresjoner. Det er heller ingen baseline drift i disse signalene. En typisk serie med kompresjoner i et slikt signal er vist i Figur 2.3 på neste side.

Annoteringen av kompresjoner i kraftsignalet ble utført på den måten at hvert toppunkt som ble funnet i kraftsignalet ble merket og plottet, som vist i Figur 2.3 på neste side. Det aktuelle toppunktet ble merket med en rød sirkel. For hvert toppunkt som ble plottet ble det bestemt om det var en kompresjon eller ikke. Slik ble alle toppene gjennomgått én etter én.



Figur 2.3: Annotering av kompresjoner

Alle toppene som ble godkjent som kompresjoner ble deretter lagret som en egen fil. Toppene ble lagret som to tallverdier, en verdi som beskrev amplituden og en som beskrev hvor i tid i signalet den aktuelle kompresjonen befinner seg. Et utsnitt fra en slik fil er vist under:

19801	2934
20305	3608
20817	4288
21381	4613
21905	4658
22505	4810
23069	4963
23641	5094
24217	4615
24793	4425

Tabell 1: Kompresjonsdata

Den første kolonnen viser toppunktets plassering, samplenummer, i signalet. Den andre kolonnen viser toppunktets amplitude.

2.5 Vurdering av metodene

2.5.1 Parametre

For å vurdere de forskjellige metodene mot hverandre, regner vi ut de tre parametrene P_D , P_F og P_M [12] for hver av disse metodene. Der P_D er sannsynligheten for at en kompresjon blir korrekt detektert. Vi estimerer denne parameteren slik:

$$\hat{P}_D = \frac{N_D}{N_D + N_M}$$

der N_D er antall korrekt detekterte kompresjoner, og N_M er antall udetekterte kompresjoner.

P_F er sannsynligheten for en falsk deteksjon, altså at en topp som ikke er en kompresjon, blir godkjent som en kompresjon. Denne parameteren estimeres som følger:

$$\hat{P}_F = \frac{N_F}{N_D + N_F}$$

der N_D er antall korrekt detekterte kompresjoner, og N_F er antall falske deteksjoner.

P_M er sannsynligheten for at en kompresjon ikke blir detektert. Den har følgende sammenheng med P_D :

$$P_D = 1 - P_M$$

og estimeres følgelig som:

$$\hat{P}_M = 1 - \hat{P}_D$$

Her trenger en altså å vite antall korrekt detekterte kompresjoner, N_D , antall udetekterte kompresjoner, N_M , og antall falske deteksjoner, N_F . For å finne dette bruker vi de annoterte kompresjonene, “fasiten” vår.

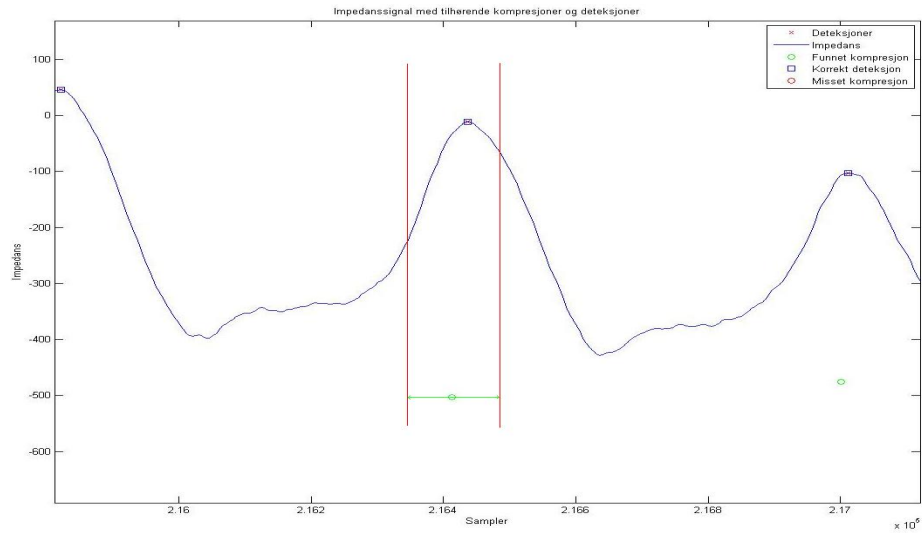
2.5.2 Klassifisering av deteksjoner

For å finne N_D , N_M og N_F ble det laget en algoritme, som sammenlignet deteksjoner i impedansdataene med de annoterte kompresjonene for hver måling.

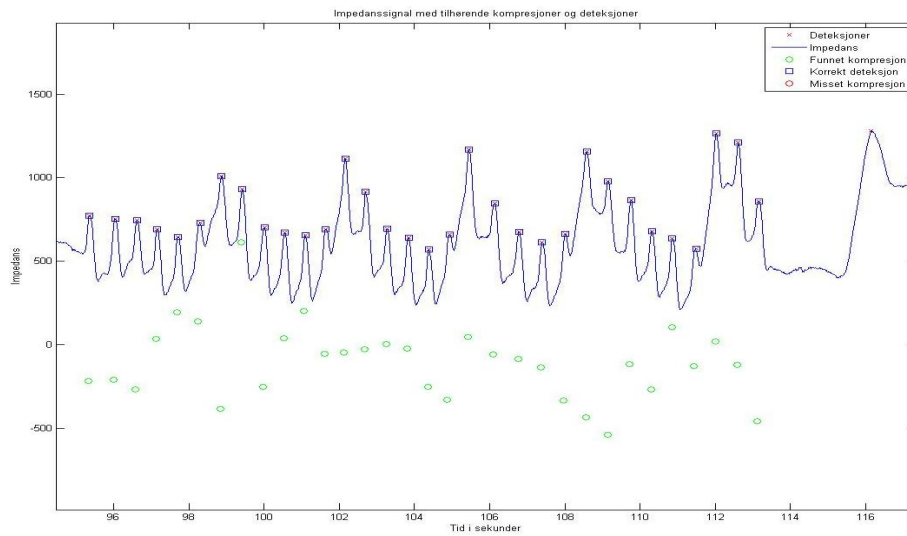
- N_D ble funnet ved at en starter med den første annoterte kompresjonen i fasiten, og sjekker om det finnes en deteksjon som ligger tilstrekkelig nær i tid, i impedanssignalet, til at den skal godkjennes som en korrekt detektert kompresjon. Med tilstrekkelig nær menes innenfor et fastsatt intervall², se Figur 2.4 på neste side. Eventuelle korrekte deteksjoner ble lagret på tilsvarende måte som de annoterte kompresjonene i Tabell 1 på side 7. Antall korrekte deteksjoner tilsvarte da N_D .
- N_M ble funnet på en lignende måte som N_D . Forskjellen er at for N_M ble annoterte kompresjoner der det ikke ble funnet noen deteksjoner tilstrekkelig nær lagret. Antall annoterte kompresjoner uten noen tilhørende deteksjoner tilsvarte N_M .
- N_F ble funnet ved at alle de godkjente deteksjonene, i N_D , ble trukket fra den fullstendige “lista” med deteksjonene som tilhørte den aktuelle målingen. Følgelig ble man stående igjen med deteksjoner som ikke var godkjent som korrekte. Antallet ikke godkjente kompresjoner tilsvarte da N_F .

I Figur 2.5 på neste side vises et utsnitt fra plottede data der deteksjoner og annoterte kompresjoner er merket:

²Intervallet ble satt til ca 60 sampler i hver retning fra fasitpunktet. Dette tallet kom en fram til ved å finne gjennomsnittsavstanden mellom kompresjonene i fasitene, som ble 140 sampler, og dele resultatet på to. Det ble også lagt inn en “buffer” på 10 sampler. Det ble også satt opp et 99% konfidensintervall for avstanden mellom kompresjonene for å være sikker på at intervallet var mindre enn nedre grense i konfidensintervallet. Nedre grense i konfidensintervallet ble forøvrig 117 sampler. Dette ble gjort for å unngå at det ble overlapp mellom godkjenningsintervallene til to nabo kompresjoner, men samtidig ha litt “å gå på” da det var noe tidsforskyvelse mellom enkelte fasitsignal og impedanssignal. Dette kommer en tilbake til i 2.7.2.



Figur 2.4: Illustrasjon av godkjenningsintervallet rundt en annotert kompresjon. Så lenge en deteksjon ligger mellom de røde linjene er den nær nok den annoterte kompresjonen til å kunne godkjennes.



Figur 2.5: Impedanssignal med tilhørende kompresjoner og deteksjoner. Sirkelene under impedanssignalet angir plasseringen til annoterte kompresjonene fra kraftsignalet.

2.6 Datamaterialet

Datamaterialet som er brukt i denne oppgaven består av ett sett med data kalt treningsdata og ett sett kalt testdata. Hvert av datasettene inneholder impedansmålinger, som dem beskrevet i kapittel 2.2, og tilhørende kraftsignaler, som beskrevet i kapittel 2.4. Treningsdataene består av tre målinger og testdataene består av tolv målinger. Treningsdataene blir brukt under utvikling av metodene. Testdataene blir brukt til å verifisere resultatene en oppnår på treningsdataene. Dette er viktig for å forsikre seg om at metodene ikke er overtrente. Overtrening vil si at metodene er tilpasset treningsdataene i for stor grad, og ikke vil fungere tilfredsstillende for andre data. Med andre ord vil en da gjerne få veldig gode resultater for treningsdataene, og mindre bra resultater for testdataene. I tabellene på neste side, Tabell 2 og Tabell 3, presenteres alle målingene som er brukt i tabellform.

Treningsdata:	Antall kompresjoner i målingen:
Pasient2	155
Pasient3	585
Pasient5	545

Tabell 2: Treningsdata brukt i arbeidet

Testdata:	Antall kompresjoner i målingen:
Pasient6	1001
Pasient9	383
Pasient10	1870
Pasient11	581
Pasient12	1572
Pasient13	522
Pasient16	277
Pasient19	1308
Pasient20	2427
Pasient22	2126
Pasient24	126
Pasient25	1393

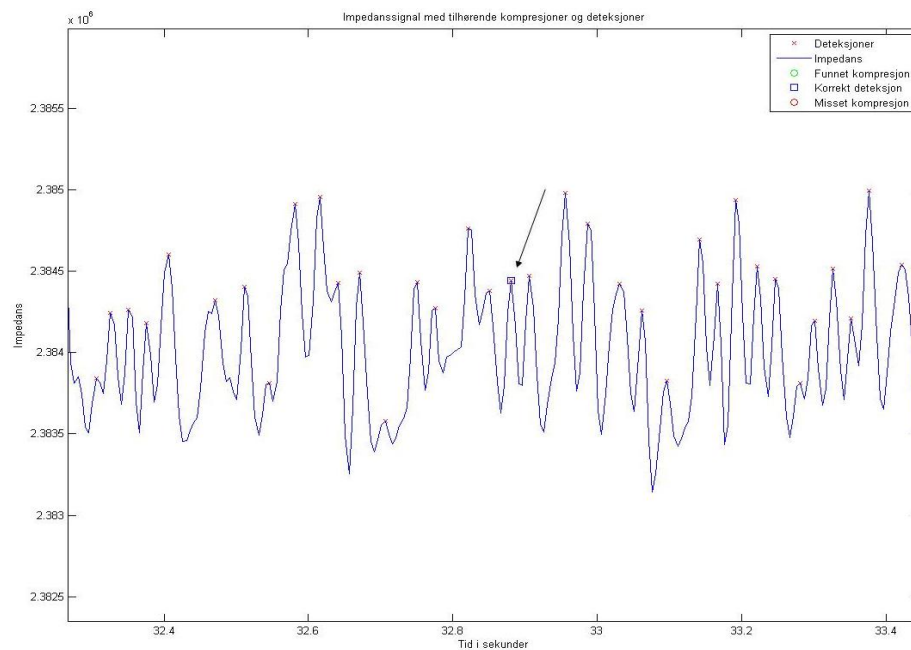
Tabell 3: Testdata brukt i arbeidet

Som en ser på nummereringen er det noen datasett som mangler. Dette kommer av at ikke alle datasettene var brukbare og måtte fjernes. Totalt ble 10 av 25 datasett fjernet.

2.7 Forbehandling av data

2.7.1 Støy

Noen av målingene som ble brukt i forsøkene krevde noe forbehandling for å kunne brukes. Flere av dem inneholdt områder med hvit støy der det var umulig å detektere kompresjoner som var tilstede. Disse områdene ble derfor fjernet³ fra datasettene de opptrådte i. Under vises et utsnitt fra en måling der slik støy opptrer.



Figur 2.6: Støy og feilaktig godkjent deteksjon

Pila markerer en topp som ligger der det skal være en kompresjon i følge fasiten. Denne toppen har ingenting med en eventuell kompresjon å gjøre, og er bare et

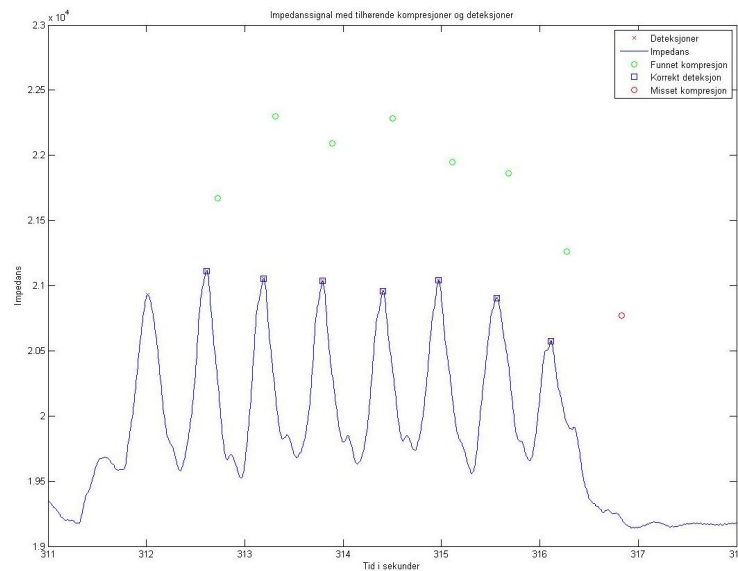
³Gitt verdien NaN(Not a Numer), og dermed ekskludert fra eksperimentene.

resultat av en tilfeldig topp i støyen som har havnet akkurat der fasiten sier at det skal være en kompresjon.

2.7.2 Tidsforskyvelse

I andre datasett var signalet fra kraftmåleren, det som ble brukt som fasit, og impedanssignalet, som detekteringen skulle utføres på, forskjøvet i tid i forhold til hverandre. Hvor mye de var forskjøvet varierte litt, men i noen tilfeller var de forskjøvet så mye at det ville være umulig å bruke kraftsignalet som fasit. Dette grunnet at annoterte kompresjoner sammenfalt i tid med nabokompresjonen til kompresjonen som den egentlig tilhørte, se Figur 2.7.

I tillegg var datasettene gjerne synkronisert i starten av dataene, men ble så gradvis mer forskjøvet. Dette umuliggjorde det å eventuelt synkronisere datasettene ved bare å forskyve et av datasettene med en konstant. Disse datasettene med forskyvelser ble derfor ekskludert fra dette arbeidet.



Figur 2.7: forskjøvet data

I figuren over, Figur 2.7, ser vi at vi har en serie på åtte kompresjoner og åtte annoterte kompresjoner. Da disse er forskjøvet i forhold til hverandre blir den første toppen ikke godkjent som en korrekt deteksjon, som den egentlig er. Den siste annoterte kompresjonen blir satt som ikke funnet.

Totalt 7 av de 10 datasettene som ble fjernet, ble fjernet grunnet tidsforskyvelse.

2.7.3 Andre problemer

Et par datasett måtte også fjernes fra arbeidet da de var for store for programvaren som ble brukt. Programvaren klarte ikke håndtere dem.

Ett datasett ble fjernet da det kun bestod av en impedansmåling, og ikke hadde noe kraftsignal som kunne brukes som fasit.

2.8 Metodene

I dette delkapittelet presenteres de forskjellige “metodene” som ble brukt til selve detekteringen. Metode 1, 2 o.s.v. er kanskje ikke de beste navnene en kan bruke, men de gjør det lettere å referere tilbake til de enkelte “metodene”.

2.8.1 Metode 1: Toppdetektering med terskling

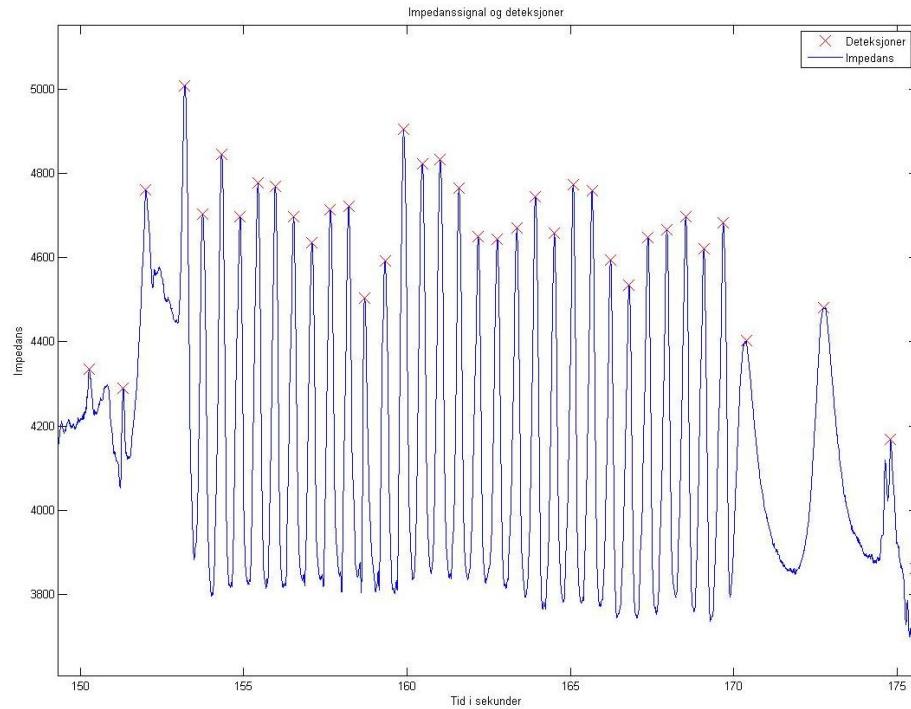
Den første metoden som ble brukt for å detektere kompresjoner var en enkel toppdeteksjonsmetode. Det ble valgt en toppdeteksjonsmetode fordi eventuelle kompresjoner vil, som nevnt innledningsvis, vises som topper i impedanssignalet, slik en ser i Figur 2.8 på neste side. I denne metoden benyttes en algoritme[13] som går gjennom impedansdataene og gir ut ett datasett med alle topppunkter. Toppunktene beskrives med to verdier; én for amplitude og én for tid, altså tiden der toppunktet opptrer i impedansdataene, tilsvarende som for dataene i Tabell 1 på side 7.

For at et toppunkt skal detekteres må det oppfylle et krav; amplitudeforskjellen mellom toppunkt og hosliggende bunnpunkt må være større enn en gitt terskelverdi⁴. Denne terskelverdien er satt relativt lav for å ikke risikere å sile ut kompresjoner med lav amplitude. Støy med svært lav amplitude, sammenlignet med eventuelle kompresjoner, vil likevel bli ekskludert.

Det at terskelverdien kun setter krav til amplitudeforskjell mellom hosliggende topp- og bunnpunkter gjør at en, forhåpentligvis, slipper å ta hensyn til eventuell baselinedrift.

⁴Terskelverdien ble her satt til 100. Amplituden på en typisk kompresjon er ca 1000. Altså 10 ganger høyere enn terskelverdien.

I figuren , Figur 2.8, har vi plottet impedansdata og toppunktene som er funnet i impedansdataene sammen:



Figur 2.8: Kompresjoner og deteksjoner

Som vi ser er de toppene med lavest amplitude ikke tatt med av deteksjonsalgoritmen.

2.8.2 Metode 2: Terskling og avstand

Den andre metoden er i utgangspunktet identisk med den første metoden, men her er det tatt noe hensyn til støy. For å fjerne støyen ble et sett med logiske tester kjørt på dataene. Det var to tester som ble utført, begge gikk på avstand mellom topper i en gruppe nabotopper. Bakgrunnen for disse to reglene er at det er gitt i retningslinjer[11] hva frekvensen på brystkompresjoner bør være⁵. En kan da anta at alt med høyere eller lavere frekvens enn dette kan betraktes

⁵Det er gitt i European Resuscitation Council (ERC) Guidelines for Resuscitation 2005 at frekvensen på kompresjonene skal være ca 100 kompresjoner pr min.

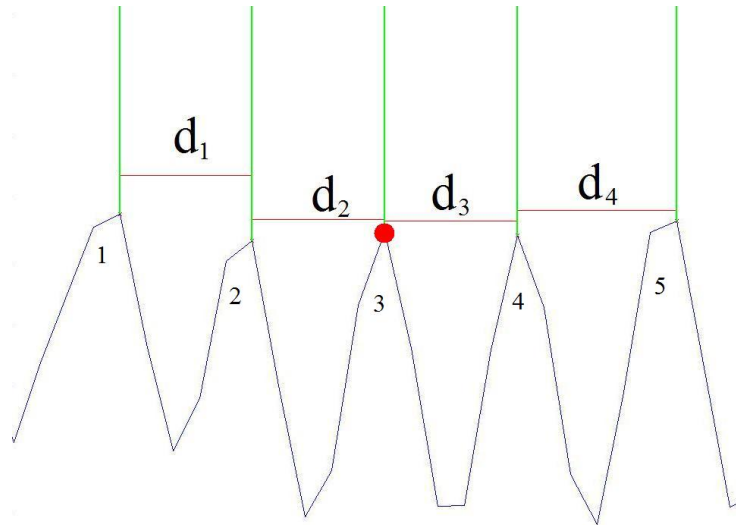
som støy, og trygt kan fjernes.

Den første regelen så på 5 nabotopper samlet. Hvis avstanden d_1 mellom topp 1 og 2, avstanden d_2 mellom topp 2 og 3, avstanden d_3 mellom topp 3 og 4 og avstanden d_4 mellom topp 4 og 5 var mindre enn avstanden $D_{h\phi y}$, ville den midterste, topp 3, bli fjernet. Topp 3 fjernes da den trolig er et resultat av støy, ettersom toppene ligger nærmere hverandre enn det kompresjoner vil gjøre, se Figur 2.9 på neste side.

Avstanden $D_{h\phi y}$ var satt såpass liten at den ville være mye mindre enn avstanden mellom to sanne kompresjoner. Den kalles D”Høy” da avstanden er satt slik at den fjerner høyfrekvent støy.

Verdien for $D_{h\phi y}$ ble funnet ved “prøving og feiling”. Først ble avstanden mellom alle kompresjonene i fasitene for treningsdataene funnet. Deretter ble den største og minste avstanden funnet. Den minste avstanden funnet mellom to kompresjoner her ble et utgangspunkt for prøvingen og feilingen. De verdiene som ble prøvd måtte være mindre enn den minste avstanden funnet mellom to kompresjoner, for å unngå å fjerne sanne kompresjoner i trenings- og testdataene. Det ble lagt inn en buffer også i tilfelle den minste verdien i treningsdataene ikke skulle være representativ for avstandene mellom eventuelle kompresjoner i testdataene. Deretter ble det prøvd ut en rekke verdier for $D_{h\phi y}$, så ble den verdien⁶ som ga best verdi for P_D og P_F valgt.

⁶Verdien som ble valgt for $D_{h\phi y}$ var 80 sampler.



Figur 2.9: Illustrasjon av avstandene d_1 , d_2 , d_3 og d_4

Regelen kort oppsummert: $d_1 \& d_2 \& d_3 \& d_4 < D_{h\phi y} \Rightarrow$ Topp 3, i Figur 2.9, merket med en rød prikk, fjernes. Hvis ikke dette kriteriet var oppfylt ble topp 3 beholdt.

Avstandene d ble funnet ved $d_i = |t_i - t_{i+1}|$ der $i = 1, 2, 3 \dots n - 1$, der n er antall topper i gjeldene impedansdata. t er plasseringen, samplenummer, til toppunktet.

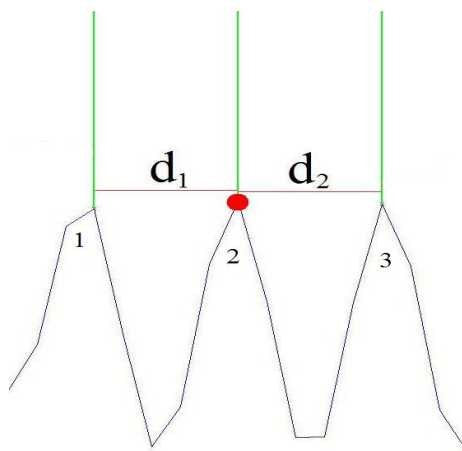
Regelen formulert mer generelt:

- $|t_{i-2} - t_{i-1}| \& |t_{i-1} - t_i| \& |t_i - t_{i+1}| \& |t_{i+1} - t_{i+2}| < D_{h\phi y} \Rightarrow t_i$ fjernes fra datasettet med toppunkter.
- ellers beholdes toppunktet t_i i datasettet med toppunkter
der t_i er plasseringen, samplenummeret, til topp i , der $i = 3, 4, 5 \dots n - 2$ og n er antall topper i impedansdataene.

Der avstanden $D_{h\phi y}$ altså er satt mindre enn det avstanden mellom to kompresjoner kan være. Dette fører til at vi får fjernet høyfrekvent støy, slik som vist i Figur 4.1 på side 26.

Den andre regelen fjerner støy med lavere frekvens enn kompresjonene. Denne regelen så på tre nabotopper.

Regelen var at dersom avstanden d_1 mellom topp 1 og 2 og avstand d_2 mellom topp 2 og 3 var større enn D_{lav} ⁷, ville den midterste toppen, topp 2, fjernes. Se Figur 2.10 under. D_{lav} ble funnet på tilsvarende måte som $P_{høy}$ på side 16, men her var det den største avstanden mellom to kompresjoner i fasiten som ble brukt som utgangspunkt.



Figur 2.10: Illustrasjon av avstandene d_1 og d_2 .

Altså: $d_1 \& d_2 > D_{lav} \Rightarrow$ Topp 2, merket med en rød prikk, fjernes.

Avstandene d ble funnet på tilsvarende måte som ved foregående regel ved

$d_i = |t_i - t_{i+1}|$ der $i = 1, 2, 3 \dots n - 1$, der n er antall topper i gjeldene impedansdata. t er plasseringen, samplenummer, til toppunktet.

Regelen formulert mer generelt:

- $|t_{i-1} - t_i| \& |t_i - t_{i+1}| > D_{lav} \Rightarrow t_i$ fjernes fra datasettet med topppunkter.
- ellers beholdes toppunktet t_i i datasettet med topppunkter.
der t_i er plasseringen, samplenummeret, til topp i , der $i = 3, 4, 5 \dots n - 2$ og n er antall topper i impedansdataene.

Som nevnt er D_{lav} satt høyere enn det avstanden mellom to kompresjoner i en serie vil være. En vil dermed sile ut støy med lavere frekvens enn eventuelle kompresjoner. Denne metoden vil dog også fjerne “enslige” kompresjoner, altså

⁷ D_{lav} ble satt til 180 sampler.

kompresjoner som kommer alene og ikke i en serie med andre. Da kompresjoner skal gis i serier på 30 i følge retningslinjene antar en at “enslige” kompresjoner opptrer såpass sjeldent at dette ikke har noen reell betydning for det totale resultatet.

Da denne metoden sammeligner avstand mellom henholdsvis tre og fem detekterte toppunkter, vil ikke de aller første og siste detekterte toppene bli tatt med av denne metoden. Dette fordi de ytterste toppene ikke har noen naboer en kan teste avstanden til.

2.8.3 Metode 3: Med glatting

Da noen av impedanssignalene inneholder en del støy med relativt lav amplitude, sammenlignet med amplituden til kompresjonene, var det naturlig å prøve glatting for å redusere denne støyen. Det ble derfor lagt inn glatting i samme algoritme som benyttes i Metode 2, for å se om det kunne ha noen positiv effekt på P_D eller P_F . Glattingen ble gjort ved bruk av et bevegelig gjennomsnitt, med en vindusstørrelse på 35.

Glatting med bevegelig gjennomsnitt foregår ved at hvert sample blir byttet ut med et gjennomsnittet av seg selv og naboene i vinduet:

$$y_s(i) = \frac{1}{2N+1}(y(i+N) + y(i+N-1) + \dots + y(i-N)) \quad (1)$$

hvor $y_s(i)$ er det i -te glattede punktet, N er antall naboer på hver side av $y_s(i)$ i vinduet, altså er størrelsen til vinduet $2N+1$.

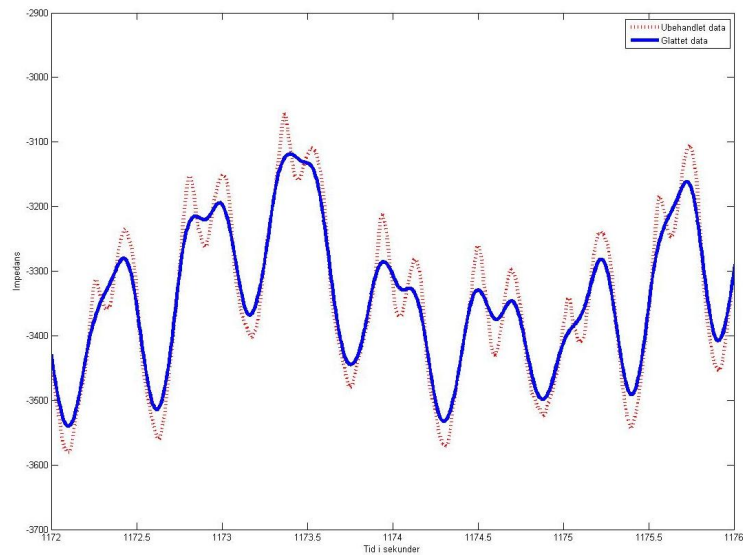
Da vindusstørrelsen som ble brukt her var 35 blir ligning 1 seende slik ut:

$$y_s(i) = \frac{1}{35}(y(i+17) + y(i+16) + \dots + y(i-17)) \quad (2)$$

Utleddningen ovenfor er stort sett hentet fra Mathworks⁸ sine hjemmesider[19].

På side 20 vises et eksempel på hva glattingen ideelt sett kan bidra til.

⁸Produsenten av Matlab.



Figur 2.11: Eksempel på glatting

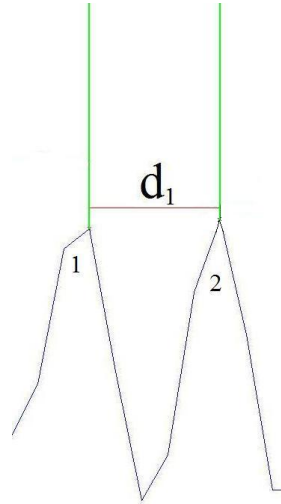
Som en ser får en i glatteprosessen jevnet ut noen støytopper. Dette kan være fordelaktig da kompresjonsdetekteringen baserer seg på toppdeteksjon.

Denne metoden tar ikke med de ytterste, de første og siste, 16 samplene. Dette er på grunn av vindusstørrelsen på 35. Det sampelet som skal glattes må ha 17 sampler på hver siden av seg for å kunne fylle vinduet, som altså har en størrelse på 35 sampler. Da impedanssignalet er samplet med en samplingsfrekvens på 200 Hz blir det ubetydelig om 16 sampler helt ytterst ikke blir glattet. 16 sampler tilsvarer $0,08$ sekunder. Til sammenligning er det $0,6$ sekunder mellom hver kompresjon, gitt i henhold til retningslinjene.

2.8.4 Metode 4: Fjerning av deteksjoner basert på gruppering og avstand

Da metode 2 ikke tar hensyn til de aller første og siste deteksjonene i et signal, vil det være noen feildeteksjoner som ikke blir oppdaget. Det er derfor her lagt til en test på de første og siste toppene. Denne testen går i all enkelhet ut på å teste om toppene er i en serie med andre topper eller ikke. Måten det gjøres på er at en finner avstanden mellom den aktuelle deteksjonen og dens nærmeste nabo. Hvis denne avstanden er for stor, til at de to kan være i en serie, kan

en anta at de to ikke er kompresjoner i en serie, og den aktuelle toppen blir ikke lenger merket som en deteksjon. Denne toppen vil da bli ekskludert fra det videre arbeidet med den gjeldende impedansmålingen.



Figur 2.12: Illustrasjon av avstanden d_1

Hvis topp 1 i figuren over, Figur 2.12, representerer første deteksjonen gjort i en impedansmåling, vil det testes om $d_1 > D$. Der D er en fastsatt avstand som er større enn det avstanden mellom to kompresjoner i en serie vil være. Hvis dette er tilfellet fjernes første deteksjon fra datasettet. Hvis ikke dette kriteriet var oppfylt, ble topp 1 beholdt.

Altså:

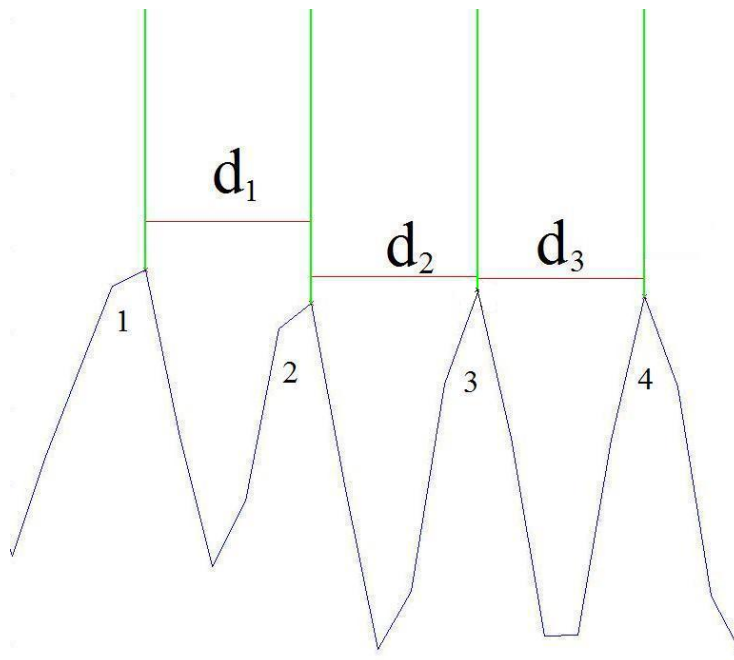
- $|t_1 - t_2| > D \Rightarrow t_1$ fjernes fra datasettet med topppunkter.
- ellers beholdes toppunktet t_1 i datasettet med topppunkter

Hvis en så antar at topp 2 i Figur 2.12 representerer siste deteksjon i et datasett, blir det fortsatt testet om $d_1 > D$. Topp 2 vil så fjernes hvis $d_1 > D$ er oppfylt.

- $|t_{n-1} - t_n| > D \Rightarrow t_n$ fjernes fra datasettet med topppunkter.
- ellers beholdes toppunktet t_n i datasettet med topppunkter

der t_i er plasseringen i tid, samplenummeret, til topp i , der $i = 1, 2, 3 \dots n-2$ og n er antall topper i impedansdataene.

Det at kompresjoner ifølge rettningslinjer[11] skal komme i serier på 30 gjør at det i utgangspunktet er nærliggende å anta at deteksjoner som ikke er i serier på 30 ikke er kompresjoner. Som nevnt innledningsvis1 er det dog store variasjoner i hvordan kompresjoner gis. Det er derfor nødvendig å vise moderasjon med tanke på hvor mange deteksjoner som må være i serie før de antas å være kompresjoner. Her velger vi å fjerne deteksjoner hvis det bare er to av dem i en serie, hvis en kan kalle to deteksjoner en serie. Det blir kjørt tester som finner avstandene mellom fire nabotopper:



Figur 2.13: Illustrasjon av avstandene d_1, d_2 og d_3 .

Når disse avstandene er funnet, blir de kjørt gjennom en logisk test.

Hvis $d_1 > D_{lav}$ & $d_2 < D_{høy}$ & $d_3 > D_{lav} \Rightarrow$ Topp 2 og 3 fjernes. Hvis ikke dette kriteriet var oppfylt ble topp 2 og 3 beholdt.

- $|t_i - t_{i+1}| > D_{lav}$ & $|t_{i+1} - t_{i+2}| < D_{høy}$ & $|t_{i+2} - t_{i+3}| > D_{lav} \Rightarrow t_{i+1}$ & t_{i+2} fjernes fra datasettet med topppunkter.
- ellers beholdes toppunktet t_i i datasettet med topppunkter

der t_i er plasseringen i tid, samplenummeret, til topp i , der $i = 1, 2, 3 \dots n-2$ og n er antall topper i impedansdataene.

3 Resultater

I dette kapittelet presenteres ytelsen til de forskjellige metodene.

Tallet før parantesen er, som kolonneoverskriften i tabellen tilsier, P_D eller P_F .

Tallene i parantesene er nedre og øvre kvartil til P_F -ene og P_D -ene, altså: P_F (nedre kvartil - øvre kvartil) og P_D (nedre kvartil - øvre kvartil).

Nedre kvartil representerer en verdi som er slik at 25% av målingene har en verdi lavere enn den. Øvre kvartil er en verdi som er slik at 75% av målingene har en verdi lavere enn den.[24]

Metode	Treningsdata		Testdata	
	P_D	P_F	P_D	P_F
1	0,9883(0,97-0,99)	0,2680(0,16-0,42)	0,9708(0,97-1,00)	0,2086(0,06-0,18)
2	0,9860(0,96-0,99)	0,0516(0,02-0,13)	0,9608(0,96-1,00)	0,1513(0,02-0,12)
3	0,9728(0,93-0,99)	0,0250(0,01-0,06)	0,9385(0,93-1,00)	0,0827(0,02-0,05)
4	0,9728(0,93-0,99)	0,0219(0,01-0,06)	0,9326(0,91-1,00)	0,0628(0,01-0,04)

Tabell 4: Oversikt over resultatene de forskjellige metodene oppnådde da de ble kjørt på trenings- og testdataene.

Treningsdata er de dataene som ble brukt under utvikling og testing av de forskjellige metodene. Testdata ble brukt til å verifisere resultatene en oppnår på treningsdatane. Hvis en får veldig gode resultater for treningsdataene og heller dårlige for testdataene tyder dette på overtrening, som vil si at metodene er for tilpasset dataene i treningsdataene og må gjøres mer generelle for å kunne fungere på andre vilkårlige datasett.

4 Diskusjon

4.1 Metode 1: Toppdetektering med terskling

Denne metoden tar ikke annet hensyn til støy enn at de toppene med aller minst amplitude ikke blir tatt med. Dette fører at det blir en del falske deteksjoner. Det ser en på P_F for denne metoden i Tabell 4. P_F er her 0,2680 og 0,2086 for henholdsvis trenings- og testdataene, som er nokså høye verdier. Samtidig har vi også høye verdier for P_D , 0,9883 og 0,9708 for henholdsvis trenings- og testdataene. Dette tyder på at toppdetektering har potensiale som en metode for å detektere kompresjoner. Dog må sannsynligheten for falske deteksjoner reduseres.

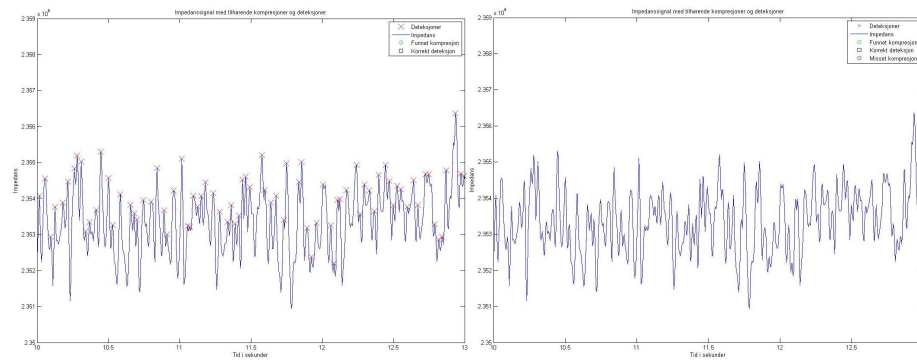
To av målingene⁹ inneholder en del støy som metoden ikke klarer å håndtere brukbart. For de to datasettene oppnås en P_F på 0,6285 og 0,9389, som en ser i Tabell 5 på side 33 og Tabell 6 på side 33. Altså har vi svært høy sannsynlighet for å gjøre falske deteksjoner i disse datasettene.

4.2 Metode 2: Støyfjerning

Den delen av denne metoden, som beskrevet i 2.8.2, som fjernet den høyfrekvente støyen ble i utgangspunktet laget for å fjerne den hvite støyen, som blir nevnt i 2.7.1. Deteksjonene gjort i områdene med hvit støy ble effektivt fjernet, se Figur 4.1.

Områdene i datasettene som inneholdt den hvite støyen ble dog fjernet fra datasettene da det var umulig å gjøre detektering i de rammede områdene, som nevnt i 2.7.1. Metoden viste seg likevel å ha en positiv effekt på P_F , og ble derfor tatt med i oppgaven. P_F går fra 0,2680, for metode 1, til 0,0516, for metode 2, for treningsdataene. Det så altså ut til at denne metoden hadde potensiale som en forbedring av metode 1. Resultatene for testdataene var derimot ikke like bra, men det var likevel en forbedring i forhold til for metode 1. P_F gikk fra 0,2086 til 0,1513.

⁹Datasettene det er snakk om er 10-Nov-2010_002.mat i Tabell 5 på side 33 og 10-Nov-2010_0024.mat i Tabell 6 på side 33.



Figur 4.1: Utsnitt av hvis støy før og etter Metode 2 er kjørt. Metode 2 har altså ingen deteksjoner i området der Metode 1 har godkjent nesten alle toppene. Vi unngår dermed mange feildeteksjoner.

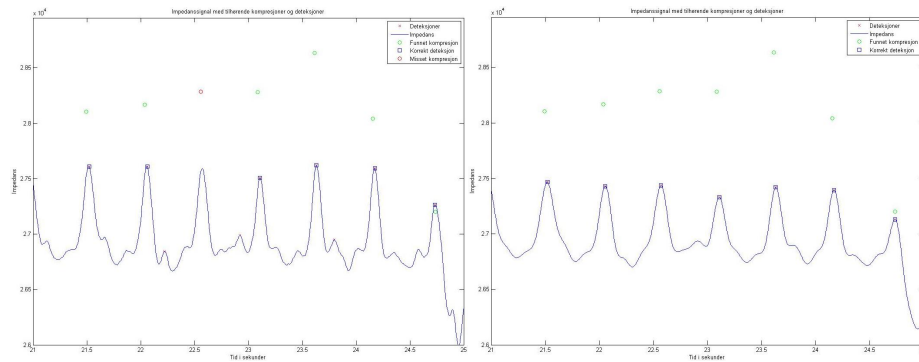
Metode 2 har altså ingen deteksjoner i området der Metode 1 har godkjent nesten alle toppene. Vi unngår dermed mange feildeteksjoner.

For metode 1 hadde vi altså veldig dårlige resultater for to av målingene, de fikk en P_F på 0,6285 og 0,9389, som nevnt i 4.2. For disse to målingene fikk vi her, med metode 2, en P_F på 0,2287 og 0,9214. Så denne metoden fjernet stort sett problemet med den ene målingen, men det er fortsatt en måling igjen med alt for dårlige resultater.

4.3 Metode 3: Glatting

Ved bruk av glatting vil en som regel ende opp med å “glatte bort” informasjon en i utgangspunktet helst ville ha beholdt. Det ser vi igjen i en redusert P_D . P_D synker fra 0,9608, fra metode 2, til 0,9385 for testdataene.

For denne metoden oppnår vi nesten en halvering av P_F , fra 0,1513 til 0,0827, i forhold til Metode 2 for testdataene.



Figur 4.2: Glatting

I figuren over, Figur 4.2, ser vi i bildet til venstre at den tredje toppen fra venstre ikke er detektert. Dette er fordi det er mye støy rundt denne toppen, deteksjonsalgoritmen har dermed ansett dette området som et område med bare støy og ekskludert alle deteksjoner i det. I bildet til høyre er en del av støyen glattet bort, og alle kompresjonene er korrekt detektert.

Heller ikke denne metoden fikk redusert P_F til den støybefengte målingen¹⁰ som nevnes i 4.1 og 4.2. Den ble noe redusert, men er fortsatt alt for høy. P_F ble redusert fra 0,9214 til 0,8570.

4.4 Metode 4: Fjerning av deteksjoner basert på gruppering og avstand

Den første delen av denne metoden ser, som nevnt i 2.8.4, kun på første og siste deteksjon i en impedansmåling. En svakhet ved denne er at den vil fjerne en korrekt deteksjon dersom denne er plassert alene, først eller sist i målingen. Totalt sett hadde denne en minimal positiv effekt på vurderingsparameteren P_F . Parameteren P_D forble uforandret. Forandringene var så små at det ikke ble sett på som hensiktsmessig å ta dette med som en egen metode.

Da den andre delen ble implementert, ble det derimot litt større utslag i P_D og P_F . Denne delen er, som beskrevet i 2.8.4, laget for å fjerne støy bestående av toppe i grupper som er så små at det er rimelig å anta at det ikke er snakk om kompresjoner. For treningsdataene forble P_D uforandret, 0,9728. P_F ble

¹⁰10-Nov-2010_0024.mat i tabellene i A

reduisert litt og gikk fra 0,0250, for metode 3, til 0,0219, for metode 4. Dette er ingen stor forbedring, men fortsatt en forbedring.

For testdataene ble P_D påvirket, den sank fra 0,9385, for metode 3, til 0,9326, for metode 4.

P_F ble for testdataene faktisk forbedret mer enn det som ble oppnådd for treningsdataene. Den sank fra 0,0827, for metode 3, til 0,0628, for metode 4.

Totalt sett har denne metoden bidratt til forbedringer, P_D er påvirket i svært liten, dog negativ, grad, og P_F har fått en betydelig reduksjon.

Heller ikke denne metoden hadde noen stor påvirkning på de dårligere resultatene som de andre metodene oppnådde for den ene målingen¹¹, den ble her redusert til 0,8089.

¹¹10-Nov-2010_0024.mat i tabellene i A

5 Konklusjon

Formålet med denne oppgaven var, som tittelen henspeiler på, å finne brystkompresjoner i hjertestansdata. Det ble utviklet fire metoder, som riktignok bygger på hverandre, for dette formålet. Metodene som er brukt i dette arbeidet er forholdsvis enkle. Det er dog vanskelig å si noe nøyaktig hvor gode metodene er, da det ikke er annet arbeid en kan sammenligne resultatene med. Arbeid med lignende problemstillinger ligger trolig bak utviklingen av en del av produktene nevnt i innledningen. Dette arbeidet er dog som bedriftshemmeligheter å regne og resultater derfra er ikke mulige å oppdrive.

På det beste oppnådde altså den siste metoden en P_D på 0,9326. Det vil si at den finner flesteparten av kompresjonene i målingene. Samtidig ble det oppnådd en P_F på 0,0628. Noe som en antar er relativt bra. En slik metode er kanskje likevel ikke robust nok til sitt formål, da den ikke klarte å håndtere en av målingene¹² som inneholdt spesielt mye støy.

I videre arbeid vil det derfor være nødvendig å finne en mer robust metode for deteksjon av kompresjoner. Det kan også være interessant å finne ut hvor vanlig det er med så mye støy som er tilstede i ovenfornevnte måling.

Hvis denne metoden likevel skulle være robust nok, kan den brukes til å finne frekvensen på kompresjonene som gis. De deteksjonene som blir funnet blir lagret med plassering i tid i signalet, dermed kan en finne avstanden mellom dem og bruke den til å beregne frekvensen. Frekvensen er en av tingene som må overvåkes for å kunne gi tilbakemelding på om HLR er i tråd med gjeldende retningslinjer.

¹²Målingen det er snakk om er 10-Nov-2010_0024.mat, som en kan se i Tabell 6 på side 33, Tabell 8, Tabell 10 og Tabell 12.

Figurer

2.1	Oppsett for impedansmåling[20]	4
2.2	Plottede impedansdata	5
2.3	Annotering av kompresjoner	7
2.4	Illustrasjon av godkjenningsintervallet rundt en annotert kompresjon. Så lenge en deteksjon ligger mellom de røde linjene er den nær nok den annoterte kompresjonen til å kunne godkjennes.	10
2.5	Impedanssignal med tilhørende kompresjoner og deteksjoner. Sirkelene under impedanssignalet angir plasseringen til annoterte kompresjonene fra kraftsignalet.	10
2.6	Støy og feilaktig godkjent deteksjon	12
2.7	forskjøvet data	13
2.8	Kompresjoner og deteksjoner	15
2.9	Illustrasjon av avstandene d_1 , d_2 , d_3 og d_4	17
2.10	Illustrasjon av avstandene d_1 og d_2	18
2.11	Eksempel på glatting	20
2.12	Illustrasjon av avstanden d_1	21
2.13	Illustrasjon av avstandene d_1, d_2 og d_3	22
4.1	Utsnitt av hvis støy før og etter Metode 2 er kjørt. Metode 2 har altså ingen deteksjoner i området der Metode 1 har godkjent nesten alle toppene. Vi unngår dermed mange feildeteksjoner.	26
4.2	Glatting	27

Referanser

- [1] M.Risdal, H.Losert, J.Nysæther, S.O. Aase, F.Sterz, and T.Eftestøl. Non-invasive ventilation monitoring during cardiopulmonary resuscitation by means of thoracic impedance.
- [2] H.Losert, M.Risdal, F.Sterz, J.Nysæther, K.Köhler, T.Eftestøl, C.Wandaller, H.Myklebust, T.Uray, and A.N. Laggner, Thoracic impedance changes measured via defibrillator pads can monitor ventilation in critical ill patients and during cardiopulmonary resuscitation, *Critical Care Medicine*, vol.34, no.9, pp.2399–2405, 2006.
- [3] Predrag Stevanović, Radislav Šćepanović, Dragan Radovanović, Đorđe Bajec, Radoslav Perunović, Dragoš Stojanović and Dejan Stevanović. Thoracic electrical bioimpedance theory and clinical possibilities in perioperative medicine, *SIGNA VITAE* 2008; 3 Suppl 1: S 22 - 27.
- [4] M.Risdal, Impedance plethysmography for ventilation and pulse monitoring during cardiopulmonary resuscitation.
- [5] 1. Benjamin S. Abella, MD, MPhil; 2. Jason P. Alvarado, BA; 3. Helge Myklebust, BEng; 4. Dana P. Edelson, MD; 5. Anne Barry, RN, MBA; 6. Nicholas O’Hearn, RN, MSN; 7. Terry L. Vanden Hoek, MD; 8. Lance B. Becker, MD, Quality of Cardiopulmonary Resuscitation During In-Hospital Cardiac Arrest, *JAMA*. 2005;293(3):305-310.
- [6] 1. Lars Wik, MD, PhD; 2. Jo Kramer-Johansen, MD; 3. Helge Myklebust, BEng; 4. Hallstein Sørebo, MD; 5. Leif Svensson, MD; 6. Bob Fellows, MD; 7. Petter Andreas Steen, MD, PhD, Quality of Cardiopulmonary Resuscitation During Out-of-Hospital Cardiac Arrest, *JAMA*. 2005;293(3):299-304.
- [7] Wik L, Steen PA, Bircher NG, Quality of bystander cardiopulmonary resuscitation influences outcome after prehospital cardiac arrest.
- [8] Van Hoeyweghen RJ, Bossaert LL, Mullie A, Calle P, Martens P, Buylaert WA, Delooz H, Quality and efficiency of bystander CPR. Belgian Cerebral Resuscitation Study Group.
- [9] 1. E. John Gallagher, MD; 2. Gary Lombardi, MD; 3. Paul Gennis, MD, Effectiveness of Bystander Cardiopulmonary Resuscitation and Survival Following Out-of-Hospital Cardiac Arrest, *JAMA*. 1995;274(24):1922-1925.

- [10] M.Risdal, M.Stavland, S.O. Aase, and T.Eftestøl, Impedance-based ventilation detection during cardiopulmonary resuscitation using a neural network, in Proceedings of 3rd European Medical & Biological Engineering Conference, 2005.
- [11] European Resuscitation Council, European Resuscitation Council Guidelines for Resuscitation 2005.
- [12] Sörnmo Leif, Laguna Pablo, Bioelectrical Signal Processing in Cardiac and Neurological Applications, Elsevier Academic Press, 2005.
- [13] <http://www.billauer.co.il/peakdet.html>
- [14] <http://www.pocketcpr.com/iphone.html>
- [15] <http://www.healthcare.philips.com/in/products/resuscitation/products/mrx/qcpr.wpd>
- [16] J.Ruiz, U.Irusta, S.Ruiz de Gauna, and T.Eftestøl, Cardiopulmonary resuscitation artefact suppression using a Kalman filter and the frequency of chest compressions as the reference signal , Resuscitation, 2010, doi:10.1016/j.resuscitation.2010.02.031.
- [17] Jon Richardsen, Ambulansemedisin 2 Medisinske oppdrag, Gyldendal Norsk Forlag AS 2008.
- [18] http://en.wikipedia.org/wiki/Cardiopulmonary_resuscitation
- [19] http://www.mathworks.com/help/toolbox/curvefit/bq_6yqb.html
- [20] http://en.wikipedia.org/wiki/File:Defibrillation_Electrode_Position.jpg
- [21] <http://www.forstehjelp.info/emner/hjertelungeredning.htm>
- [22] <http://www.americanheart.org/presenter.jhtml?identifier=4540>
- [23] <http://www.snl.no/defibrillering>
- [24] <http://www.kunnskapssenteret.com/articles/2634/1/Prosentiler-og-kvartiler/Prosentiler-og-kvartiler.html>

A Resultater for alle datasett

Årsaken til at datasettene her kan se ut til å ha litt tilfeldig nummerering, er at en den av dem ble fjernet i forbehandlingsprosessen, som beskrevet i 2.7.2 på side 13.

A.1 Metode 1

A.1.1 Treningsdata

For treningssettet med data oppnådde denne metoden følgende resultater:

Treningsdata:	P_D	P_F	P_M	N_F	N_M	N_D
10-Nov-2010__002.mat	0,9419	0,6285	0,0581	247	9	146
10-Nov-2010__003.mat	0,9915	0,1104	0,0085	72	5	580
10-Nov-2010__005.mat	0,9982	0,2116	0,0018	146	1	544
Totalt:	0,9883	0,2680	0,0117	465	15	1270

Tabell 5: Treningsdata Metode 1

A.1.2 Testdata

Testdata:	P_D	P_F	P_M	N_F	N_M	N_D
10-Nov-2010__006.mat	1,0000	0,1142	0,0000	129	0	1001
10-Nov-2010__009.mat	0,9687	0,0560	0,0313	22	12	371
10-Nov-2010__0010.mat	1,0000	0,0106	0,0000	20	0	1870
10-Nov-2010__0011.mat	1,0000	0,0893	0,0000	57	0	581
10-Nov-2010__0012.mat	0,9561	0,1915	0,0439	356	69	1503
10-Nov-2010__0013.mat	1,0000	0,0953	0,0000	55	0	522
10-Nov-2010__0016.mat	1,0000	0,1682	0,0000	56	0	277
10-Nov-2010__0019.mat	0,9580	0,1729	0,0420	262	55	1253
10-Nov-2010__0020.mat	0,9757	0,0592	0,0243	149	59	2368
10-Nov-2010__0022.mat	0,9817	0,0369	0,0183	80	39	2087
10-Nov-2010__0024.mat	1,0000	0,9389	0,0000	1937	0	126
10-Nov-2010__0025.mat	0,8835	0,2231	0,1165	353	162	1229
Totalt:	0,9708	0,2086	0,0292	3476	396	13188

Tabell 6: Testdata Metode 1

A.2 Metode 2

A.2.1 Treningsdata

Treningsdata:	P_D	P_F	P_M	N_F	N_M	N_D
10-Nov-2010_002.mat	0,9355	0,2287	0,0645	43	10	145
10-Nov-2010_003.mat	0,9897	0,0086	0,0103	5	6	579
10-Nov-2010_005.mat	0,9963	0,0372	0,0037	21	2	543
Totalt:	0,9860	0,0516	0,0140	69	18	1267

Tabell 7: Treningsdata Metode 2

A.2.2 Testdata

Testdata	P_D	P_F	P_M	N_F	N_M	N_D
10-Nov-2010_006.mat	0,9990	0,0225	0,0010	23	1	1000
10-Nov-2010_009.mat	0,9661	0,0186	0,0339	7	13	370
10-Nov-2010_0010.mat	1,0000	0,0048	0,0000	9	0	1870
10-Nov-2010_0011.mat	0,9983	0,0705	0,0017	44	1	580
10-Nov-2010_0012.mat	0,9141	0,1173	0,0859	191	135	1437
10-Nov-2010_0013.mat	1,0000	0,0679	0,0000	38	0	522
10-Nov-2010_0016.mat	0,9856	0,0808	0,0144	24	4	273
10-Nov-2010_0019.mat	0,9442	0,1418	0,0558	204	73	1235
10-Nov-2010_0020.mat	0,9712	0,0360	0,0288	88	70	2357
10-Nov-2010_0022.mat	0,9802	0,0253	0,0198	54	42	2084
10-Nov-2010_0024.mat	0,9762	0,9214	0,0238	1442	3	123
10-Nov-2010_0025.mat	0,8627	0,1447	0,1373	203	191	1200
Totalt:	0,9608	0,1513	0,0392	2327	533	13051

Table 8: Testdata Metode 2

A.3 Metode 3

A.3.1 Treningsdata

Treningsdata:	P_D	P_F	P_M	N_F	N_M	N_D
10-Nov-2010_002.mat	0,8710	0,1060	0,1290	16	20	135
10-Nov-2010_003.mat	0,9795	0,0052	0,0205	3	12	573
10-Nov-2010_005.mat	0,9945	0,0234	0,0055	13	3	542
Totalt:	0,9728	0,0250	0,0272	32	35	1250

Tabell 9: Treningsdata Metode 3

A.3.2 Testdata

Testdata	P_D	P_F	P_M	N_F	N_M	N_D
10-Nov-2010_006.mat	0,9980	0,0070	0,0020	7	2	999
10-Nov-2010_009.mat	0,9321	0,0138	0,0679	5	26	357
10-Nov-2010_0010.mat	1,0000	0,0016	0,0000	3	0	1870
10-Nov-2010_0011.mat	0,9983	0,0203	0,0017	12	1	580
10-Nov-2010_0012.mat	0,8632	0,0444	0,1368	63	215	1357
10-Nov-2010_0013.mat	1,0000	0,0315	0,0000	17	0	522
10-Nov-2010_0016.mat	1,0000	0,0246	0,0000	7	0	277
10-Nov-2010_0019.mat	0,9625	0,0837	0,0375	115	49	1259
10-Nov-2010_0020.mat	0,9209	0,0223	0,0791	51	192	2235
10-Nov-2010_0022.mat	0,9793	0,0179	0,0207	38	44	2082
10-Nov-2010_0024.mat	0,9921	0,8570	0,0079	749	1	125
10-Nov-2010_0025.mat	0,7803	0,0701	0,2197	82	306	1087
Totalt:	0,9385	0,0827	0,0615	1149	836	12750

Tabell 10: Testdata Metode 3

A.4 Metode 4

A.4.1 Treningsdata

Treningsdata:	P_D	P_F	P_M	N_F	N_M	N_D
10-Nov-2010_002.mat	0,8710	0,1000	0,1290	15	20	135
10-Nov-2010_003.mat	0,9795	0,0017	0,0205	1	12	573
10-Nov-2010_005.mat	0,9945	0,0217	0,0055	12	3	542
Totalt:	0,9728	0,0219	0,0272	28	35	1250

Tabell 11: Treningsdata Metode 4

A.4.2 Testdata

Testdata	P_D	P_F	P_M	N_F	N_M	N_D
10-Nov-2010_006.mat	0,9970	0,0020	0,0030	2	3	998
10-Nov-2010_009.mat	0,9164	0,0057	0,0836	2	32	351
10-Nov-2010_0010.mat	1,0000	0,0016	0,0000	3	0	1870
10-Nov-2010_0011.mat	0,9983	0,0136	0,0017	8	1	580
10-Nov-2010_0012.mat	0,8601	0,0357	0,1399	50	220	1352
10-Nov-2010_0013.mat	0,9962	0,0280	0,0038	15	2	520
10-Nov-2010_0016.mat	1,0000	0,0177	0,0000	5	0	277
10-Nov-2010_0019.mat	0,9625	0,0749	0,0375	102	49	1259
10-Nov-2010_0020.mat	0,9056	0,0157	0,0944	35	229	2198
10-Nov-2010_0022.mat	0,9788	0,0156	0,0212	33	45	2081
10-Nov-2010_0024.mat	0,9921	0,8089	0,0079	529	1	125
10-Nov-2010_0025.mat	0,7602	0,0578	0,2398	65	334	1059
Totalt:	0,9326	0,0628	0,0674	849	916	12670

Tabell 12: Testdata Metode 4

B Matlabkode

I denne delen forklares matlabkoden som er brukt helt generelt.

For alle metodene er det laget en matlabfunksjon som inneholder selve metoden, kalt FunksjonFinnKorrekt2, den finner topppunkter, ved kalling av funksjonen peakdet.m, og avstanden mellom dem o.s.v. I disse funksjonene er også koden som sjekker eventuelle deteksjoner opp mot annoterte kompresjoner i fasiten, som ble laget med Fasit.m. Denne funksjonen finner N_F , N_M og N_D , det er disse en finner i A, for dataene som den mates med, og returnerer dem til programmet som funksjonen kalles fra.

Hovedprogrammet, kalt apnefiler2.m, som funksjonen kalles fra sender alle målingene, en om gangen, til den aktuelle funksjonen. I retur får den som nevnt N_F , N_M og N_D fra hvert datasett. Disse legger programmet sammen og regner ut de totale verdiene for alle datasettene, det er disse verdiene en finner i Tabell 4. Dette programmet er likt for alle de forskjellige metodene.

For å enklere kunne få oversikt over hva som faktisk ble gjort for hvert datasett, ble det laget et program, kalt Finnkorrekt4.m, som kjørte toppdetektering, sammenlignet resultatene med fasiten og plottet alle deteksjoner, missede deteksjoner osv for enkeltmålinger. Det er dette programmet Figur 2.7,2.8, 4.1 og 4.2 er laget i.

All kode som nevnes overnfor ligger på vedlagt CD i mappen Matlabkode. I denne mappen er det en undermappe for metode. I disse fire undermappene ligger koden brukt for hver av metodene. Det er også en mappe kalt fasitfiler, der finner vi koden for å annotere kompresjoner og alle de annoterte kompresjonene som .mat filer.