

Automatisk sporing av Dopplerspektrum

Gaute Vartdal

Master i kommunikasjonsteknologi
Oppgaven levert: Juni 2011
Hovedveileder: Ilangko Balasingham, IET
Biveileder(e): Torbjørn Hergum, ISB

Problembeskrivelse

Dopplerultral lyd er et viktig verktøy for å evaluere hastigheten til blod ved gitte steder i hjertet. I noen applikasjoner er det ønskelig å spore konturen av spektrumet automatisk. Det er for eksempel kjent at dP/dt og relaksasjonskonstanten i venstre ventrikkel kan finnes fra Dopplerspektrum av mitralinsuffisienser. I denne sammenhengen er det nødvendig å måle stigningen til konturen av spektrumet fra tidlig og sen systole. Denne masteroppgaven skal foreslå en automatisk sporing algoritme for generelle Dopplerspektrum og spesifikt for Dopplerspektrum av mitralinsuffisienser.

Sammendrag

En rekke verdier i hjertet som man normalt finner ved invasiv undersøkelse med kateter, kan med stor nøyaktighet beregnes ut fra hastigheter man finner ved Dopplerultral lyd. Eksempelvis er trykket i hjertekammere og blodårer viktig informasjon i forbindelse med undersøkelser av hjertefunksjonen til en pasient. Ved å studere konturene av hastigheten til blodet ved gitte punkt i hjertet kan man med bruk av Bernoulli's forenklete ligning beregne disse verdiene uten å penetrere pasienten med kateter. Maksimal positiv og negativ trykkøkning (dP/dt) i ventrikkelen er eksempler på verdier man kan finne, og er to av de mest utbredte indikatorene på ventrikkelfunksjonen. Disse kan beregnes ut fra hastigheten til lekkasjer fra ventrikkelen til atrium, også kalt mitralinsuffisienser. Muligheten til å måle disse, samt andre verdier med Dopplerultral lyd, har en enorm fordel fremfor bruk av kateter.

Som oftest må sporingene av et Dopplerspektrum gjøres manuelt, en prosess som er tidkrevende og vanskelig. Denne masteroppgaven foreslår en algoritme som automatisk kan spore konturene av Dopplermålinger. Algoritmen er tilpasset Dopplerspektrum av mitralinsuffisienser, men fungerer generelt for alle typer spektrum. Algoritmen gjør også et forsøk på å håndtere delvis svake/manglende kanter i spektrumet.

Resultatene er sammenlignet med manuelt sporede kanter, og viser at algoritmen med stor nøyaktighet kan beregne verdier som dP/dt og maksimal hastighet. Maksimal og minimal dP/dt kan beregnes med en gjennomsnittlig forskjell fra den manuelle sporingen på under 100 mmHg/s, og toppunktet med forskjell under 0.05 m/s. Resultatene viser at så lenge kvaliteten på Dopplermålingene er akseptabel, sporer algoritmen konturen nøyaktig, og fjerner effektivt støy og artifakter langs konturen. Forskjellen mellom automatisk og manuelt ervervet maksimal dP/dt har et standardavvik så lavt som 79 mmHg/s når spektrumene har god kvalitet.

Også i spektrum hvor deler av signalet er svakt kan verdier som dP/dt predikeres av algoritmen, og når mindre enn 60% av et spektrum må predikeres kan maksimal og minimal dP/dt finnes veldig nøyaktig.

Forord

Denne rapporten er et resultat av en masteroppgave ved Institutt for elektronikk og telekommunikasjon ved Norges teknisk-naturvitenskapelige universitet. Oppgaven er utarbeidet i samarbeid med Dopplergruppen ved Institutt for sirkulasjon og bildediagnostikk. Oppgaven er utført under veiledning av professor Hans Torp og postdoc Torbjørn Hergum.

Jeg vil takke mine veiledere Torbjørn Hergum og Hans Torp for gode råd, innspill og veiledning under arbeidet. Jeg vil også rette en takk til øvrige medlemmer av Dopplergruppa for praktisk hjelp i forbindelse med programvare.

Jeg vil til slutt få takke Trond Vartdal (spesialist i indremedisin, Ph.D-student i hjertemedisin og storebror) for innspill og hjelp med å spore Dopplerspektrum manuelt.

Trondheim, 13. juni 2011

A handwritten signature in black ink, appearing to read 'Gaute Vartdal'. The signature is fluid and cursive, with the first name 'Gaute' written in a larger, more prominent script than the last name 'Vartdal'.

Gaute Vartdal

Innhold

Problembeskrivelse	i
Forord	v
1 Innledning	1
1.1 Problemstilling	1
1.2 Målsetting	2
1.3 Tidligere forskning	3
2 Teori	5
2.1 Anatomi	6
2.1.1 Hjertesyklusen	6
2.1.2 Kontraksjoner og elektriske impulser	7
2.1.3 Oversikt over hjertesyklusen	8
2.2 Ultralyd/ekkokardiografi	9
2.2.1 Dopplerultralyd	10
2.2.2 Kontinuerlig og pulset Doppler	11
2.3 Bernoulli's prinsipp	12
2.4 Ekkokardiografi som klinisk verktøy	13
2.5 Verktøy	14
2.5.1 DICOM	14
2.5.2 Matlab	14
2.5.3 GcLab	14
3 Metode	15
3.1 Datamateriale	16
3.2 Valg av metode	17
3.3 Kvalitet på spektraldata	18
3.4 Preprossesering av Dopplerspekteret	19
3.4.1 Filtrering	20
3.4.2 <i>snake</i>	23
3.5 Etterbehandling av sporingen	25
3.5.1 Analyse	26
3.5.2 Interpolering og ekstrapolering	28

3.5.3	Modelltilpasning/predikering	29
3.5.4	Utjevning	32
3.5.5	Utrekninger	33
3.6	Oversikt over algoritmen	34
4	Resultater	37
4.1	Kantsporing	38
4.2	Ferdig behandlede Dopplersporinger	40
4.2.1	Sporing av spektrum av dårlig kvalitet	41
4.2.2	Bruk av modeller	42
4.3	Sammenligninger	45
4.3.1	Andre metoder	45
4.3.2	Sammenligning med manuelle sporinger	46
4.4	Øvrige resultat	49
4.4.1	Variasjon i utregningene	49
4.4.2	Tidsbruk	49
5	Diskusjon	51
5.1	Resultater	51
5.2	Videre arbeid	53
6	Konklusjon	55
	Programkode	57
	Bilder	59

Figurer

2.1	Hjertet	6
2.2	Elektrokardiogram (EKG)	7
2.3	Hjertesyklusen	8
2.4	Ultralyd av hjertet	9
2.5	Dopplereffekten	10
2.6	Eksempler på Dopplerspekter	11
2.7	Skjerm bilde av GcLab	14
3.1	Eksempel på spektraldata	16
3.2	Eksempler på dårlige Dopplermålinger	18
3.3	Eksempel på spektraldata før og etter preprosessering	20
3.4	Tre eksempler på histogram til en blokk i bildet	20
3.5	Spektraldata etter median og gaussisk filtrering	21
3.6	Utsnitt av figur 3.5	21
3.7	Binærbilde og bildet med kun ytterkantene	21
3.8	Kantdeteksjon	22
3.9	Sporet kant før og etter metoden <i>snake</i>	23
3.10	Kantdeteksjon med ujevn kant	25
3.11	Dopplerspektrum med tydelig klaffehastighet	25
3.12	Diverse bilder som krever predikering	26
3.13	Viktige punkt langs den sporede kanten	27
3.14	To bilder som krever predikering	29
3.15	Interpolasjon	29
3.16	Interpolasjon	30
3.17	Bilde av kanten delt opp i tre segment	30
3.18	Kant som ikke er utjevnet	32
3.19	Ikke utjevnet trykkgradientskurve og dens differensierte	32
3.20	Utjevnet trykkgradientskurve og dens differensierte	33
3.21	Oversikt over algoritmen	34
4.1	Eksempler på kantsporing av klare og tydelige spektre	38
4.2	Eksempler på kantsporing av Dopplerspektrum av dårlig kvalitet	39
4.3	Ferdig behandlede sporinger	40
4.4	Sporinger av svake spektrum	41

4.5	Sporingsalgoritmen med bruk av modeller	42
4.6	Sporet kant hvor toppen er fjernet	43
4.7	Gjennomsnittlig forskjell mellom verdier utregnet fra predikerte og ikke predikerte spektrum	43
4.8	44
4.9	Sporing med og uten 60% fjernede kanter samt den predikerte	44
4.10	Sporing med og uten 80% fjernede kanter samt den predikerte	45
4.11	Sammenligning mellom algoritmer. Eksempel 1	45
4.12	Sammenligning mellom algoritmer. Eksempel 2	46
4.13	Utvalg av bilder som er blitt sammenlignet	46
4.14	Sammenligning 1	47
4.15	Sammenligning 2	47
4.16	Sammenligning 3	47
4.17	Sammenligning 4	48
4.18	Sammenligning 5	48

Forkortelser

A-V	Atrio-ventrikulær
CW	Continous-wave doppler
DICOM	Digital Imaging and Communications in Medicine
EKG	Elektrokardiogram
IVRT	Isovolumetrisk relaksasjonstid
PW	Pulsed-wave doppler
QRS	Karakteristisk del av en EKG

Kapittel 1

Innledning

1.1 Problemstilling

Ved bruk av Dopplerultral lyd er det mulig å måle en rekke verdier i hjertet som alternativt kun kan måles ved en invasiv undersøkelse. Dopplerultral lyd kan måle hastigheten til blod og vev i bevegelse ved hjelp av ultralydbølger. Disse hastighetene kan videre benyttes for å kvantifisere flere verdier ved hjertefunksjonen.

Pasienter med en mitralinsuffisiens vil ha en lekkasje i mitralklaffen som fører til at blod strømmer tilbake til venstre atrium i det venstre ventrikkel trekker seg sammen. Med Dopplerultral lyd kan vi måle og visualisere hastigheten til blodstrømmen, og på grunn av relasjonen mellom trykk og hastighet gitt av Bernoulli's likning, kan trykk og trykkgradienter over mitralklaffen kvantifiseres. Kunnskap om disse verdiene kan fortelle mye om ventrikkelens kontraktilitetsevne. Slike verdier finner man normalt ved å måle trykket direkte inne i ventrikkelen, men allerede i 1976 ble nøyaktigheten ved å beregne disse verdiene direkte fra Dopplermålinger verifisert av Holen et.al.[1].

Uten dedikert programvare må kanten av Dopplerspekteret spores manuelt for å gjøre utregningene. Dette er både tids- og ressurskrevende, og krever at sporingen gjøres veldig nøyaktig, spesielt om man ønsker å regne ut den instantane hastighetsøkningen som er veldig følsom for punkt-til-punkt-variasjonene langs kanten av spekteret.

Det er tidligere foreslått noen algoritmer for å automatisk spore langs kanten av Dopplerspekter, med varierende resultat. Per i dag mangler likevel slik funksjonalitet ofte på utstyr og i applikasjoner for behandling av ultralydbilder. Det er derfor ønskelig å utvikle en algoritme som effektivt kan spore kanten av Dopplerspektrum. Ettersom det finnes forskjellige typer Dopplerspektrum, både i form av målemetode og i kvalitet, må algoritmen være generell og robust.

1.2 Målsetting

Da en algoritme som automatisk kan spore dopplerspektrum og regne ut ønskede verdier i høy grad vil effektivisere analyse av Dopplermålinger, er dette en attraktiv funksjonalitet som er ønskelig å implementere. GcLab er et Matlab-basert forskningsverktøy som blant annet brukes for å studere Dopplerspektrum, men muligheter for å spore spektrumene automatisk er ikke tilgjengelig. En slik mulighet er også ønskelig å ha på ultralydutstyr som brukes til diagnostisering av pasienter.

Dopplerspektrumet viser som nevnt hastigheten til for eksempel blod mellom hjertekammerene, og det finnes to metoder for å erverve den, kontinuerlig og pulset Doppler. Et Dopplerspektrum kan representere forskjellige blodstrømmer, for eksempel fra ventrikkel til atrium eller fra atrium til aorta. Samtidig vil kvaliteten på spektrumet variere avhengig av signal/støy-forholdet og hvordan målingene er utført.

En algoritme som skal spore spektrumet må derfor ta hensyn til dette, og på en generell basis fungere både for kontinuerlig og pulset Doppler. Målsettingen er derfor å utvikle en algoritme som er spesielt tilpasset Dopplerspektrum av mitralinsuffisienser, men som skal fungere for alle typer Dopplerspektrum. Samtidig skal algoritmen være robust i situasjoner hvor deler av spektrumet er mangelfullt. Det endelige resultatet skal sammenlignes med manuelt sporede kanter utført av en spesialist. Målsettingen kan settes opp i en liste:

- Sette seg inn i teori rundt Dopplerultralyd og hjertet
- Sette seg inn i tidligere forskning
- Utvikle en generell springsalgoritme som er spesielt tilpasset mitralinsuffisienser
- Foreslå en form for predikering for spektrum med delvis svake/manglende kanter
- Sammenligne automatisk sporede kanter med manuelt sporede kanter, og sammenligne forskjellen mellom verdier som er utregnet fra dem
- Resultatet skal være en fungerende algoritme som er implementert i GcLab

1.3 Tidligere forskning

Det finnes tidligere arbeid som har tatt for seg samme problemstilling [2, 3, 4, 5]. I forhold til den foreslåtte metoden har arbeidet til Juerg Tschirren et.al. [4] og Hayit Greenspan et.al. [5] flere likheter, slik som filtrering og terskelsetting. Metodene i [2] og [3] er på den andre siden basert på en modelltilpasning basert på kjennskap til formen på strukturene i spektraldataen.

Alle disse algoritmene har store forskjeller seg i mellom og i forhold til den foreslåtte algoritmen i denne oppgaven. Felles for disse algoritmene er at de er spesifikt rettet mot en type applikasjon. Metoden i [3] er for eksempel rettet mot PW Dopplerspektrum av mitralinsuffisienser og i [5] er metoden rettet mot Dopplerspektrum av atrieflimmer. Mange av disse artiklene tar tilsynelatende kun for seg klare og tydelige Dopplerspektrum, mens den foreslåtte metoden i denne oppgaven også prøver å tilby en form for predikering i Dopplerspektrum med svake konturer.

Metoden i [3] demonstrerer også en tilnærming til problemet hvor algoritmen er spesifikt tilpasset et sanntidssystem. Dopplerspektrumet blir prosessert en ramme/tidsenhet om gangen etterhvert som de leses inn.

Eksemplene både fra denne artikkelen og [5] viser at metodene i mange tilfeller har problemer med å detektere opp- og nedstigende kontur av spekteret. For utregning av dP/dt er det vitalt at disse segmentene finnes korrekt.

Kapittel 2

Teori

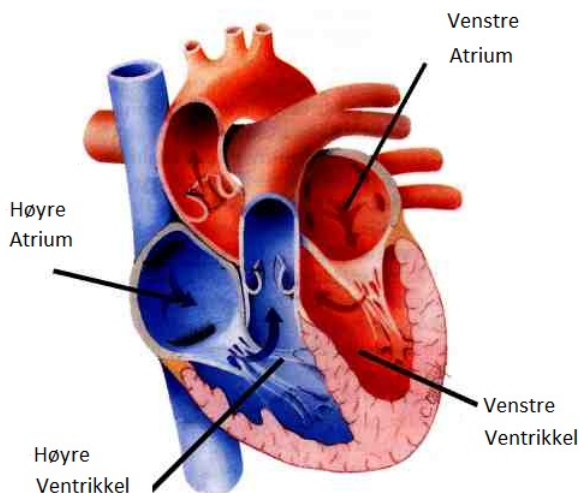
Teoridelen vil ta for seg teori som er nødvendig for å sette seg inn i stoffet rundt prosjektet, og annen relevant teori som vil gi leseren nok bakgrunnstoff til å lese videre.

Første del av teorien, kapittel 2.1, gir en kort beskrivelse av anatomi som er relatert til oppgaven. Deretter vil teori om ultralyd generelt, og Dopplerultralyd spesifikt, bli presentert i kapittel 2.2. Siste del av teorien tar for seg ekkokardiografi som klinisk verktøy, samt programvare og maskinvare som er benyttet.

All teori er grundig beskrevet og presentert sammen med figurer og formler.

2.1 Anatomi

Anatomidelen vil i enkelhet beskrive anatomen i de delene av hjertet som er relevante. Med mindre annet er nevnt er teori i anatomidelen hentet fra Szabo's "Biomedical signal and image processing"[6] og www.cvphysiology.com[7].



Figur 2.1: Hjertet

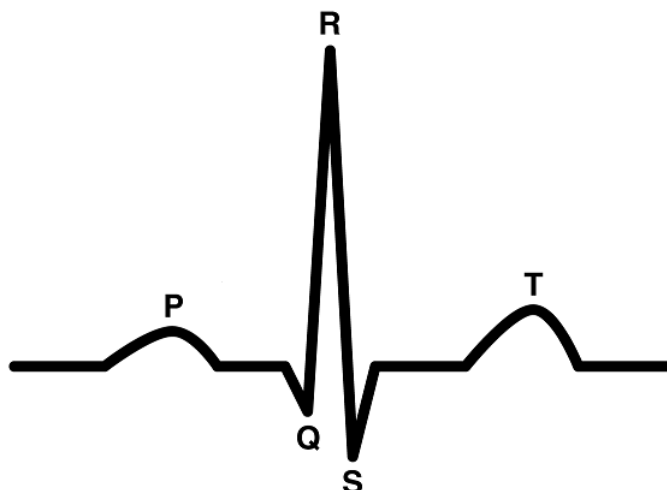
2.1.1 Hjertesykklusen

Figur 2.1 viser en enkel oversikt over hjertemuskel og dens 4 kammer. Hjertets oppgave er i all hovedsak å pumpe blod fra og til lungene og videre rundt i kroppen. Hjertet er en fantastisk muskel som periodisk trekker seg sammen for å pumpe blod mellom hjertekammerene og gjør dette kontinuerlig gjennom hele livet.

Hjertet består som nevnt av fire kammer, venstre og høyre atrium (forkammer) og ventrikke l(hovedkammer), som vist i figur 2.1. Alle kammer og blodårer er separert med klaffer. Klaffene skal i teorien kun åpne seg en vei, og skal forhindre at blodet strømmer tilbake. Via lungevenene kommer oksygenert blod fra lungene til venstre atrium. Etter en gitt fyllingstid vil det komme en kontraksjonsfase, hvor musklene rundt atriet kontraherer slik at blodet strømmer inn i venstre ventrikkel gjennom mitralklaffen. Fra ventrikkelen pumpes blodet videre ut i det store blodkretsløpet.

2.1.2 Kontraksjoner og elektriske impulser

Den delen av hjertesyklusen hvor hjertet trekker seg sammen kalles systolen, mens diastolen er neste del av syklusen hvor hjertemuskelene avslappes. Kontraksjonen er et resultat av at elektriske impulser forplanter seg i hjertet og disse kan måles med en EKG. På en EKG er det mulig å se når de enkelte fasene av syklusen inntreffer, da en slik måling registrerer utslag av elektriske impulser i hjertet. En vanlig EKG-måling er vist i figur 2.2.



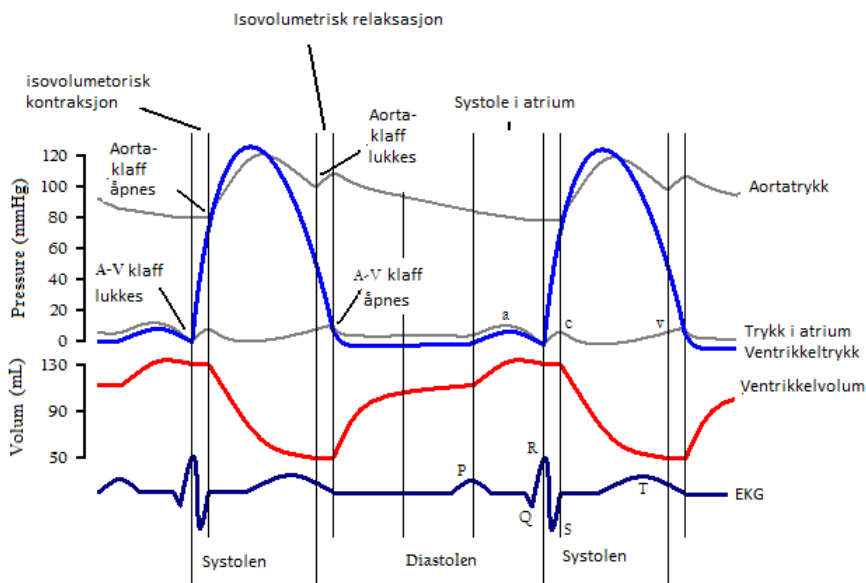
Figur 2.2: Elektrokardiogram (EKG)

P-bølgen viser sammentrekning av atrium. Både venstre og høyre atrium kontraherer. Kontraksjonen er et resultat av at muskelfibrene i sinus-knuten depolariseres fordi cellene utveksler ioner med omgivelsene. Cellene trekker seg sammen, og dette forplanter seg videre i hjertet til AV-noden, som ligger mellom atrium og ventrikkel, og videre via His-bunten til Purkinjefibrene. Disse elektriske impulsene gir utslag på en EKG, og QRS-komplekset man ser etter P-bølgen, er punktet hvor ventrikkelen har begynt å trekke seg sammen, den ventrikulære systole. QRS-komplekset er opphavet til hertepuls. T-bølgen er repolarisering av ventrikkelen. Repolariseringen av atrium er ikke synlig på EKG. I noen tilfeller har man også en U-bølge, som forekommer etter T-bølgen, som antas å komme av repolarisering av Purkinjefibrene.

Denne syklusen gjentar seg periodisk. Etter at ventrikkelen har pumpet blod inn i aorta og rundt i kretsløpet, fylles for- og hovedkammer seg opp igjen.

2.1.3 Oversikt over hjertesyklingen

Figur 2.3 viser hvordan trykket i atrium, ventrikkelen og aorta utvikler seg gjennom hjertesyklingen, og henger sammen med volum og EKG. Den lyseblå kurven viser hvordan trykket i ventrikkelen utvikler seg gjennom ventrikulær systole og diastole. I slutten av diastolen, etter at trykket i ventrikkelen har oversteget trykket i atrium lukker mitralklaffen (A-V-klaffen) seg. Systolen er startet, og man kan tydelig se hvordan trykket i ventrikkelen stiger til et maksimum, før trykket faller til et trykk lavere enn trykket i atrium, hvor mitralklaffen åpnes og ventrikkelen igjen fylles opp med blod.



Figur 2.3: Hjertesyklingen

En stund etter at ventrikulær systole inntreffer, i det trykket i ventrikkelen overstiger trykket i aorta, åpner aortaklaffen seg, hvorpå blodet strømmer inn i aorta. Tidspunktet når aortaklaffen lukker seg sammenfaller som regel med det tidspunktet hvor det høyeste trykkfallet (maksimum negativ dP/dt) i ventrikkelen er. Den første perioden av systolen går gjerne under navnet isovolumetriske kontraksjon, mens isovolumetriske relaksasjon brukes om første del av diastolen hvor ventrikkelen har minimalt volum og trykket flater ut.

Av figuren kommer det klart frem at trykket i ventrikkelen er mye større enn trykket i atrium gjennom store deler av systolen.

Under diagnostisering vil kunnskap om de forskjellige parameterene, slik som volum, maks trykk, maks trykkøkning og IVRT med flere, si mye om hjertets tilstand, og kan hjelpe legen med å finne avvik fra en normal hjertefunksjon. IVRT er tiden fra aortaklaffen lukker seg til mitralklaffen åpner seg [8].

2.2 Ultralyd/ekkokardiografi

Ultralyd som medisinsk bildediagnostikk baserer seg på transmisjon og mottak av lydbølger av høy frekvens via menneskekroppen[6]. Lydbølgene som brukes har en frekvens godt over det hørbare området, og ligger typisk mellom 1,5 og 7,5 MHz[9]. Lydbølgene dannes via piezo-elektriske elementer som vibrerer i takt med en påført strøm, og det er vanlig at de reflekterte lydbølgene som er reflektert fra kroppsvevet mottas i samme probe som sendte dem ut.



Figur 2.4: Ultralyd av hjertet

Ultralydbølgene penetrerer kroppen, og avhengig av hvilke typer vev de møter, vil refleksjoner av varierende amplitude registreres i mottakeren. På den måten er det mulig å skape et bilde av innsiden av kroppen, på samme måte som for eksempel røntgenbilder, men med akustiske bølger, og ikke ioniserende(ergo skadelig) røntgenstråling.

Figur 2.4 viser et eksempel på et bilde av hjertet tatt med ultralyd, og man ser de fire hjertekammerene og klaffene mellom dem. Bildet viser hvordan vegger og tykkere vev reflekterer mer av den utsendte strålingen enn blodet og luften i kammerene. I kombinasjon med Dopplerultralyd(se kapittel 2.2.1) er det mulig å vise blodstrømmene mellom kammerene i et slikt bilde, da gjerne som røde og blå blodstrømmer avhengig av retningen på blodet.

2.2.1 Doppleralyd

Dopplereffekten ble beskrevet av danske Christian Doppler i 1842, og er en effekt som oppstår når en kilde som sender ut bølger beveger seg relativt til mottakeren. Hvis kilden og/eller mottakeren beveger seg mot hverandre vil bølgene mottakeren mottar oppfattes som å ha kortere bølgelengde enn hva kilden sendte ut, og omvendt lengre om kilde og mottaker beveger seg fra hverandre. Man kan forestille seg effekten som at bølgen blir “presset“ sammen av at kilde og mottaker nærmer seg hverandre.

Det er den relative hastigheten mellom kilde og mottaker som gjelder, og hvem som beveger seg har ingenting å si. Samtidig har vinkelen mellom dem mye å si for hvor stor effekten blir. For en kilde og en mottaker som beveger seg relativt til hverandre er den mottatte frekvensen i forhold til den utsendte gitt av formel 2.1.

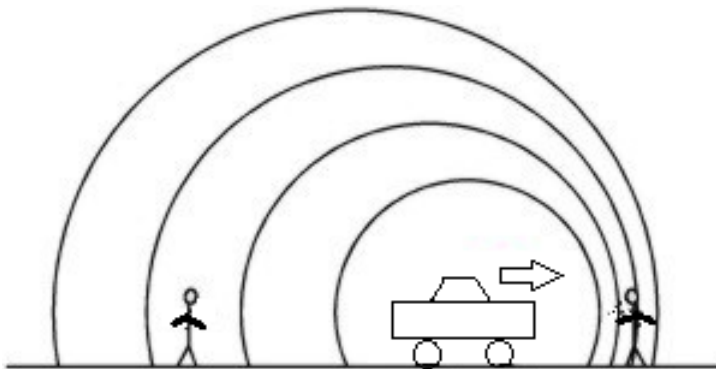
$$f = f_0 * \frac{v}{v - (v_k * \cos(\phi))} \quad (2.1)$$

v_k er hastigheten kilden beveger seg med, v er bølgehastighet og f_0 den utsendte frekvensen. ϕ er vinkelen mellom fartsretningen og den rette linjen mellom mottaker og kilden. Om mottaker og kilde beveger seg rett mot hverandre ($\phi = 0$) blir formel 2.1:

$$f = f_0 * \frac{v}{v - v_k} \quad (2.2)$$

I medisinsk bruk kan dette prinsippet utnyttes for å måle hastigheten på blod og vev inne i kroppen[6, p. 352], ergo hastighetene på blodstrømmene mellom hjertekammere. Som man kan se av formel 2.1 bør ultralydstrålen være parallell med bevegelsesretningen til det man vil måle.

Figur 2.5 demonstrerer Dopplereffekten når en bil som sender ut en lyd nærmer seg en mottaker. For personen som bilen nærmer seg ankommer bølgetoppene med kortere intervall enn personen som bilen fjerner seg fra. Dermed oppfattes bølgelengdene som kortere, og frekvensen som høyere.

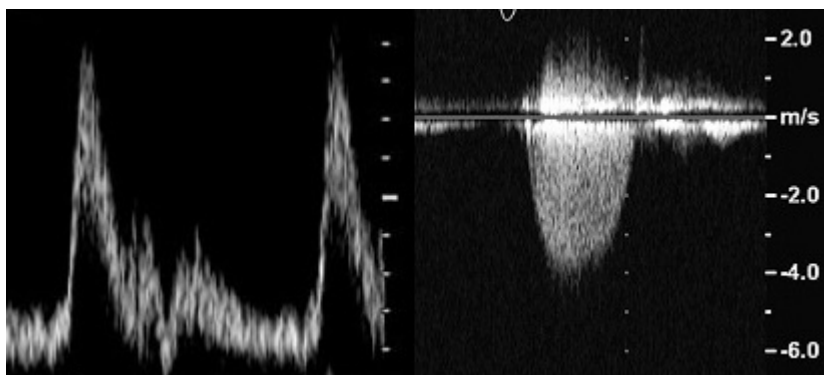


Figur 2.5: Dopplereffekten

2.2.2 Kontinuerlig og pulset Doppler

For medisinsk billedannelse av hjertet kan en Dopplermåling gi informasjon om hastighet og retning av en blodstrøm i hjertet. I dopplerultralyd måles egentlig ikke frekvensforandringer, men faseforandringer, de relative forskinkelsene i de reflekterte ultralydbølgene. Det finnes forskjellige metoder og forskjellige måter å presentere resultatene på. Dopplerspektrum presenterer blodstrømmen grafisk med fart i forhold til tid. Se figur 2.6.

Per i dag er det to metoder med forskjellige fordeler man bruker for å danne dopplerspektrum, pulset(PW) og kontinuerlig(CW) Doppler.



Figur 2.6: Eksempler på Dopplerspekter

Med pulset Doppler sender man ut pulser og måler de relative forsinkelsene mellom dem. Med PW kan man i tillegg til hastighet og retning også diskriminere mellom forskjellige dybder, man kan avgjøre ved hvilke distanser de forskjellige bevegelsene forekommer. Følgelig kan man bestemme ved hvilken dybde man ønsker å måle hastigheten, ved å overse refleksjoner som kommer utenfor et gitt tidsrom. Ulempen med PW er at man kan oppleve aliasing hvis hastighetene overskrider en grense gitt av nyquist-teoremet[10].

Begge former for Doppler har sine fordeler og ulemper som vist i tabell 2.1. Fordelen CW har over PW når det gjelder høye hastigheter gjør CW til et attraktivt valg når det gjelder å måle alvorlige avvik(som i stenoser) anført av høye blodhastigheter.

	CW	PW
Kan måle høye hastigheter	Ja	Nei
Kan diskriminere mellom dybde	Nei	ja

Tabell 2.1: Fordeler og ulemper med CW og PW Doppler

2.3 Bernoulli's prinsipp

Dopplerultralud gir en mulighet til å måle hastigheten til bevegelser inne i hjertet, i det aktuelle tilfellet blod som strømmer mellom kammere og blodårer. Selv om hastighetene i seg selv kan være gode indikatorer på egenskaper ved hjertefunksjonen, er man ofte mer interessert i å vite mer om trykk, volum og andre fysiske parametere. Med et enkelt prinsipp har man muligheten til utlede disse verdiene fra Dopplerspekteret, et prinsipp fremlagt av Daniel Bernoulli i 1738 og derfor bedre kjent som Bernoulli's prinsipp. Prinsippet er mest kjent for å (delvis) forklare hvorfor fly ikke faller til bakken, men kan også forklare andre fysiske effekter. Prinsippet er basert på prinsippet om energibevarelse[11], og gir sammenhengen mellom hastighet og trykk mellom to tverrsnitt av en sylinder. Bernoulli's likning sier at:

$$p + \frac{1}{2} * \rho * v^2 + \rho * g * h = \textit{konstant} \quad (2.3)$$

Hvor p er trykk, ρ er tetthet, v er hastighet, g er gravitasjon, og h er høyde. Likningen gjelder for væsker som er antatt stasjonære, ikke-viskøse og ikke-kompressible [12]. Gitt at en væske strømmer over en sammensnevring, vil trykket og hastigheten til væsken på hver sin side av sammensnevringen ha følgende sammenheng (det siste leddet på venstre side i likning 2.3 forsvinner):

$$p_1 - p_2 = \frac{1}{2} * \rho * (v_2^2 - v_1^2) \quad (2.4)$$

Vet man hastigheten før og etter innsnevringen kan man da regne ut trykkgradienten. I den aktuelle applikasjonen av likningen, i kardiologi, kan den ytterligere forenkles. Gitt at viskøsiteten og hastigheten er lav og neglisjerbar før innsnevringen, som i kardiologi kan opptre i form av klaffeåpninger, kan Bernoulli's prinsipp forenkles til å inneholde kun én variabel. Formel 2.5 viser den forenklede Bernoulli-likningen.

$$\Delta P = 4 * v^2 \quad (2.5)$$

ΔP symboliserer trykkgradienten mellom to volum før og etter en innsnevring ($p_1 - p_2$), mens v symboliserer hastigheten til blodet. Konstanten 4 er gitt av $1/2 * \rho$ (≈ 3.86), hvor ρ er tettheten til blod. Likningen gir oss altså trykkgradienten over en innsnevring i hjertet hvis vi har hastigheten til blodet. I tilfeller hvor trykket på den ene siden er neglisjerbart i forhold til den andre siden kan ΔP brukes som en indikator på trykket på denne andre siden, for eksempel ventrikkelen.

Gitt en hastighetskurve slik man får fra en Dopplermåling, kan denne kurven i henhold til likning 2.5 konverteres til en trykkgradientkurve.

2.4 Ekkokardiografi som klinisk verktøy

Ekkokardiografi er et effektivt verktøy på mange måter for å undersøke hjertet. Bruk av Doppler sonografi gir en også muligheten til å måle hastighet og retning på vev og blod i bevegelse. En Dopplerundersøkelse kan brukes for å måle parametere som klaffeareal og klaffefunksjon, om det er lekkasjer, volumet av blodet som strømmer samt andre parametere. Kunnskap om hastighetene til blodet kan også gi kunnskap om trykket ved gitte områder i hjertet, på grunn av forholdet mellom trykk og hastighet som beskrevet av Bernoulli's prinsipp og forklart i kapittel 2.3. Videre følger en beskrivelse av hvordan denne kunnskapen kan benyttes i diagnostisering/forskning.

Å undersøke ventrikkelfunksjonen er ofte viktig i undersøkelser av pasienter og ved å studere lekkasjer gjennom mitralklaffen, fra venstre ventrikel til atrium, er det mulig å si mye om denne. Klaffene i hjertet skal sørge for at blod kan strømme i den korrekte retningen, og forhindre at blodet strømmer tilbake. Noen ganger forekommer det lekkasjer i disse klaffene og mindre blod vil leveres til sirkulasjonssystemet. Slike lekkasjer er vanlige i forbindelse med flere diagnoser, og alvorlighetsgraden kan varieres fra mild til alvorlig. Mitralinsuffisiens(eng: mitral regurgitant jet) er begrepet på en lekkasje gjennom en skadet mitralklaff. Skaden på klaffen kan være medfødt, men er ofte et resultat av et hjerteinfarkt eller andre lidelser.

En slik lekkasje, så lenge den er kraftig nok, vil vises på ultralydbilder og med ekkokardiografi kan man måle hastigheten på denne lekkasjen. Holen et.al. viste i 1976 at trykkgradienten man finner fra Dopplerspektrum med stor nøyaktighet kan estimere den reelle trykkgradienten i hjertet[1]. Tanken er at Dopplermålinger skal være et alternativ til å måle trykk i hjertet med en invasiv undersøkelse. Trykket i atrium kan anses som stabilt og neglisjerbart i forhold til trykket i ventrikkelen, og trykkgradienten over mitralklaffen kan derfor anses som en god indikator på trykket i ventrikkelen[13, p.1]. Spesielt forandringen i ventrikketrykket(dP/dt) er en vanlig parameter i kartlegging av ventrikkelfunksjonen[14]. I blant annet [8, 13, 14, 15] verifiseres nytteverdien av å bruke Dopplermålinger til å kvantifisere parametere som RPR, dP/dt og τ (relaksasjonskonstanten). RPR er et anslag for maksimal dP/dt , og er gitt av stigningstallet til trykkøkningen mellom 1 og 3 m/s. τ er en indikator på isovolumetrisk relaksasjon. Det er også mulig å kvantifisere andre parametere ut fra Dopplermålingene, gjerne hvis man i tillegg har kunnskap om andre fysiologiske parametere.

Ikke bare mitrallekkasjer kan benyttes til slike formål. I en artikkel av Nakatani et.al. [16] beskrives en metode for å finne dP/dt i venstre atrium ved å måle trans-mitrale blodstrømmer og blodstrømmer fra pulmonale vener. En algoritme for å spore konturene av Dopplerspektrumet bør derfor kunne benyttes for andre typer spektrum enn fra bare mitralinsuffisienser.

2.5 Verktøy

2.5.1 DICOM

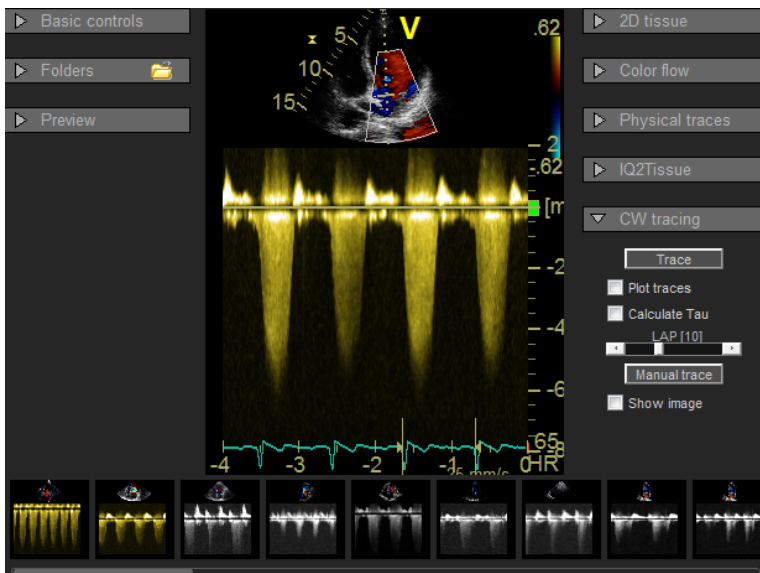
DICOM er en standard for å blant annet lagre og sende informasjon innenfor medisinsk billedannelse[17]. Dette innebærer et filformat for lagring av pasientinformasjon og billedata. Standarden støtter en hel rekke medisinske tomografimetoder, også ultralydbilder. Med DICOM kan radiografiapparater, arkiv og datamaskiner snakke sammen, og gir en effektiv informasjonsutveksling som gir raskere pasientbehandling og diagnostisering.

2.5.2 Matlab

For utvikling og testing er kode skrevet i Matlab. Matlab versjon 7.11.0 (R2010b) er installert på en PC med Windows 7. PC'en har en Intel Core i3 prosessor, et NVIDIA GeForce 310M grafikkort samt 4 GB minne.

2.5.3 GcLab

GcLab er et Matlab-basert dataverktøy for visning og analysing av ultralydbilder. Programmet er utviklet ved ISB, NTNU i samarbeid med GE Vingmed Ultrasound AS. Programmet har tilgang til ultralyd rådata gjennom en ActiveX komponent som kobler seg på display-enheten til GE Vingmed Ultrasound AS.



Figur 2.7: Skjerm bilde av GcLab

Kapittel 3

Metode

I dette kapitlet vil den foreslåtte algoritmen bli beskrevet i detalj. I forbindelse med denne oppgaven er kode og arbeid utført i GcLab. Kode presentert er derfor skrevet i Matlab, og er tilpasset GcLab, men selve algoritmen er universell. Metoden vil presenteres stegvis hvor hvert steg av prosessen vil argumenteres for med pseudokode og forskjellige eksempler på dopplespektrum.

I seksjon 3.1 vil datamaterialet som koden er blitt testet på beskrives. Videre i seksjon 3.2 argumenteres det for hvorfor fremgangsmåten er valgt, og hvilke fordeler den har. Det vil også bli beskrevet for hvilke type spekter algoritmen er rettet mot, og hva som må til for å gjøre den universelt fungerende. Seksjon 3.3 vil ta for seg kvaliteten på spektrumene som er brukt i testingen, og vurdere hvordan algoritmen skal håndtere spektrum av varierende kvalitet.

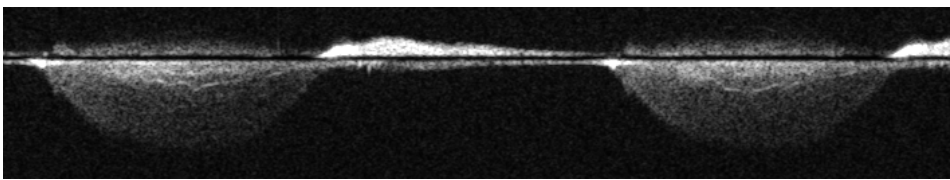
Ved hjelp av figurer, pseudokode og tekst vil første del av algoritmen bli beskrevet i seksjon 3.4. I seksjon 3.5 vil den andre delen av algoritmen beskrives, før man i seksjon 3.6 oppsummerer hele algoritmen med en figur og en beskrivelse av metodene i programmet

3.1 Datamateriale

Med GcLab kan man hente ut spektraldata i form av et to-dimensjonalt bilde. For eventuelle andre applikasjoner som for eksempel en implementasjon på et ultralydapparat, vil algoritmen fungere om man fryser Dopplermålingene, og utfører algoritmen på denne.

Spektraldataen som er brukt i testing og implementering av algoritmen er hentet ut i form av 38 Dopplerspektrum(CW) fra DICOM-bilder utlevert fra institutt for sirkulasjon og bildediagnostikk ved NTNU. Spektraldataen leses ut i form av et to-dimensjonalt bilde, med en vertikal oppløsning på 128 piksler, samt et antall horisontale piksler som alle representerer rammer på noen fåtalls millisekunder. I motsetning til hva som er vanlig i mange applikasjoner, går den vertikale oppløsningen i Matlab fra topp til bunn. Med andre ord er piksel 1 øverst, og piksel 128 nederst. Videre i rapporten må det derfor tas hensyn til dette når det vises til toppunkt og bunnpunkt på den detekterte kanten.

Fra DICOM-filene er det også tilgang til annen data slik som EKG-data, tid-, og hastighetsakse og 2D-ultralyd.



Figur 3.1: Eksempel på spektraldata

Figur 3.1 viser et eksempel på spektraldata som prosesseringen er utført på. Bildet viser tydelige strømmer av blod, i dette tilfellet blod som strømmer tilbake til atrium som en konsekvens av en mitralinsuffisiens(kapittel 2.4). Den svarte horisontale linjen kalles “baseline“, og er stedet hvor hastigheten er 0. Over og under denne streken øker hastigheten enten med positivt eller negativt fortegn, avhengig av om blodet strømmer mot eller fra ultralydproben. Hastighetene under baselinjen har i alle bildene benyttet i prosjektet negativt fortegn. Den vertikale aksene spenner dermed fra en maksimal positiv hastighet øverst, og til en maksimal negativ hastighet nederst. Samtidig strekker den horisontale aksene seg over et tidsrom på et visst antall sekunder, og dermed tilsvarer en piksel et tidsrom gitt av det totale tidsrommet til spekteret delt på antall horisontale piksler.

3.2 Valg av metode

Den foreslåtte algoritmen er valgt fordi prinsippet bak den er fra erfaring ansett for å detektere kanter effektivt, er robust mot høye mengder støy i spektraldataene, samt at den i motsetning til arbeidet i [2, 3, 4, 5] gjør et forsøk på å predikere deler av spektraldata som er ufullstendige. Denne predikeringen gjøres med en modelltilpasning basert på analyser av manuelle sporinger og ved at algoritmen ved erfaring lærer seg formen på Dopplerspektrum. Denne læringen er basert på forutsetningen at et Dopplerspektrum reflekterer trykket i hjertet, og derfor har en noenlunde konstant form[3].

Alternative forslag til automatiske sporingsalgoritmer er også presentert i eksempelvis [3] og [5], og det er tydelig der at sporingen av konturene av blodstrømmen i flere tilfeller bommer grovt, selv på bilder med tydelige kanter, med det resultat at toppunktet ikke finnes nøyaktig eller at opp og nedstigende kontur ikke detekteres korrekt.

Et essensielt punkt i vurderingen av algoritmen er ikke bare evnen til å detektere kanter, men også evnen til å etterarbeide konturene som er ekstrahert, predikere manglende informasjon og preparere dem for videre utregninger. Det implementerte programmet inneholder også funksjonalitet for å regne ut trykkverdier, gradienter og andre parametere basert på de ekstraherte konturene.

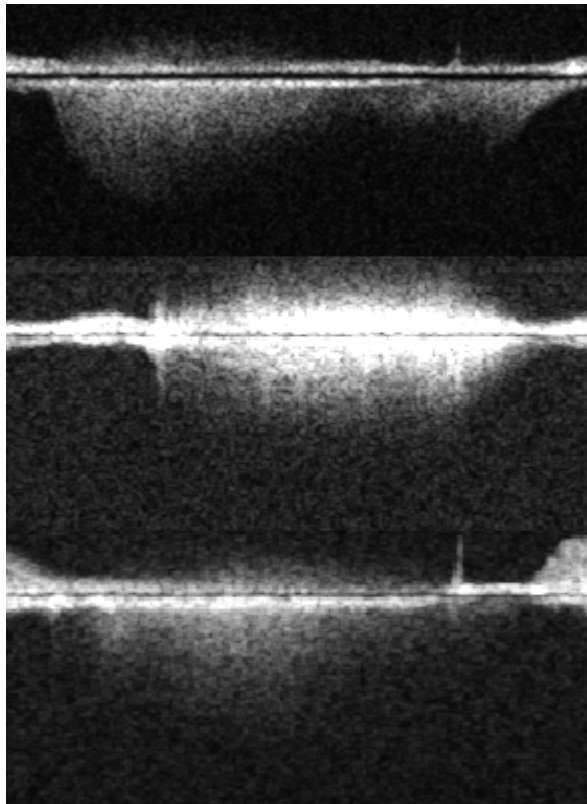
Algoritmen er basert på fryste bilder, noe som innebærer at den ikke fungerer for sanntidsbehandling. Dette er gjort fordi det er ikke nok å tjene på å implementere en sanntidsalgoritme i forhold til hvordan algoritmen fungerer nå. Om en skal analysere spektraldata er det enkelt å fryse et bilde og starte programmet. Man får da fordeler ved at det er enklere å finne kanter, og at man kan bruke data fra andre strukturer for å predikere mangler ved andre strukturer i bildet.

Algoritmen er først og fremst tilpasset målinger av mitralinsuffisiens med CW Doppler. Modellen som blir brukt er også under utvikling basert på CW Doppler, men algoritmen generelt fungerer i teorien like godt for både PW som CW.

Et av argumentene for å ikke la algoritmen basere seg 100% på modeller, slik som annet tidligere arbeid har gjort er at den er fleksibel i forhold til andre typer blodstrømmer man ønsker å måle, for eksempel i forbindelse med atrialflimmer, eventuelt i forhold til bruk av PW Doppler. Hadde man basert hele algoritmen på en modell, måtte modeller utarbeides for alle typer spekter. Slik det fungerer nå er predikeringen tilpasset spekter av mitralinsuffisiens, mens kantdeteksjonen er universell. Å tilpasse modellen til andre typer spekter vil kreve at man studerer formen på disse spektrene, og lager en modell. Samtidig vil en algoritme som baserer seg kun på en modell i mange tilfeller gi feil resultat i sporing av Dopplerspektrum av god kvalitet, noe man kan se i [3].

3.3 Kvalitet på spektraldata

Det er varierende grad av kvalitet på spektraldataen. Figur 3.2 viser eksempler hvor spektraldata har dårlig kvalitet, og som er bortimot umulig å spore selv manuelt.



Figur 3.2: Eksempler på dårlige Dopplermålinger

Tidligere arbeid på sporing av spektraldata er utført på spektraldata av god kvalitet, men denne algoritmen vil også gjøre et forsøk på å predikere verdier hvor spektraldataen er mangelfull. Nytteverdien er diskutabel, i [18] argumenteres det for viktigheten av klare konturer av hastigheten for å finne riktige verdier, og det vises til store avvik i korrekte verdier for bilder med dårlig kvalitet. Ved å studere manuelle sporinger gjøres det likevel et forsøk på å predikere kanter, spesielt rettet mot bilder hvor kun en del av bildet er svakt, med øvrige klare og skarpe kanter, slik som øverst i figur 3.2.

Til slutt er det likevel opp til radiografen eller legen å avgjøre om predikeringen er god nok, og om kantdeteksjonen viser riktige verdier. De endelige resultatene bør i alle tilfeller, uavhengig av kvalitet, avgjøres av lege om de er troverdige eller ei.

3.4 Preprossesering av Dopplerspekteret

For å finne konturene må det gjøres en del arbeid som forsterker de lyse områdene og svekker de mørke. Funksjonen *filterImage* tar inn et spektralbilde slik som for eksempel figur 3.1. For at algoritmen skal være robust uavhengig av nivå av støy og kontrast foretas det først en enkel analyse av bildet. Bakgrunnsstøyen kvantifiseres i form av en mean-verdi μ_1 . For å finne verdien velges et område i bildet som er antatt å ikke inneholde annet enn bakgrunnstøy, og er derfor valgt basert på en blokk øverst til venstre i bildet.

Det neste steget er å nøytralisere bakgrunnstøyen. Da vil man blant annet unngå at kantdeteksjonsmekanismen vil skape uønskede artifakter på grunn av bakgrunnsstøy. Det er viktigere å finne kantene nøyaktig ved opp og nedstigende kontur, for det er her unøyaktig sporing vil få størst konsekvenser for utregning av parametere. Et ørlite avvik fra den virkelige kanten vil gi helt andre verdier enn man ønsker. På toppen av konturen, nær maks hastighet (medio systole), ønsker man en glattet kontur, for det er slik formen på ventrikeltrykket er (figur 2.3). Parameterene man ønsker å regne ut her er heller ikke avhengig av punkt-til-punkt variasjoner slik det er ved tidlig og sen systole. Den vertikale akse på bildet representerer hastighet, og den horisontale tid. Av disse grunnene ser man på bildet som todelt, hvor den nedre delen som omslutter de høye hastighetene separeres fra den andre. Man subtraherer μ_1 fra den delen med lave hastigheter, og $\frac{\mu_1}{2}$ fra den andre delen.

Videre analyse av bildene viser at amplituden på spektraldataen blir lavere med økt hastighet på blodet. I bilder med for mye bakgrunnstøy (ved at μ_1 overskrider predefinert terskel), kompenserer man for dette ved å multiplisere bildet med en lineært økende vektor vertikalt, man øker med det "gain" mer ved de høye hastighetene enn ved de lave. Dette viste seg å gjøre at konturene av hastigheten ble bedre sporet helt til toppen. Prosessen er beskrevet i algoritme 1.

Algoritme del 1 preprossesering

```

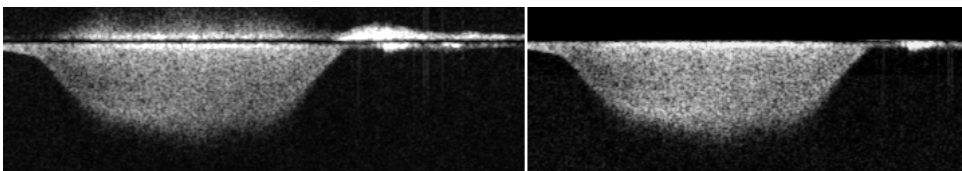
 $\mu_1 = \text{mean}(\text{bakgrunn})$ 
val = et sted mellom baseline og høyde
if  $\mu_1 > 6$  then
    vektor = 1:0.5/høyde:1.5
    for  $i = 1 \rightarrow \text{høyde}$  do
        bilde(i,:) = bilde(i,:)*vektor(i)
    end for
end if
for  $i = 1 \rightarrow \text{val}$ ,  $k = 1 \rightarrow \text{bredde}$  do
    bilde(i,k) = bilde(i,k) -  $\mu_1$ 
end for
for  $i = \text{val} \rightarrow \text{høyde}$ ,  $k = 1 \rightarrow \text{bredde}$  do
    bilde(i,k) = bilde(i,k) -  $\frac{\mu_1}{2}$ 
end for

```

3.4.1 Filtrering

Det er brukt to forskjellige filter på bildet. Det ene er et lavpass Gaussisk filter med varians $\sigma = 0.5$, valgt empirisk, det andre er et medianfilter. Det gaussiske filteret er implementert for å fjerne støy og for å glatte ut skarpe kanter. Medianfilteret er implementert fordi bakgrunnstøyen synes som impulsstøy, hvor individuelle piksler, eller grupper av piksler, er lysere enn omgivelsene. Tanken bak medianfilteret er å nøytralisere dette.

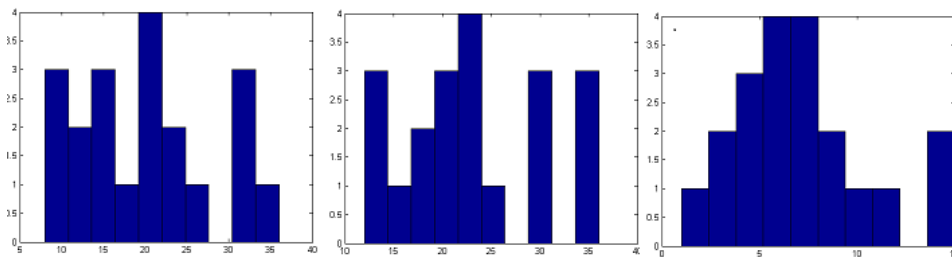
Valg av filter er til dels basert på prøving og feiling, hvor mange forskjellige kombinasjoner er forsøkt og testet.



Figur 3.3: Eksempel på spektraldata før og etter preprossesering

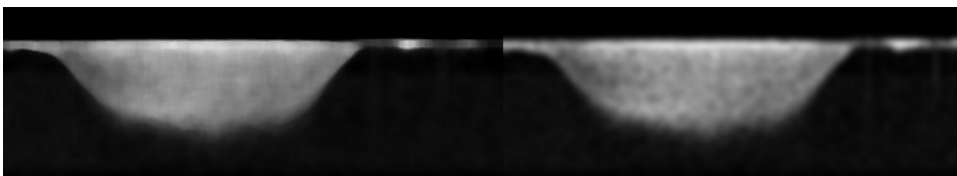
Figur 3.3 viser spektraldata før og etter preprossesering beskrevet i seksjon 3.4. Man ser også at all informasjon over baselinje (null-hastighetspunktet) er fjernet.

Medianfilteret er et filter av størrelse 10×10 , som konvuleres med bildet og setter pikslene i blokken til medianen av blokken. Denne type filter fungerer veldig godt på bilder hvor en liten gruppe piksler skiller seg ut fra omgivelsene, noe som ofte er tilfelle i spektraldataen, som observert i histogrammene i figur 3.4.



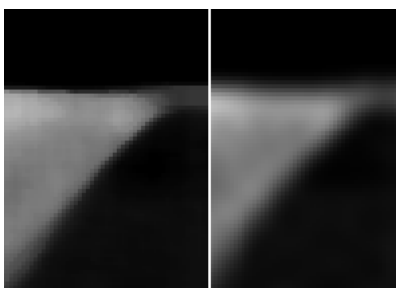
Figur 3.4: Tre eksempler på histogram til en blokk i bildet

På grunn av dette ble medianfilteret valgt ut som et alternativt filter, hvor det har blitt sammenlignet med andre lavpassfilter.



Figur 3.5: Spektraldata etter median og gaussisk filtrering

Figur 3.5 viser forskjellen på filtrering med et medianfilter og et gaussisk filter (et average-filter ga lignende resultat). Medianfilteret fjerner effektivt de mørke og lyse områdene (prikkene), og beholder samtidig kantene mye bedre enn det vanlige lavpassfilteret. Disse resultatene gikk igjen med forskjellige parametere og medianfilteret ble derfor benyttet videre. Figur 3.6 viser et utsnitt som demonstrerer forskjellen på filterene bedre.



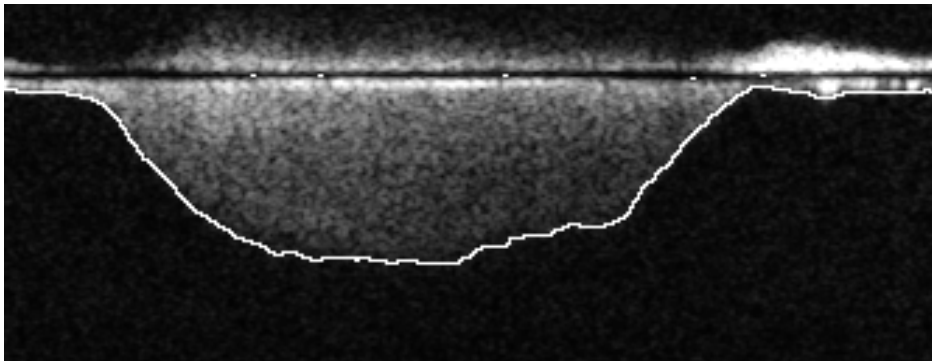
Figur 3.6: Utsnitt av figur 3.5

Neste steg i prosessen vil være å finne en terskel for å separere de lyse områdene fra bakgrunnen. Terskelverdien må basere seg på amplituden av den resterende bakgrunnstøyen. I stedet for å sette alle pikslene i bildet med verdi lavere enn terskelen til 0, filtreres bildet en gang til, med et gaussisk filter med størrelse 10×1 , og $\sigma = 0.5$. Dette viste seg å forbedre kandedeksjonen ytterligere. Terskelen settes lik medianen til bakgrunnstøyen, \tilde{x} , og alle piksler lysere enn denne settes til 255, og resten til 0. Man sitter da igjen med et binærbilde (sort-hvitt), hvor ytterkantene enkelt kan spores.



Figur 3.7: Binærbilde og bildet med kun ytterkantene

Denne metoden er effektiv og robust mot forskjellige nivå av støy. Samtidig er den rask, og kjøres i Matlab på brøkdelen av et sekund. Resultatet vises i figur 3.8. Selve kantdeteksjonen er likevel bare én del av algoritmen. For at algoritmen skal kunne brukes for videre utregninger må konturen jevnes ut og den må ta hensyn til ufullstendige strukturer. Denne prosessen er beskrevet i seksjon 3.5.



Figur 3.8: Kantdeteksjon

Den delen av algoritmen som innebærer filtrering og tersking er demonstrert i del 2 av algoritmen.

Algoritme del 2 filtrering og tersking

```

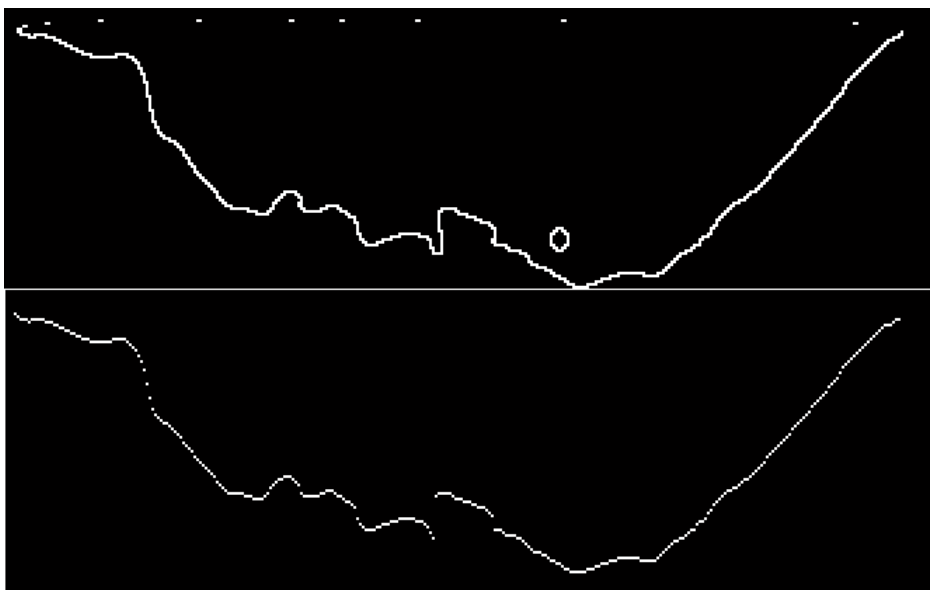
 $\tilde{x}$  = median(bakgrunnsstøy)
filtrer(medianfilter, [10 10])
filtrer(gaussisk filter, [10 1])
for  $i = 1 \rightarrow$  høyde,  $k = 1 \rightarrow$  bredde do
  if  $\text{bilde}(i, k) / \tilde{x} > 1$  then
    bilde(i,k) = 255
  else
    bilde(i,k) = 0
  end if
end for
for  $i = 1 \rightarrow$  høyde,  $k = 1 \rightarrow$  bredde do
  if  $\text{bilde}(i, k) \in \text{kant}$  then
    bilde(i,k) = 255
  else
    bilde(i,k) = 0
  end if
end for

```

3.4.2 *snake*

Første steg etter filtrering er å spore hele kanten og lagre den for videre prosessering. Bildet gjennomføres og kanter som ikke er relevante (artifakter) forkastes. Når en gyldig kant er funnet spores hele denne og lagres. Bildet deles opp i forhold til start og slutt på hver hjertesykklus med hjelp av starten på hvert QRS-kompleks. Dette gjøres i en funksjon kalt *snake*. Når man har funnet en lys piksel, arbeider man ut i fra denne, og hopper videre til en av naboene, markerer denne og lagrer den. Slik fortsetter man til det ikke er flere piksler igjen å finne. Noen ganger ender man opp på en blindvei slik at man ikke klarer å følge resten av kanten. Det er da implementert en robust mekanisme for å søke videre i den retningen man var i til man igjen kan fortsette sporingen. Etter at en fullstendig kant er funnet, fjerner man overflødig punkt, slik at man kun sitter igjen med et punkt per horisontal piksel.

Etter filtreringen vil man ofte sitte igjen med små artifakter i bildet, spesielt i bilder med mye støy. For å avgjøre om kanten er en artifakt eller er en gyldig kantsporing, må man sette noen krav. Denne utvelgelsen baserer seg på at man kun har en blodstrøm per hjertesykklus. Dermed velges den som oppfyller riktige krav i forhold til høyde, lengde, start- og slutt punkt. Det er i ytterst sjeldne tilfeller at det er et problem å skille mellom den virkelige kanten og en artifakt, da artifaktene opptrer som små flekker i forhold til den virkelige kanten. Figur 3.9 viser en detektert kant før og etter *snake*.



Figur 3.9: Bildet viser den sporede kanten av figur 3.10 før (øverst) og etter (nederst) metoden *snake* er kjørt. Det er tydelig hvordan artifakter er fjernet.

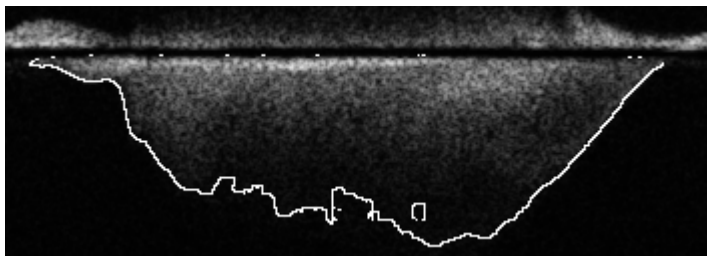
Underveis i prosessen lagres informasjon om høyeste og laveste punkt med mer. Algoritmen er enkelt beskrevet under.

Algoritme del 3 snake

```
input = binærbilde
while flere_hvite_piksler = sant do
  Gå gjennom bilde, finn hvit piksel.
  while flere_naboer = sant do
    Lagre piksel. Farg piksel
    Finn nabo
  end while
  if struktur = gyldig then
    En piksel per horisontale koordinat
    Lagre struktur
  end if
end while
```

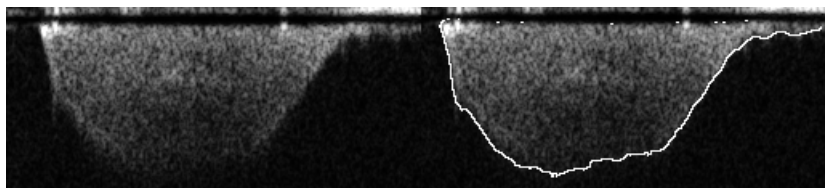
3.5 Etterbehandling av sporingen

Eksempelet i figur 3.8 viser et bilde hvor kantdeteksjonen viser et jevnt og godt resultat. Videre etterarbeid er dermed begrenset til utjevning av konturene. Å finne så gode konturer er derimot ikke alltid tilfelle, som man kan se i figur 3.10.



Figur 3.10: Kantdeteksjon med ujevn kant

Det er hovedsaklig to grunner til at det er viktig å etterbehandle sporingene, det ene er for å predikere kanter som ikke er synlige på spektraldataene, men som man vet skal være der, og samtidig fjerne kanter som av diverse årsaker ikke skal være der. Det andre er for å jevne ut konturene, siden de detekterte kantene i utgangspunktet har store variasjoner fra punkt-til-punkt, som kan anses som støy. Samtidig er deler av Dopplerspekteret et resultat av andre faktorer vi ikke ønsker å måle. Dopplermålinger diskriminerer ikke mellom hva som forårsaker hastigheten. Et eksempel på dette er hastigheter forårsaket av klaffer som åpner seg, som ikke uten videre kan skilles fra hastigheten til blodet. Figur 3.10, og spesielt figur 3.11, viser spektraldata hvor det er en tydelig ekstra påvirkning på oppadstigende kontur. Dette er antatt å være forårsaket av mitralklaffen som åpner seg, og bør således ikke tas med i utregningene av trykket.

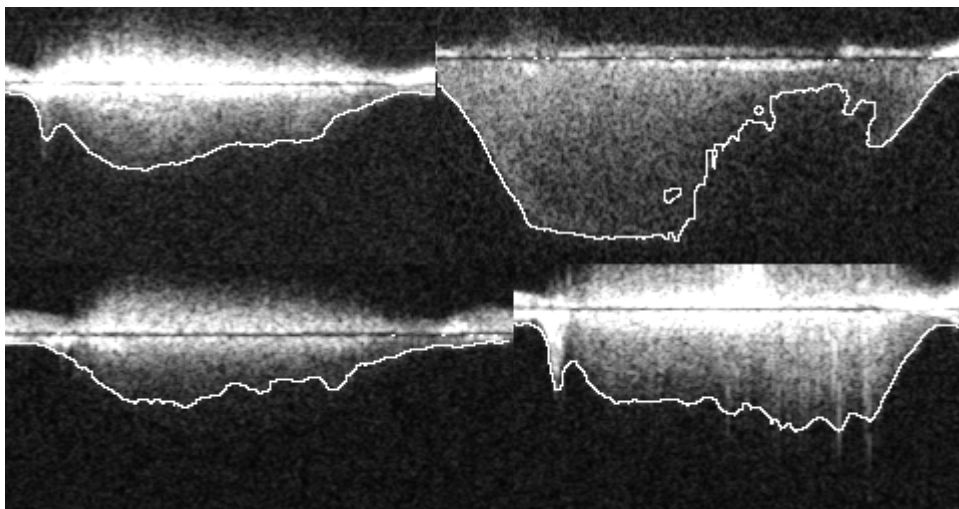


Figur 3.11: Dopplerspektrum med tydelig klaffehastighet med og uten sporet kant

Et annet eksempel på dette er målinger av blodhastigheter i pasienter med mitralinsuffisiens. Disse målingene kan brukes til å indikere trykket i venstre ventrikel, og vi vet at ventrikketrykket synker til en verdi lavere enn Dopplermålingene viser. Hastighetene i en slik Dopplermåling synker ikke så lavt fordi hastighetene er påvirket av andre faktorer enn bare trykket i ventrikkelen, men også trykket i atrium. Også dette er tydelig vist i figur 3.11, hvor hastighetene under den isovolumetriske relaksasjon ikke faller mot null.

3.5.1 Analyse

Som det skal vise seg senere fungerer kantdeteksjonen godt på bilder av høy kvalitet. Men å analysere sporingen er essensielt for senere å kunne regne ut verdier, da kantene i større eller mindre grad er ujevne. Algoritmen er dessuten ment å være robust mot bilder av dårlig kvalitet. Analyse er da nødvendig for å avgjøre om sporingen trenger predikering, om deler av Dopplerspektrumet er forårsaket av andre ting enn blod, og avgjøre hva som skal gjøres videre. Riktigheten av predikeringen kan ikke garanteres og det må poengteres at det er opp til behandleren av bildene å avgjøre nøyaktigheten til predikering.



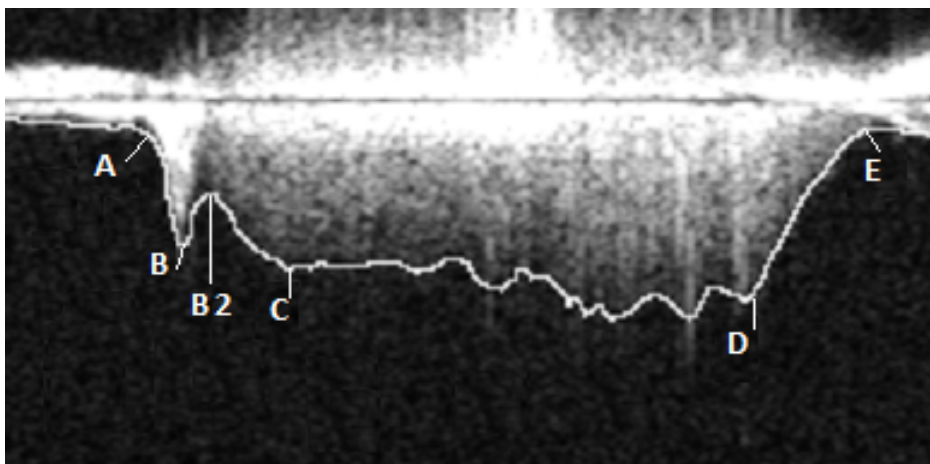
Figur 3.12: Diverse bilder som krever predikering

Figur 3.12 viser noen eksempler på bilder som krever predikering. Det er tydelig at kantdeteksjonen fungerer godt, men det er rett og slett for dårlig kvalitet på spektraldataen. På bildene øverst til venstre og nederst til høyre i figuren er det en tydelig påvirkning av klaffeåpningen på oppstigende kontur, og på alle er det tydelig at toppen av konturen er mangelfull og ujevn. I bildet øverst til høyre ser man hvordan et stort segment mangler. Man vet hvordan det egentlig skal se ut da vi vet hvordan trykket i ventrikkelen utvikler seg, vi vet det fra tidligere erfaringer (a priori), samt at man, selv om det i dette tilfellet er umulig, kan skimte en meget svak kant som kantdeteksjonen ikke fant.

Alle disse problemene må algoritmen ta hånd om, og første steg er å analysere kantdeteksjonen og jobbe ut i fra dette.

Analysen foretas ved at man følger kanten fra start. Fra DICOM-bildene har man informasjon om maksimum og minimum hastighet i tillegg til tid, og man kan derfor direkte konvertere den horisontale og vertikale koordinaten til pikselen til en gitt tid og hastighet. Figur 3.13 demonstrerer viktige punkt man ønsker å lokalisere langs konturen.

A er punktet som markerer starten på mitralinsuffisiensen. B er det punktet som



Figur 3.13: Bildet viser en sporet kant, og markerte viktige punkter langs kanten

eventuelt viser hvor påvirkningen av en klaff slutter. C markerer slutt punktet på den oppadstigende kurve hvor vi må predikere fra, alle verdier etter dette punktet er usikre. D er starten på den nedadstigende kurve, og E er slutten. Trykket i ventrikkelen faller mot null, så kurven må predikeres videre mot null fra både A og E.

B er et punkt som bare finnes i enkelte bilder, og punktet finnes ved å følge langs kanten til man finner et klart definert "brudd" i kurven. For at man skal vite at dette punktet egentlig ikke er det samme som punkt C må B finne sted før en viss hastighet er nådd, for eksempel før halvparten av topphastigheten. B2 er punktet hvor man antar at hastigheten igjen kommer fra blodet, og derfra følges sporingen til punkt C. C finnes ved å se etter hvor oppstigende kontur blir brutt, noe man kan se ved at den flater ut. Data etter dette ikke er ansett som ikke pålitelig. Både D og E finnes på tilsvarende måter, men fra andre siden. Både A og E finnes ved at man starter fra midten av opp- og nedstigende kurve til vi finner punktet hvor kanten flater ut.

Den siste avgjørelsen en slik analyse må foreta seg er hvorvidt kanten trenger predikering, eller om verdiene mellom C og D kun er "støy". Figur 3.13 viser et bilde hvor det er tydelig for et menneske å se at det er behov for predikering, dette fordi kanten er relativt flat mellom C og D, mens man vet at en slik hastighetskurve har en gradvis stigning mot et toppunkt. Ett kriterie for å avgjøre om en kant trenger predikering kunne derfor vært å se om kanten mellom C og D har en slik form. Forslag til slike kriterier vurderes i seksjon 3.5.3.

Det er klart at enkelte aspekt ved analysemetoden ikke er 100% pålitelig, men feilanalyse er som oftest begrenset til spektre av dårlig kvalitet.

3.5.2 Interpolering og ekstrapolering

Interpolering innebærer at man fyller inn manglende verdier på en linje basert på verdiene før og etter “hullet”. Ekstrapolering vil si å forlengende en linje slik at man får en naturlig forlengelse av linjen. Begge deler er en predikering, og verdienes nøyaktighet kan dermed ikke garanteres.

Analysen av kanten i forrige seksjon har gitt oss en rekke markeringer på kanten som definerer viktige steder. Sagt på en annen måte så er kanten delt opp i segmenter hvor alle kan bearbeides på forskjellige måter ut i fra forskjellige behov. A-B2, et segment som finnes i enkelte sporinger, er forårsaket av en klaffehastighet og må predikeres. B2-C(eventuelt A-C) er en del av den oppadstigende kurven, og skal kun utjevnes. C-D må enten predikeres eller utjevnes. D-E skal kun utjevnes. Verdiene før A og etter E må ekstrapoleres. En konsekvens av denne oppdelingen er at verdier som måles andre steder enn mellom A/B2 og C eller mellom D og E må antas å være predikerte, og ikke 100% pålitelige.

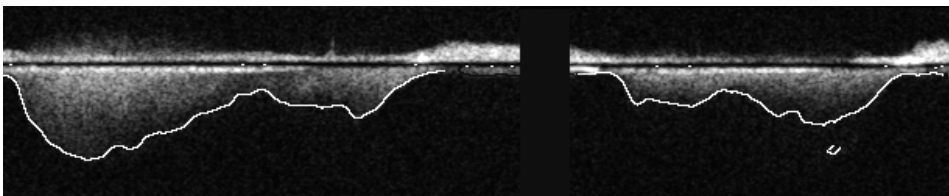
Bearbeiding av kanten må også ses i lys av hvilke parametere det er aktuelt å finne. Vil eventuell predikering ødelegge troverdigheten til disse? Figur 2.3 viser en oversikt over hvordan trykket i ventrikkelen og atrium utvikler seg. Seksjon 2.4 tar for seg hva man ønsker å kvantifisere med Dopplerultral lyd, og enkelte av parametrene er mer utsatt for unøyaktigheter enn andre. Maksimal og minimal dP/dt er eksempler på parametere som er veldig følsomme for små variasjoner. Også relaksasjonskonstanten τ vil forandre seg mye hvis nedadstigende kontur og ekstrapolering etter E er unøyaktig. Oppadstigende og nedadstigende kant må derfor bevares så godt som mulig. Om vi har påvirkning fra en klaff, det vil si at det finnes en B, må segmentet A-B predikeres. Hvis ikke vil vi få en veldig bratt kant og vi kan få falske verdier for den maksimale dP/dt .

For å interpolere har Matlab en innebygget interpoleringsfunksjon, *interp1*, som gir en muligheten til å erstatte manglende verdier langs en kurve basert på en underliggende funksjon. Det finnes forskjellige funksjoner å bruke hvor for eksempel lineær interpolering ikke er annet enn en rett linje istedetfor de manglende verdiene. En rett linje er derimot ikke en realistisk interpolering. I mange praktiske sammenhenger er det bedre å bruke en såkalt kubisk spline[19]. Denne har også blitt brukt i andre prosjekter som omhandler ekkokardiografi, for eksempel i [20, 21]. Spline-metoden gir en realistisk interpolering, og er bedre enn å bruke et høyere-ordens polynom, da du slipper uønskede svingninger i kurven, det såkalte “Rungesfenomenet”. Med en spline interpoleres det med et lavere-ordens polynom mellom hvert punkt som interpoleres, og alle leddene tilpasses hverandre. En slik type interpolering brukes i design av for eksempel båter og fly, og har vist seg å minimere interpolasjonsfeilen.

For å ekstrapolere konturens fall mot 0 m/s(verdiene etter E), brukes det lineære stigningstallet til nedadstigende kontur(stigningstallet mellom E og D), for å predikere hvor den vil treffe null-punktet. Verdiene mellom E og dette punktet interpoleres så med en kubisk spline. Ekstrapoleringen er basert på hvordan trykket er ventet å utvikle seg ved å studere figur 2.3.

3.5.3 Modelltilpasning/predikering

Toppen av kurven, definert som segmentet mellom punkt C og D, er ofte såpass mangelfull at en interpolering ikke er nok. De to bildene i figur 3.14 viser noen eksempler på bilder som krever predikering.

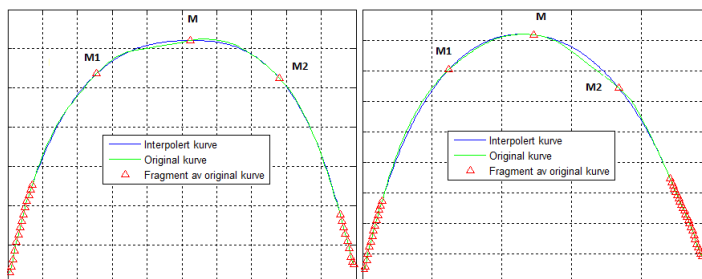


Figur 3.14: Spektre som må predikeres

For å avgjøre om et bilde krever predikering må enkelte kriterer ligge til rette. I eksemplene ovenfor er det ganske tydelig at det er mye som mangler av informasjon i C-D-segmentet(figur 3.13), men i andre bilder igjen er det ikke fullt så tydelig. Under analysen av strukturen avgjøres det om C-D har små variasjoner og om dette segmentet ligger høyere enn både punkt C og D. Om ikke disse kriteriene møtes må segmentet fra C-D predikeres.

For å finne en korrekt modell som kan erstatte den manglende informasjonen har 20 manuelt sporede kanter blitt studert. Målet har vært å finne en generell form som kan minimere feilen mellom de egentlige verdiene og de predikerte. Eksempler på noen modeller som i utgangspunktet ble vurdert var å erstatte C-D med en sinuskurve eller lignende, eventuelt en generelt tilpasset manuelt sporet kant fra andre bilder. Den endelige løsningen ble en kombinasjon av predikering og interpolasjon basert på studering av manuelle sporinger.

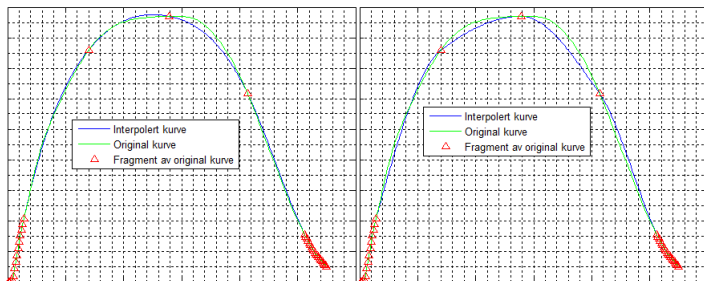
I seksjon 3.5.2 ble vi introdusert til begrepet interpolering. Mulighetene for å nøyaktig predikere mellomliggende verdier gjorde at interpolering ble valgt som basis for modellen.



Figur 3.15: Viser at om man kan predikere M1, M og M2 nøyaktig, så vil en spline interpolere de mellomliggende verdiene nøyaktig

Figur 3.15 viser eksempler på noen manuelt sporede kanter. Den grønne kurven viser denne. For å teste hvor god en interpolering er ble alle punkt på kurven

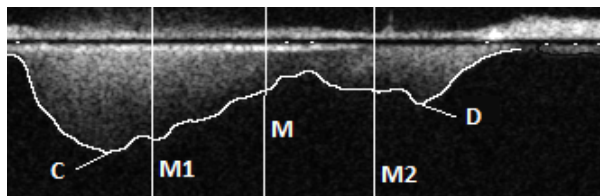
utenom start og slutt, samt tre punkt ved 25(M1), 50(M) og 75% (M2) av lengden forkastet (de røde punktene), og hele kurven ble så interpolert med en stegvis spline mellom punktene. Den blå kurven viser den interpolerte. Figur 3.16 viser forskjellen når man interpolerer med et polynom, og viser hvor mye bedre spline interpolerer.



Figur 3.16: Viser at om man kan predikere M1, M og M2 nøyaktig, så vil en spline (til venstre) interpolere bedre enn et polynom

Figur 3.15 og 3.16 viser at så lenge man kan finne M1, M og M2 (se figur 3.17) kan en spline brukes for å interpolere de mellomliggende verdiene nøyaktig selv om store deler av den originale kurven mangler. Predikeringsalgoritmen baserer seg derfor på å finne M1, M og M2 ut fra data som allerede er funnet. I Dopplermålinger hvor noen spektrum er fullstendige, mens andre er mangelfulle kan M1, M2 og M fra de fullstendige spektrumene brukes i de ufullstendige. I spektrum hvor dette ikke er mulig må punktene predikeres, og algoritmen er basert på å trenes opp til å finne punktene.

Data samlet på 20 manuelle spinger viser at forholdet mellom høyde M1 og høyde M ($\frac{M1}{M}$) er 87% med et standardvvik σ lik 3.9, og forholdet mellom M2 og M ($\frac{M2}{M}$) er 85% med σ lik 4.4. Dette standardavviket viser at disse forholdstallene holder seg noenlunde konstant. For hver gang algoritmen kjøres på et klart og tydelig Dopplerspektrum, vil dette forholdstallet trenes, til man etterhvert har et klart forholdstall mellom M1/M2 og M. Denne algoritmen baserer seg dermed på at en spline klarer å interpolere så godt som i figuren ovenfor og at punktene kan predikeres nøyaktig. Det beste er derfor om man kan benytte klare Dopplerspektrum fra samme måling, og erstatte M1, M2 og M fra disse.



Figur 3.17: Figur som viser markeringer ved henholdsvis 25, 50 og 75% av lengden fra A til E

En slik generell modell vil aldri gi helt riktige verdier, men det vil ingen modelltilpasninger kunne gi. Modellen som ble brukt beskrives som følger; Punkt C og D's horisontale plassering i forhold til M samt deres høyde blir lagret. Det punktet av C og D som er nærmest M brukes videre. Ligger for eksempel C nærmere M enn M1 vil M få en høyde gitt av formel 3.1:

$$M_{h\ddot{o}yde} = C_{h\ddot{o}yde} * \frac{1}{\frac{M1_{h\ddot{o}yde}}{M_{h\ddot{o}yde}} + \frac{C-M1}{M-M1}} \quad (3.1)$$

Hvor M, C og M1 er horisontale posisjoner. Verdien til nevneren blir et tall mellom $\frac{M1_{h\ddot{o}yde}}{M_{h\ddot{o}yde}}$ og 1. Om D hadde vært nærmere M enn C, ville beregningene blitt gjort med utgangspunkt i D, fordi vi da må predikere mindre. Formel 3.1 måtte da forandres til:

$$M_{h\ddot{o}yde} = D_{h\ddot{o}yde} * \frac{1}{\frac{M2_{h\ddot{o}yde}}{M_{h\ddot{o}yde}} + \frac{M2-D}{M2-M}} \quad (3.2)$$

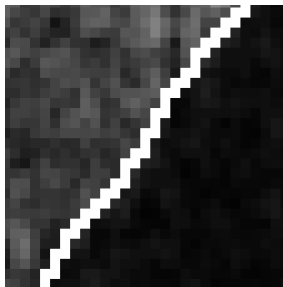
M2, eller eventuelt M1 får høyden $\frac{(M1\text{eller}M2)_{h\ddot{o}yde}}{M_{h\ddot{o}yde}} * M_{h\ddot{o}yde}$. Segmentet C-M1, M1-M, M-M2 og M2-D interpoleres så med en spline. $\frac{M1_{h\ddot{o}yde}}{M_{h\ddot{o}yde}}$ og $\frac{M2_{h\ddot{o}yde}}{M_{h\ddot{o}yde}}$ trenes for hver gang algoritmen kjøres på et klart og tydelig Dopplerspektrum

En svakhet ved metoden dukker opp når punkt C eller D ligger langt fra M1 eller M2. Ligger C før M1 må høyden til M1 predikeres på et vis. Dette er også gjort med utgangspunkt i studering av manuelle sporinger. Med utgangspunkt i posisjonen til C(eventuelt D), finnes M1/M2 basert på et gjennomsnittlig stigningstall på 0.6.

Predikeringen kan forenkles mye om man har fullstendige spektrum fra samme måling. Da kan man benytte informasjon fra disse for å finne høyden ved M1, M og M2.

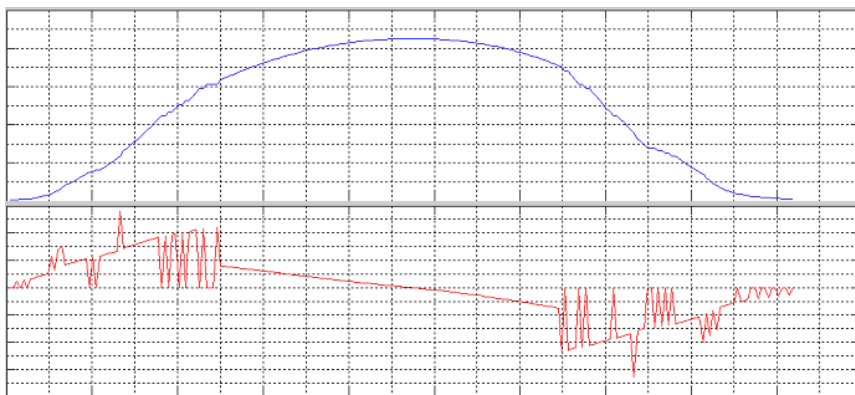
3.5.4 Utjevning

Den endelige kanten er nå behandlet for artefakter og eventuelt predikert. Det siste steget i prosessen er da å utjevne kanten for å fjerne punkt-til-punkt variasjonene slik at kurven kan differensieres for videre utregninger. Dette gjøres med et enkelt lavpassfilter, og i Matlab er dette mulig å gjøre med funksjonen *smooth*. Denne metoden benytter et såkalt “moving average“-filter av valgfri lengde. På grunn av den diskrete oppløsning til spektraldataen, er også kanten diskret. Det er alltid et heltall antall piksler i forskjell mellom to sideliggende punkt på kanten som man ser i figur 3.18.



Figur 3.18

Skal man differensiere denne kanten vil man få en kurve med store diskrete variasjoner. Figur 3.19 viser et eksempel på en uglattet kant og dens differensierte. Merk her at kanten nå er snudd på hodet, hvor toppunktet er øverst.

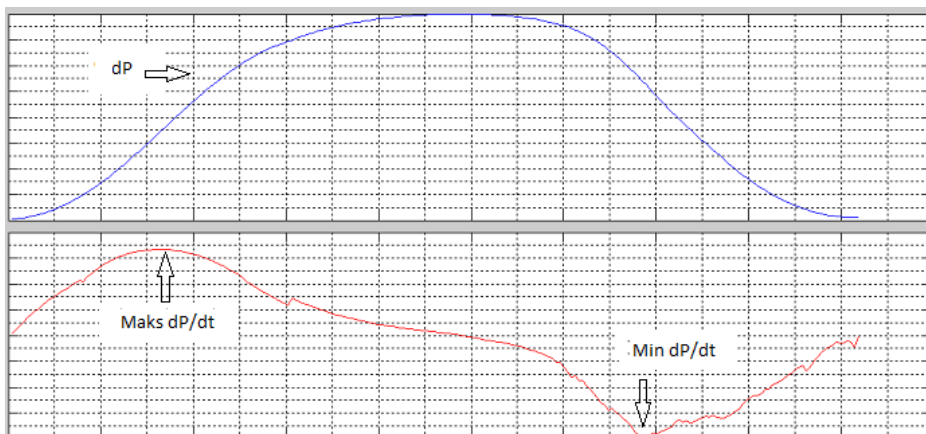


Figur 3.19: Ikke utjevnet trykkgradientskurve og dens differensierte

Denne kurven må derfor lavpassfiltreres til den er glatt. Samtidig så må man være oppmerksom på at for mye filtrering vil potensielt forandre hele kanten, slik at man underestimerer den virkelige stigningen til kanten. Lengden på filteret må derfor velges akkurat langt nok til at variasjonene mellom sideliggende punkt fjernes. Etter at kanten er glattet ut velges et antall av punktene langs kanten, og

det interpoleres mellom disse. Variasjoner i mellomrommet mellom disse punktene viser seg å forandre de utregnede verdiene minimalt, og kan derfor velges relativt vilkårlig, for eksempel med et mellomrom på 10.

Figur 3.20 viser en glattet/filtrert kant og dens differensierte. Forskjellen fra figur 3.19, hvor kanten følger diskrete steg, er betydelig. Merk at både i figur 3.19 og 3.20 er det trykkgradientskurven og ikke hastighetskurven man ser.



Figur 3.20: Utjevnet trykkgradientskurve og dens differensierte

En god indikator på at kurven er godt nok utjevnet er at den differensierte kurven er støyfri. “Moving average“-filteret som er benyttet har derfor en lengde på 10, og er valgt akkurat slik at at den differensierte kurven blir glatt nok, men samtidig ikke så lang at den utjevnete kurven “kutter“ naturlige svinger.

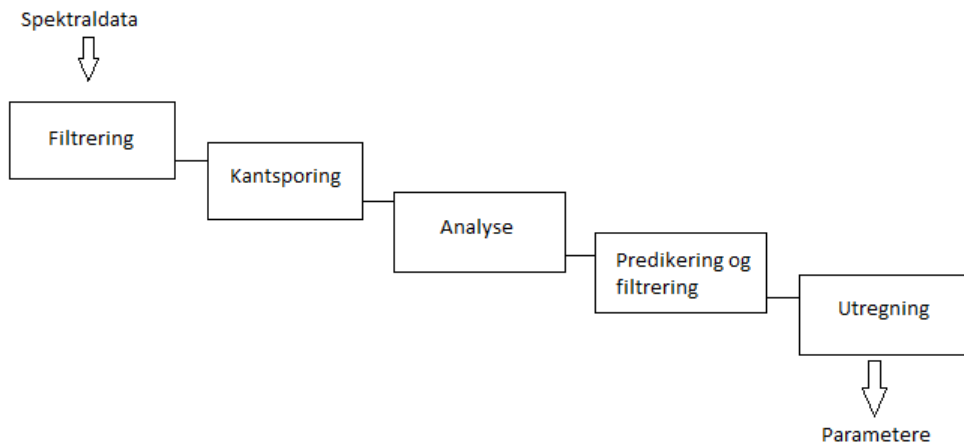
3.5.5 Utregninger

Etter at kanten er etterarbeidet, interpolert og utjevnet kan man regne ut ønskede verdier og utregningene av noen av de viktigste verdiene er allerede implementert i GcLab. Metoden *calculate* transformerer hastighetskurven til en trykkgradientkurve med Bernoulli's likning, og differensierer denne. Kurven er lagret i tabellen *data*, og differensieringen foretas med formelen $\frac{data(i+1)-data(i)}{tid}$. Metoden *dpdt* konverterer koordinater i bildet til tilsvarende tid og hastighet. Via GcLab kan man som tidligere nevnt hente ut den totale tiden som spektraldataen spenner over samt maksimal og minimal hastighet, og dermed kan man hente ut informasjon om tid og hastighet til et hvert punkt i bildet.

Verdier som utregnes er maksimal trykkgradient(dP), maksimal og minimal dP/dt (differensiert) samt relaksasjonskonstanten τ . Også RPR regnes ut for sammenligning, den midlede dP/dt mellom 1 og 3 m/s. Andre aktuelle verdier som er ønskelig å regne ut kan lett implementeres i metoden *calculate*.

3.6 Oversikt over algoritmen

En oversikt over algoritmen følger under.



Figur 3.21: Oversikt over hele prosessen fra filtrering til utregning av parametere

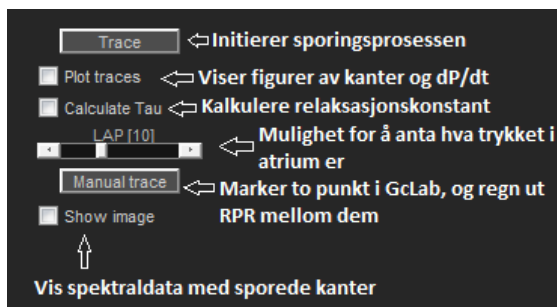
Spektraldataen hentes ut i form av et bilde. Denne blir videre filtrert og gjort om til et binærbilde som viser kantene i bildet. Kantene spores og lagres deretter i en tabell for videre prosessering.

Kanten man nå har ekstrahert fra spektraldataen er ikke klar for å gjøre utregninger på og må derfor prosesseres i etterkant. En av årsakene er at spektraldataen viser hastigheten til ikke bare blodstrømmen man vil måle, men også hastigheten til for eksempel klaffer. Samtidig, i spektraldata av dårlig kvalitet, er det ofte vanskelig å detektere hvor den egentlige kanten går. Noen ganger vises den ikke på spektraldataen heller, selv om vi vet fra et fysiologisk perspektiv at den skal være der. En annen årsak til at vi må etterprosessere kanten er kanten vi har fått fra filtreringen er full av støy, og da nærmere bestemt en form for kvantiseringstøy; fordi spektraldataen har en begrenset oppløsning vil verdiene til hvert punkt på kanten variere med et heltall antall piksler mellom hverandre.

Av disse årsakene må bildet først og fremst analyseres, for å se om noe må predikeres eller interpoleres, og deretter lavpassfiltreres/utjevnes. Analysedelen er derfor en vital del av algoritmen, og er basert på å markere viktige punkter langs den sporede kanten. De markerte punktene representerer skiller mellom separate segment langs kanten, som alle må prosesseres forskjellig ut fra forskjellige behov.

Med den ferdig behandlede kanten, kan beregninger utføres, og resultatet er en rekke parametere som eventuelt legen kan bruke videre i sin diagnose.

Algoritmen er implementert i GcLab i en klasse med navn “Mode_CW_trace“. Nedenfor vil alle metodene i denne klassen beskrives.



I det man kjører programmet leses det inn en rekke data fra DICOM-filene, inklusiv informasjon om hastighet, tid, EKG-data med mer. Spektraldataen deles også opp i segment, fra start til slutt på en hjertesykklus.

filterImage()

Tar inn spektraldata, og utfører de nødvendige filtreringene og bildebehandlingene. Metoden returnerer et binært bilde, hvor de detekterte kantene i bildet er hvite streker på en svart bakgrunn.

extractStructures()

Denne metoden søker gjennom bildet til den møter på en hvit piksel. En hvit piksel symboliserer en kant, og koordinaten sendes til metoden snake().

snake()

Får inn koordinaten til en piksel som er en del av en kant. Metoden hopper til naboen til denne pikselen, farger denne svart, lagrer posisjonen, og hopper videre til alle piksler i kanten er funnet. Hvis denne metoden stopper opp fordi den har møtt en “blindvei“, må metoden prøve å søke seg tilbake slik at den kan fortsette å spore kanten. Metoden må også sørge for å forkaste kanter som er artefakter grunnet bakgrunnstøyen i bildet.

calculate()

Den lagrede kanten fra snake() tas inn, og denne metoden finpusser den. Etter at metoden calculate() er kjørt er informasjon om høyde, lengde og midtpunkt på kanten med mer lagret. Kanten er også forberedt slik at den kun inneholder et punkt per horisontale koordinat.

prepareShape()

Denne metoden tar seg av analysen av kanten, og avgjør om kanten må predikeres, om algoritmen skal fjerne hastigheter fra klaffer etc. og eventuelt om hele kanten skal forkastes. Etter at analysen er gjennomført, vil `prepareShape()` utføre selve predikeringen, som innebærer interpolasjon, ekstrapolasjon og modelltilpasning.

calculate2()

Den ferdig preparerte kanten tas inn. Metoden går gjennom hvert punkt på kanten og konverterer koordinaten til en hastighet med metoden `dpdt()`. Med bernoulli's likning konverteres hastigheten til en trykkgradient. Resultatet er en trykkgradientkurve som utvikler seg over tid. Kurven differensieres og maksimal positiv og negativ differensiert lagres (maksimal dP/dt og minimal dP/dt), sammen med andre parametere. I denne metoden er det også støtte for å regne ut τ og andre parametere man vil finne. Metoden som er brukt for å regne ut τ er hentet fra [8] og [15], hvor man benytter lineær regresjon for å finne den lineære linjen som passer best med nedstigende segment, fra punktet hvor dP/dt er minst til punktet hvor trykket er 0. τ er den inverse av stigningstallet til denne linjen.

dpdt()

Metoden brukes for å konvertere x og y-aksen i spektralbildet til tids- og hastighetsakser. Metoden tar inn en eller to koordinater, regner ut ved hvilke hastigheter de ligger og tiden mellom dem, konverterer hastigheten til trykk, og returnerer dP/dt . Metoden returner eventuelt kun en hastighet hvis innputt kun er en koordinat.

manualTracing()

Denne metoden tar som input to koordinater som er pekt ut i GcLab av brukeren og benytter metoden `dpdt()` for å regne ut dP/dt mellom dem. Denne metoden gir brukeren muligheter for enkelt å via GcLab peke på et punkt på spekteret og se ved hvilken hastighet punktet er, samt enkelt regne ut dP/dt mellom to punkt. GcLab-vinduet hvor en fremvisning av spekteret er, kan justeres

Kapittel 4

Resultater

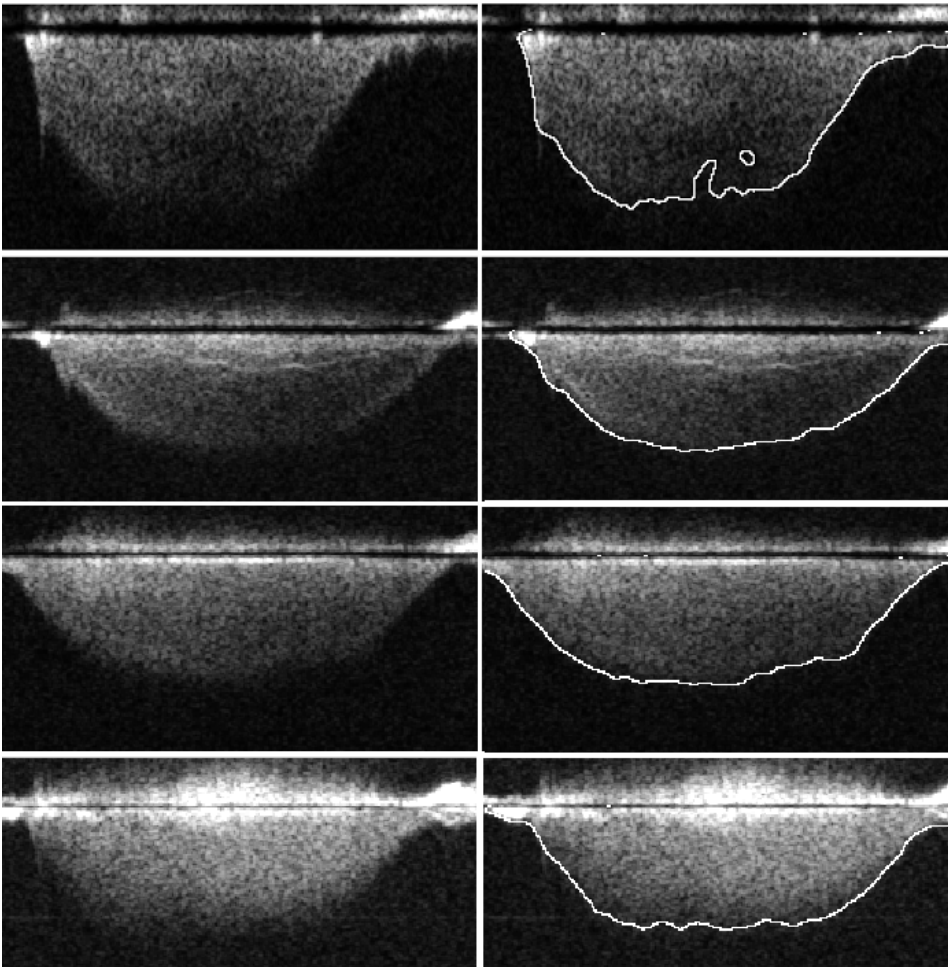
I denne seksjonen vil resultatene presenteres. DICOM-filer er benyttet for testing av algoritmen. DICOM-filene inneholder en mengde informasjon om pasienten, i tillegg til blant annet spektraldata og 2D-ultralyd. Disse filene inneholder Dopplerdata av varierende kvalitet, og dermed er algoritmen blitt utprøvd på en rekke forskjellige typer bilder.

Kapittelet er delt opp i 4 deler, hvor den første viser hvordan kantdeteksjonen fungerer på bilder av varierende kvalitet. Den neste delen viser hvordan predikering kan lage fullstendige sporinger selv på bilder av dårlig kvalitet. De predikerte resultatene vil også sammenlignes med de upredikerte. I den tredje delen demonstreres en sammenligning mellom manuelt sporet data og den automatiske sporingsalgoritmen, hvor den manuelle sporingen er utført av en lege. Forskjellen mellom dem er presentert i form av en tabell som viser den gjennomsnittlige forskjellen mellom utvalgte parametere utregnet fra sporingen. Den siste delen tar blant annet for seg tidsbruken til algoritmen.

Resultatene er stort sett presentert i form av bilder som viser sporingene. Å vurdere kantdeteksjon er i høy grad mulig å inspisere visuelt. Å vurdere korrektheten av verdiene man har utregnet kan derimot kun verifiseres gjennom invasive målinger i hjertet med kateter. Dette har dessverre ikke vært mulig å få gjennomført i løpet av prosjektet, men nøyaktigheten til verdiene man kan oppnå med Dopplermålinger kontra katetermålinger er fra før verifisert i eksempelvis [8, 13, 15, 22]. Fokus på verifisering av metoden er derfor gjort med utgangspunkt i en sammenligning med manuelt sporede kanter.

4.1 Kantsporing

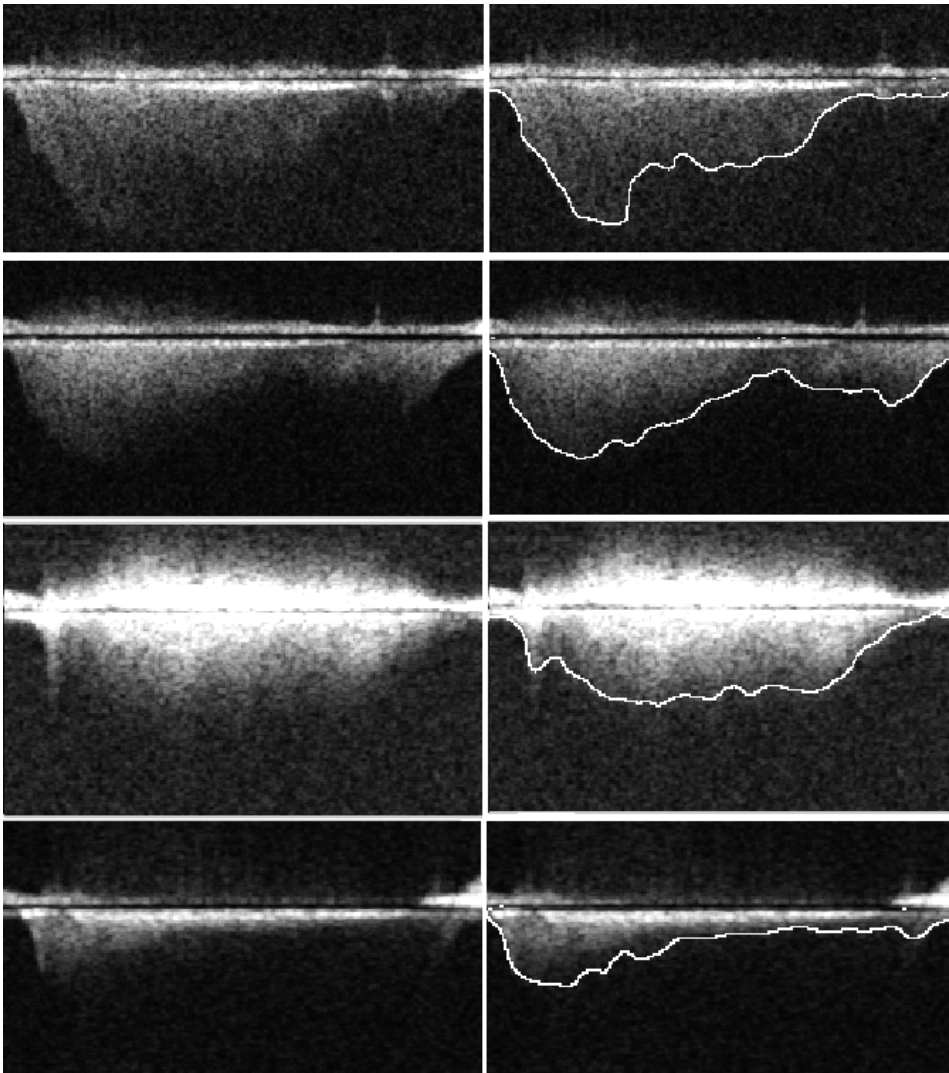
I denne seksjonen presenteres en del bilder før og etter kantsporingen. Resultatene her viser tydelig at mange spekter krever ytterligere behandling før de kan benyttes i utregninger. Eksemplene resresenterer spektrere av forskjellig kvalitet, alt fra veldig tydelige spektrere, til de som er meget utydelige og som i noen tilfeller er ansett som umulige å spore.



Figur 4.1: Eksempler på kantsporing av klare og tydelige spektrere

Figur 4.1 viser at kantdeteksjonen fungerer meget godt på spektrere av klar og tydelig karakter. Samtidig viser disse resultatene påvirkningen en klaff vil gi på et Dopplerspektrum. Eksemplene viser også at kanten er tydelig ujevn og inneholder enkelte artifakter.

Figur 4.2 viser noen eksempler på Dopplerspektrum av dårlig kvalitet og viser den store forskjellen fra spektrumene i figur 4.1.

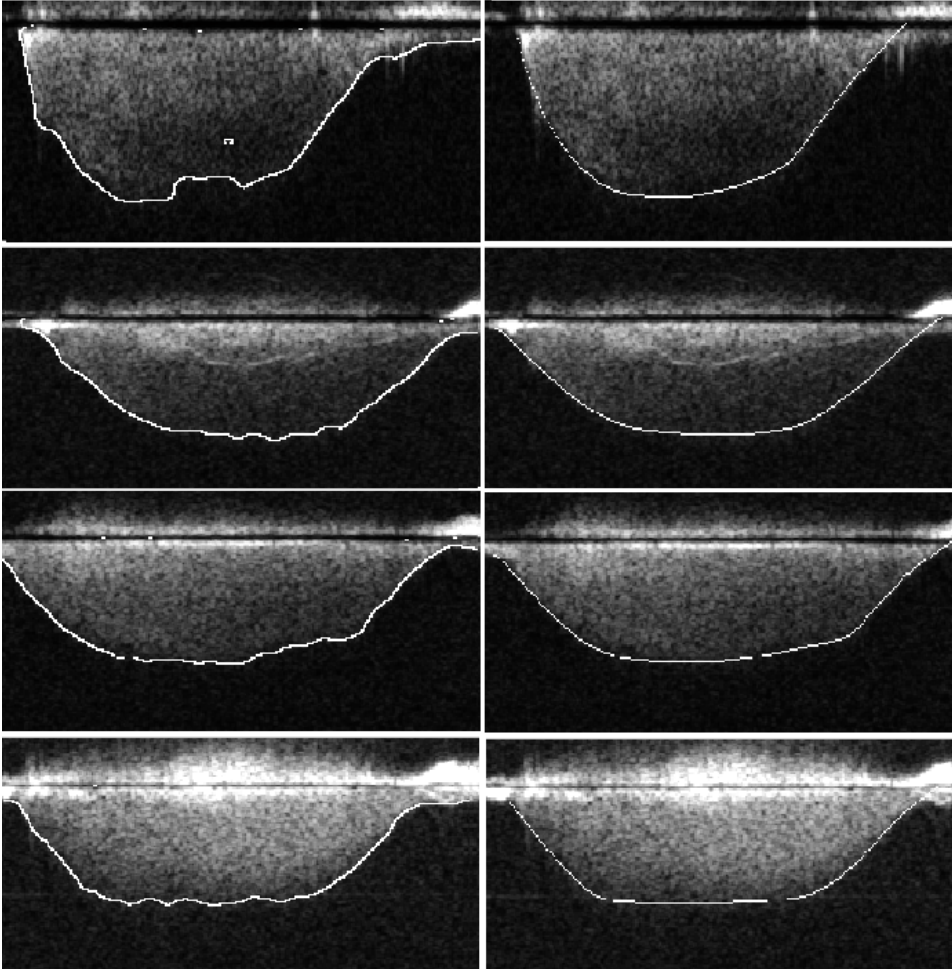


Figur 4.2: Eksempler på kantsporing av Dopplerspektrum av dårlig kvalitet

Disse eksemplene viser tydelig hvordan noen Dopplerspektrum har veldig dårlig kvalitet. Samtidig viser disse resultatene at kantsporingen fungerer som den skal, og sporer kantene der de er synlige i spekteret. Det nederste eksempelet i figur 4.2 demonstrerer et spektrum som algoritmen ville forkastet. Det nest nederste viser et spektrum som har veldig svakt definerte kanter i motsetning til de andre tilfellene. Dette eksempelet viser også tydelig påvirkningen av en klaff på oppstigende kurve.

4.2 Ferdig behandlede Dopplerspringer

Her presenteres de ferdig behandlede springene av spektrumene i figur 4.1 hvor ingen modeller er brukt. Figurene viser spektre som er kantdetektert, ekstrapolert og utjevnet.

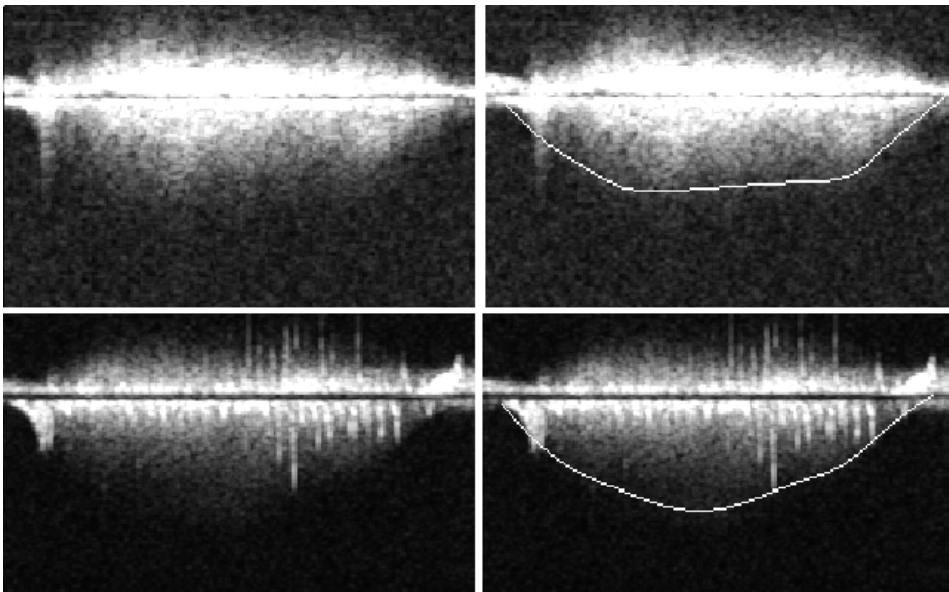


Figur 4.3: Ferdig behandlede springer

Resultatene viser at når Dopplerspektrumet er klart og tydelig gir algoritmen et godt og tydelig resultat. Den glatter effektivt ut oppstigende kontur hvor det opprinnelig var en kant på grunn av en detektert kant, og ekstrapolerer samtidig nedstignene kontur mot 0.

4.2.1 Sporing av spektrum av dårlig kvalitet

Denne seksjonen viser noen eksempler på sporinger av spekter med dårlig kvalitet uten bruk av modeller. Dårlig kvalitet vil først og fremst bety at kantene til spekteret er diffuse, og ikke klart definerte som i eksemplene i figur 4.4.

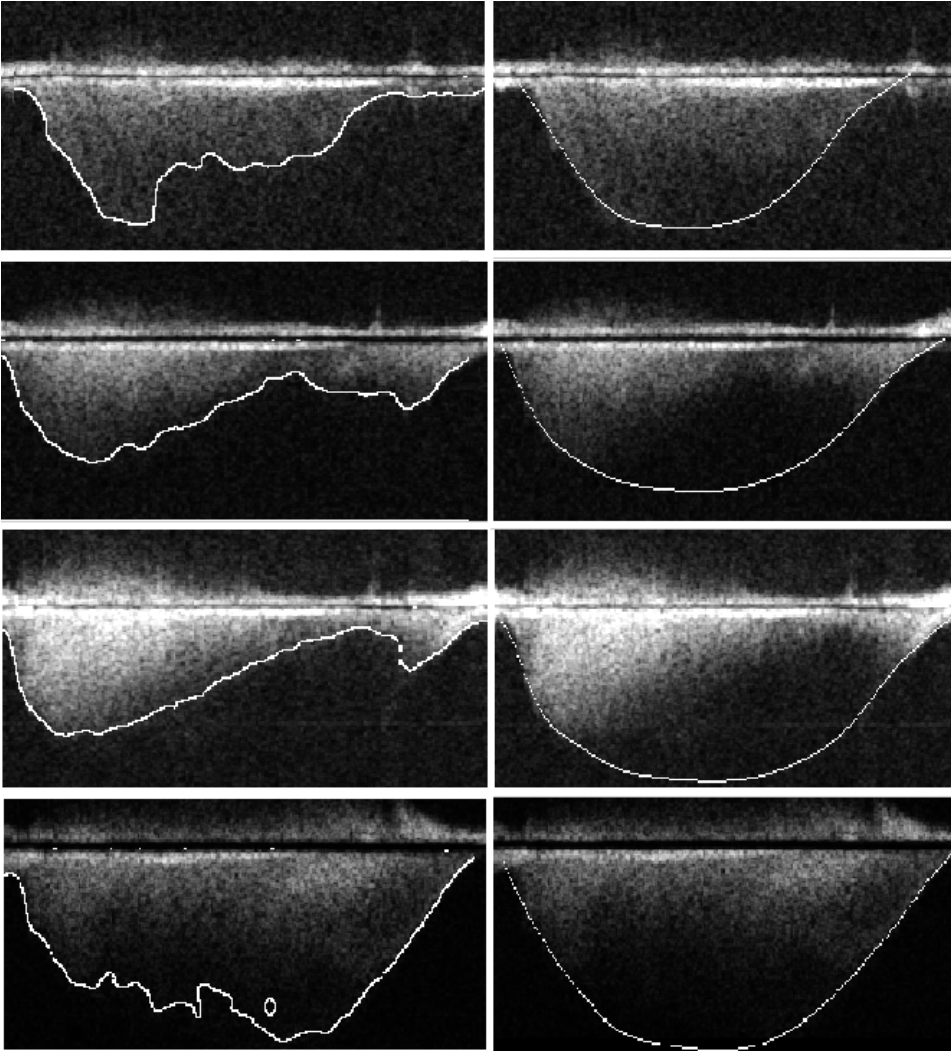


Figur 4.4: Sporinger av svake spektrum

Disse viser at kantdeteksjonen og den resulterende sporingen er robuste mot høye mengder av støy og svake kanter. Samtidig er det tydelig at toppen av sporingen antageligvis underestimerer den reelle kanten i slike typer bilder. En modell kan være løsningen i slike tilfeller, men vi vet at en modell ikke er annet enn en predikering og bør således unngås i så stor grad det er mulig.

Eksemplene viser at selv på spektrum med dårlig kvalitet og mye støy vil algoritmen være robust, mens den i enkelte tilfeller ikke fungerer like godt. Med så dårlige Dopplermålinger ville antageligvis en lege likevel forkaste slike bilder selv om ikke algoritmen gjør det.

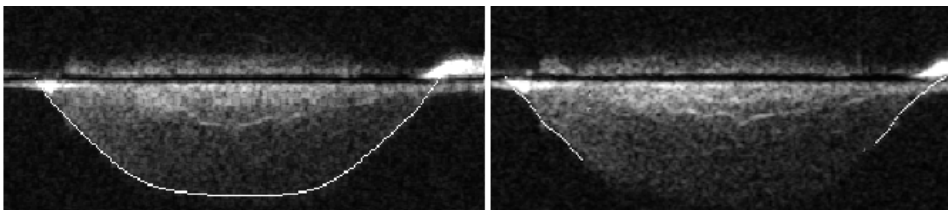
4.2.2 Bruk av modeller



Figur 4.5: Sporningsalgoritmen med bruk av modeller

Figur 4.5 viser fire eksempler på Dopplerspektrum hvor en modell er benyttet for å predikere de mangelfulle områdene i spekteret. Disse resultatene viser at ved å benytte en modell så bevarer de nøyaktig detekterte kantene samtidig som man predikerer resten av spektrumet hvor kanten ikke er synlig. De to øverste spektrumene er de samme som de to øverste i figur 4.2.

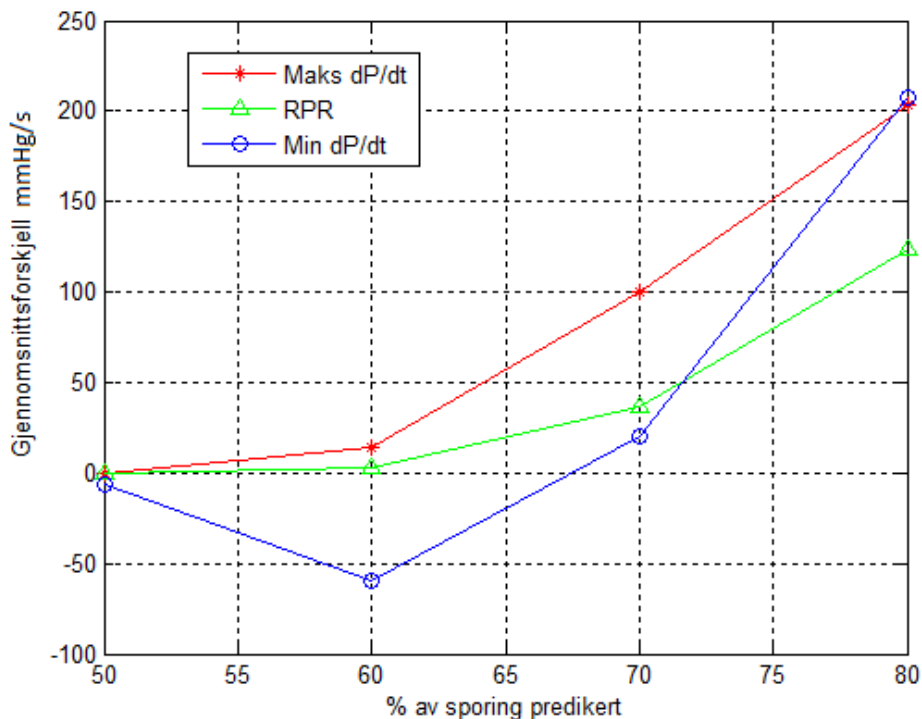
For å teste hvor godt modellen fungerte ble 15 tydelige sporede spektrum med klare kanter valgt. Deretter ble toppen av disse sporingene fjernet (se figur 4.6) og erstattet med en modellpredikering.



Figur 4.6: Sporet kant hvor toppen er fjernet

Deretter ble verdiene regnet ut med og uten predikering sammenlignet. Sammenligningen ble gjort med henholdsvis 50, 60, 70 og 80% av sporingen fjernet.

Figur 4.7 viser den gjennomsnittlige forskjellen mellom utregnet RPR samt maksimal og minimal dP/dt .

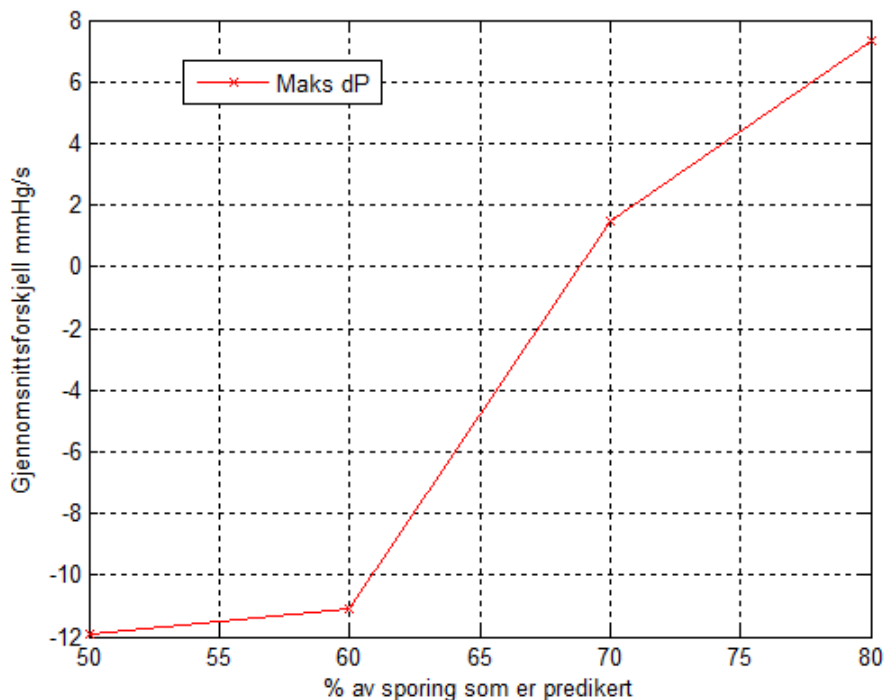


Figur 4.7: Gjennomsnittlig forskjell mellom verdier utregnet fra predikerte og ikke predikerte spektrum

Positive verdier betyr at modellen underestimerer verdiene. Ved for eksempel

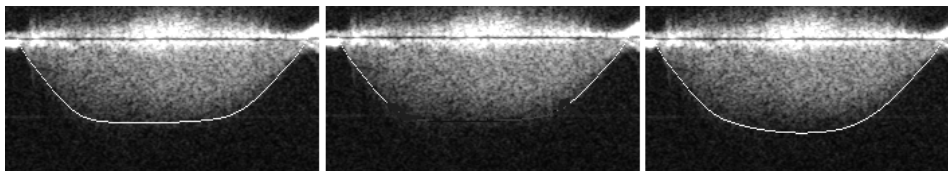
80% ser man at modellen underestimerer maksimal dP/dt med gjennomsnittlig 200 mmHg/s. Disse resultatene viser at når mindre enn 60% av spektrumet er predikert er feilen veldig lav og indikerer at modellen fungerer veldig godt. Med veldig få unntak underestimerer modellen verdiene RPR og maksimal dP/dt .

I figur 4.8 er en sammenligning av maksimal dP . Modellen går fra å underestimere dP ved mye predikering til å overestimere ved lite predikering.



Figur 4.8

Figur 4.9 viser et eksempel på et spektrum hvor 60% av toppen av sporingen er fjernet, for å så bli predikert. Man ser at toppunktet er høyere med predikering, men at spektrumet ligner mer på et typisk Dopplerspektrum for mitralinsuffisienser.

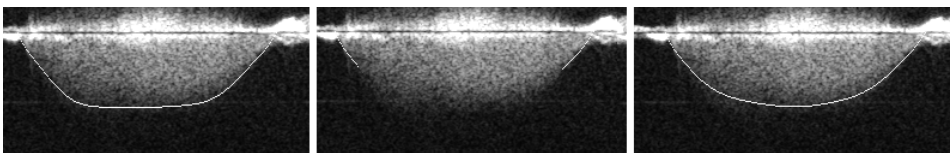


Figur 4.9: Sporing med og uten 60% fjernede kanter samt den predikerte

Dette viser et eksempel på hvorfor maksimal dP i noen tilfeller estimeres an-

nerledes enn den originale sporingen, og det er fordi den originale sporingen ikke har form som et typisk Dopplerspektrum.

Figur 4.10 viser samme eksempel hvor 80% av toppen av sporingen er fjernet, for å så bli predikert.

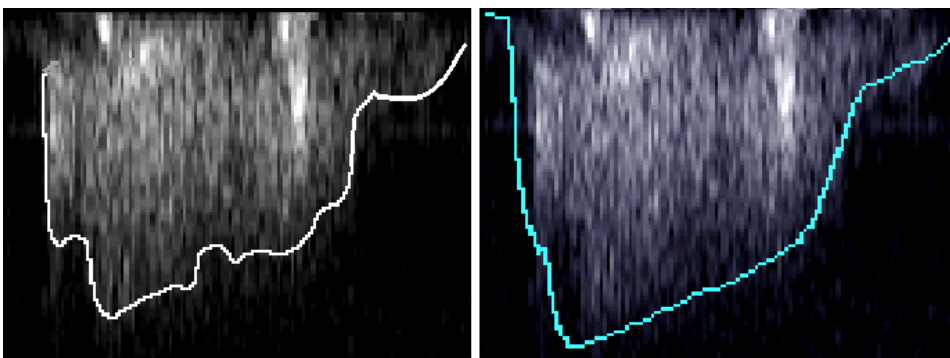


Figur 4.10: Sporing med og uten 80% fjernede kanter samt den predikerte

4.3 Sammenligninger

4.3.1 Andre metoder

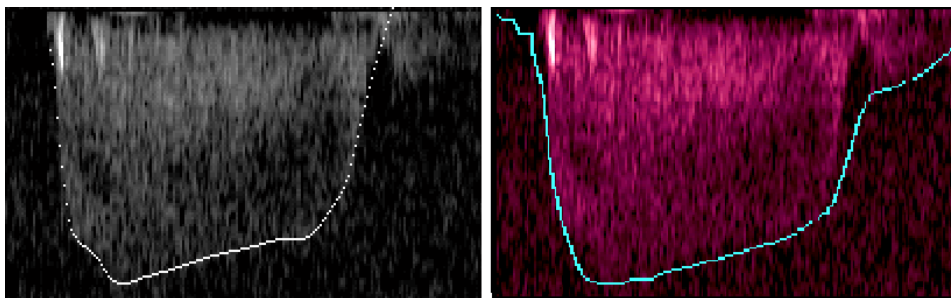
Algoritmen er ment å være et alternativ til andre algoritmer og målsetningen har vært å forbedre svakhetene i disse. Figur 4.11 og 4.12 viser en sammenligning mellom den foreslåtte algoritmen og den i [3]. Det har ikke vært mulig å oppdrive datamaterialet som ble benyttet i hverken denne, eller i [2], [4] og [5]. Bildene er derfor tatt direkte fra den publiserte artikkelen [3, p. 5], og er to forskjellige Dopplerspektrum fra en PW Doppler-måling.



Figur 4.11: Eksempel 1: Den foreslåtte metoden til venstre, den fra [3] til høyre

Figur 4.11 viser det ene eksempelet fra [3] sammenlignet med den foreslåtte metoden, hvor kun kantdeteksjon er utført. Denne sammenligningen viser at den foreslåtte algoritmen forbedrer oppstigende kontur, men samtidig underestimerer toppunktet.

Figur 4.12 viser et annet eksempel fra artikkelen sammenlignet med den aktuelle metoden, denne gangen med etterbehandling og utjevning. Resultatet her viser at også i dette eksempelet klarer den foreslåtte algoritmen å finne opp- og



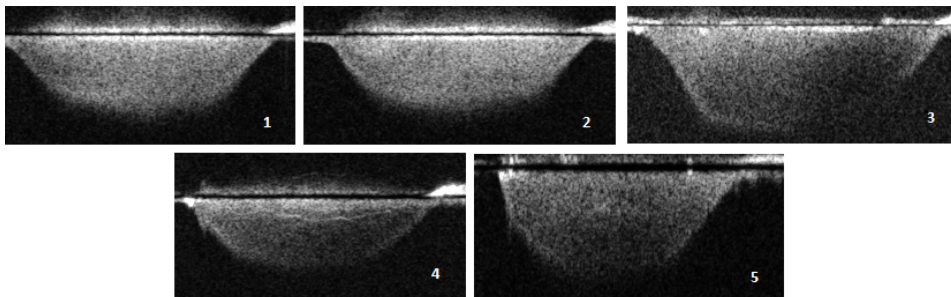
Figur 4.12: Eksempel 2: Den foreslåtte metoden til venstre, den fra [3] til høyre

nedstigende kontur mer effektivt. Ut i fra disse eksemplene er det tydelig at den foreslåtte algoritmen klarer å finne konturene i spektrene bedre enn algoritmen i [3]. Disse resultatene viser også at algoritmen fungerer med dopplerspektrum fra PW Doppler.

4.3.2 Sammenligning med manuelle springer

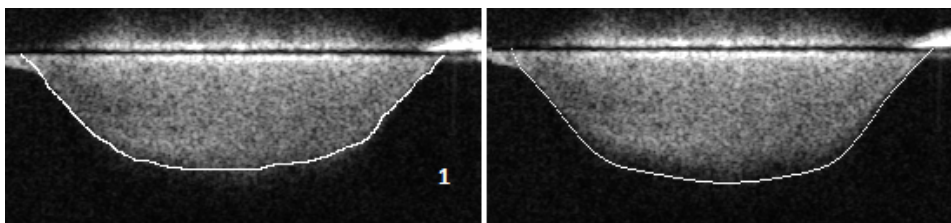
For å validere at algoritmen klarer å spore spektrene korrekt, burde ideelt de parameterene som kan utregnes fra sporingen, sammenlignes med verdier man har funnet ved å måle inne i hjertet. Korrelasjonen mellom parametere slik som maksimal dP/dt og τ som man finner med Dopplermålinger og de man måler inne i hjertet er grundig dokumentert i tidligere litteratur (kapittel 2.4) og har vist seg å være høy [8, 13, 14, 15]. Manuelle sporinger utført av en kvalifisert person er derfor satt som sammenligningsgrunnlag for algoritmen. Et utvalg av bilder er blitt sporet manuelt og automatisk og sammenlignet. Videre følger noen bilder som viser manuell (til venstre) kontra automatisk (til høyre) sporing, en sammenligning mellom spektrenes maksimale dP/dt er også tatt med.

Figur 4.13 viser bilder som er blitt sammenlignet.



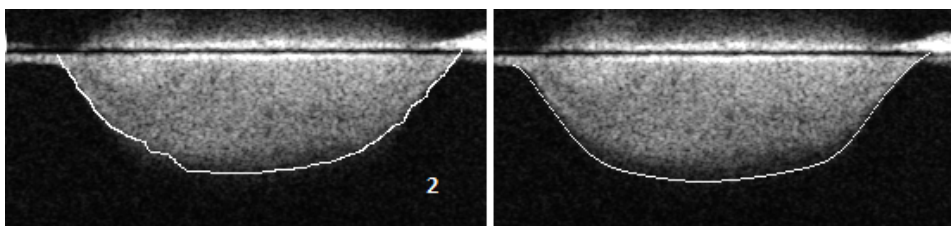
Figur 4.13: Utvalg av bilder som er blitt sammenlignet

Figur 4.14 viser den manuelle og automatiske sporingen av bilde nr. 1. Dette spekteret har ingen klart definert kant på toppen, og det er tydelig at algoritmen



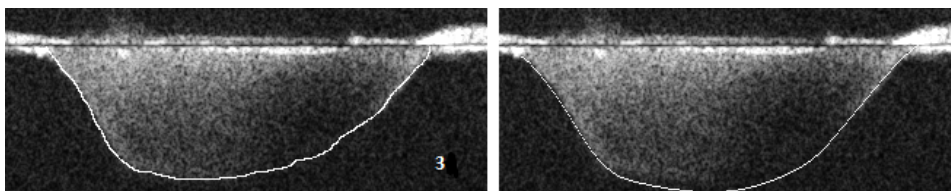
Figur 4.14: maks dP/dt : 987 mmHg/s vs. 1401 mmHg/s

antar at kanten ligger høyere enn hva den manuelle sporingen gjør. Figur 4.15 viser et annet bilde med en uklar kant øverst, men her er forskjellen mellom den manuelle sporingen og den automatiske mindre.



Figur 4.15: maks dP/dt : 834 mmHg/s vs. 1331 mmHg/s

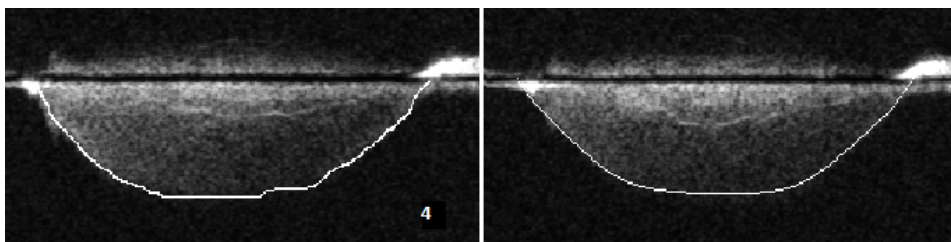
Både figur 4.14 og 4.15 viser likevel tydelig at algoritmen sporer opp- og nedstigende kontur godt.



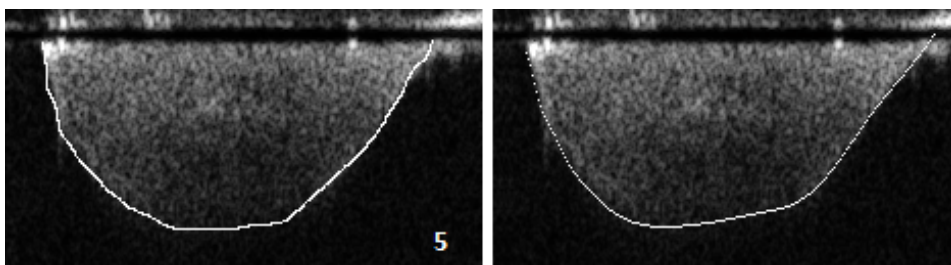
Figur 4.16: maks dP/dt : 1170 mmHg/s vs. 1213 mmHg/s

Denne figuren viser en sammenligning hvor algoritmen har benyttet en modell for å predikere det mørke segmentet i spekteret. Resultatet er at toppunktet og nedstigende kontur overestimeres, men at sporingen ligner mer på et typisk Dopplerspektrum enn den manuelle sporingen.

Figur 4.17 viser et Dopplerspektrum hvor algoritmen har sporet kantene godt, men gir en litt svakere stigning i starten og på slutten enn den manuelle sporingen.



Figur 4.17: maks dP/dt: 623 mmHg/s vs. 717 mmHg/s



Figur 4.18: maks dP/dt: 1153 mmHg/s vs. 1037 mmHg/s

Figur 4.18 viser at den automatiske sporingen gir en glatt og fin kant. Man ser likevel en viss forskjell fra den manuelle sporingen, og viser hvordan en bias, forhåndskunnskap om formen på Dopplerspektrum, påvirker den manuelle sporingen.

Tabell 4.1 viser snittforskjellen $\mu_{forskjell}$ mellom parametere som er utregnet fra manuelt og automatisk sporede kanter. 12 bilder er sammenlignet, og det viste seg at den automatiske sporingen har en tendens til å overestimere både maksimal dP/dt, maksimal hastighet og RPR, mens den underestimerer minimal dP/dt.

	maks dP/dt	maks hastighet	RPR	min dP/dt
$\mu_{forskjell}$	96 mmHg/s	0.03 m/s	53 mmHg/s	85 mmHg/s

Tabell 4.1: Forskjell mellom manuell og automatisk sporing

Forskjellen varierer mellom bildene, men enkelte spektrum ble grovt overestimert, som for eksempel spektrumet i eksempel 1 (figur 4.14). Her ble dP/dt overestimert med hele 400 mmHg/s. Dette er fordi spektrumet hadde en diffus kant øverst, hvor de to sporingene var uenige om hvor kanten gikk. Dette demonstrerer også hvordan relativt små forandringer i sporingen kan påvirke dP/dt i høy grad. For bilder med klare kanter som i figur 4.17 og 4.18 er forskjellen betydelig mindre, og varierer grovt sett med ± 100 mmHg/s, med et standardavvik σ på 79. Dette standardavviket betyr at for de fleste spektrumene vil feilen ligge innenfor 78 mmHg/s. Forskjellen i RPR er enda mindre, og ligger for mange av bildene under

20 mmHg/s. Tatt i betraktning hvor lite som skal til for å påvirke dP/dt er disse forskjellene relativt små og veldig lovende.

4.4 Øvrige resultat

4.4.1 Variasjon i utregningene

Et annet viktig moment i valideringen av algoritmen kan være å se på reliabiliteten. Gitt en rekke hjertesykluser fra samme Dopplermåling vet vi at gitte parametere av hjertefunksjonen vil holde seg noenlunde stabile over tid, eventuelt gradvis øke eller synke i verdi. Tabell 4.2 viser derfor det gjennomsnittlige standardavviket σ til de samme parameterene som i tabell 4.1. Store avvik mellom hjertesykluserne i en Dopplermåling kan tyde på at algoritmen regner ut feil verdier.

	maks dP/dt	maks trykk	RPR	min dP/dt
σ	78 mmHg/s	4.67 mmHg	62 mmHg/s	107 mmHg/s
$\mu_{forskjell}$	130 mmHg/s	7.14 mmHg	104 mmHg/s	189 mmHg/s

Tabell 4.2: Forskjeller mellom verdier utregnet fra hjertesykluser fra samme måling(10 målinger)

$\mu_{forskjell}$ er den gjennomsnittlige forskjellen mellom maksimal og minimal verdi fra alle syklusene i en Dopplermåling. En snittforskjell på 130 mmHg/s mellom syklusene fra samme måling er et relativt lavt tall. Standardavviket viser det gjennomsnittlige standardavviket til verdiene fra målingene. Dette tallet viser at verdiene mellom hjertesykluser i en måling stort sett er stabile, og at forskjellen for eksempelvis maksimal dP/dt stort sett ligger mellom 0 og 78 mmHg/s.

4.4.2 Tidsbruk

Tidsbruken kan være avgjørende. Denne algoritmen er ikke beregnet for sanntidsbehandling, men tidsbruk sier likevel mye om algoritmen. Tidsbruken til algoritmen er avhengig av støy-til-signalforholdet og antall hjertesykluser i bildet. Desto mer støy desto flere artifakter i bildet som må behandles. Tabell 4.3 viser gjennomsnittlig tidsbruk for algoritmen under filtrering, i forhold til antall hjertesykluser i spekteret og ved modellbasert predikering.

	Tid
Filtrering	0.44 s
Tid per hjertesyklus uten modellpredikering	1.36 s
Tid per hjertesyklus med modellpredikering	1.32 s

Tabell 4.3

Kapittel 5

Diskusjon

5.1 Resultater

Resultatene viste en rekke eksempler på spektrum sporet automatisk med den foreslåtte algoritmen. I forhold til vurderingen av hvor godt algoritmen fungerer må man ha klart for seg skillet mellom klare og tydelige spektrum og svakere spektrum med diffuse og mangelfulle kanter. Noen spektrum har delvis klare kanter, og delvis svake/manglende kanter. Støynivået i spektrumene varierer også. De spektrumene med svake kanter kjennetegnes ved et manglende klart skille mellom blodstrømmen og bakgrunnen og hvor intensiteten gradvis blir svakere. Slike spektrum ville normalt blitt forkastet ved en undersøkelse, fordi det er vanskelig å definere hvor konturen til hastigheten går. Eventuelle verdier som må regnes ut er da heller ikke pålitelige.

Resultatene viste også eksempler på sporinger av spektrum med klare konturer. Visuell inspeksjon av disse viste at den resulterende sporingen fulgte kanten av signalet godt og fjernet artifakter forårsaket av støy og andre ting enn blod. Deteksjon av kanter i spektrumet er kun en del oppgaven til algoritmen og resultatene demonstrerte viktigheten av etterbehandling av kantdeteksjonen, som blant annet innebærer glatting og ekstrapolering av kanten. Analysen av kanten er derfor en essensiell del av prosessen, blant annet for å avgjøre om spektrumet må forkastes eller om det må foretas videre predikering.

Predikering foretas på mangelfulle spektrum. Dette er spektrum hvor kantene stort sett er skarpe, men hvor deler av det er svakt. Det er ønskelig at algoritmen skal håndtere slike type spektrum, slik at man kan støtte seg til en predikering av disse når målingene er dårlige. Modellen som blir brukt for å erstatte de mangelfulle områdene er basert på å lære seg formen på et typisk Dopplerspektrum og deretter benytte denne for å predikere de mangelfulle områdene. Eventuelt kan predikering foretas med utgangspunkt i fullstendige spektrum fra samme måling da vi vet at formen på spektrumene varierer minimalt mellom etterfølgende hjertesykluser. Dette ble også vist i resultatene. Det bør være inneforstått at en predikering ikke innebærer en garanti for at man finner riktige verdier. I resultatene ble det gjort

en sammenligning mellom utvalgte verdier utregnet fra et klart spektrum og det samme spektrumet hvor deler av det ble predikert. Resultatene av denne sammenligningen viste tydelig hvordan maksimal og minimal dP/dt og RPR nøyaktig ble estimert ved predikering, avhengig av hvor mye av spekteret som måtte predikeres. Nøyaktigheten til andre verdier enn de som ble sammenlignet kan ikke garanteres.

For å verifisere nøyaktigheten til sporingen er det ikke nok å inspisere dem visuelt, og det er derfor utført en sammenligning mellom den automatiske sporingen og manuelle sporinger utført av en spesialist i hjertemedisin. Å regne ut verdier fra de manuelle sporingene innebærer en viss usikkerhet fordi disse ikke er utjevnet, og dette gjelder spesielt verdier som dP/dt . Sammenligninger viste også eksempler på spektrum hvor kantene enkelte steder var diffuse, og hvordan algoritmen da fant kanten et annet sted enn hva den manuelle sporingen gjorde. Dette gjorde også sitt utslag på verdiene som ble utregnet. For spektrum med klare kanter var forskjellen betydelig mindre og for maksimal dP/dt lå feilen stort sett fra ± 100 mmHg/s, og standardavviket var på 79 mmHg/s. Det må også tas med i betraktningen at verdiene som er regnet ut fra de manuelle sporingene har en viss usikkerhet ved seg siden disse sporingene ikke er utjevnet.

Alt i alt viser resultatene at den automatiske sporingen effektivt finner kanter, utjevner dem og finner tilnærmet korrekte verdier. Også den modellbaserte predikeringen fungerer effektivt, og gir riktige verdier for dP/dt så lenge ikke for mye av spektrumet må predikeres. Så lenge sporingen ikke utføres på utydelige spektrum fungerer den godt, men fordi noen spektrum er av en mer utydelig natur er det viktig at de utregnede verdiene verifiseres av personen som analyserer dem, da ved å visualisere sporingen.

Selve utarbeidelsen av algoritmen, og spesielt valg av filter, er i høy grad basert på empiri, men tiltak er innført for å forsikre seg om at algoritmen er robust mot varierende nivå av støy. Resultatet viser at i spektrum hvor kantene er greit synlige med det blotte øyet, fungerer algoritmen godt, selv med høye støynivå. Algoritmen er beregnet på å brukes på stillbilder, og er dermed ikke tilpasset for sanntidsmålinger. En algoritme som opererer i sanntid har uten tvil visse fordeler, men vil også vanskeliggjøre bruken av filter-og-terskel-teknikken denne algoritmen har benyttet seg av. I en tenkt situasjon hvor denne algoritmen skal implementeres på ultralydutstyr kan en delvis sanntids-basert versjon brukes, hvor skjermbilder fryses etterhvert som målingen foretas. Man arbeider da ikke punkt for punkt, men får den fordel at man enklere kan benytte modellen som algoritmen foreslår, og bruke verdier fra tidligere hjertesykluser i modellen.

Tidsbruken til algoritmen er spesielt viktig om den skal benyttes til nevnte formål. Tidsbruk er uansett relevant, og bør minimeres. Resultatene viser at tidsbruken er avhengig av antall hjertesykluser i spektrumet, og ligger litt over sekundet per syklus. Dette er en akseptabel tid, men kan med fordel gjøres enda mindre. Predikering ser heller ikke ut til å påvirke tidsbruken.

Oppgaveteksten sier at metoden skal være generelt rettet mot alle typer Doppplerspektrum, men spesifikt rettet mot Dopplerspektrum av mitralinsuffisienser. Den resulterende algoritmen vil fungerere for alle typer spektrum, fordi den består av filtrering og terskling. Det må små grep til for å tilpasse den til andre typer spekt-

rum. Modellen som er foreslått er tilpasset Dopplerspektrum fra mitralinsuffisier, og ønsker man modeller for andre typer spektrum må modeller lages for disse. Selve kantdeteksjonen og etterbehandlingen i algoritmen er likevel slik at den kan benyttes både for PW og CW Doppler, samt spektrum av forskjellige former.

I forhold til andre algoritmer viser resultatene at den foreslåtte algoritmen finner opp og nedstigende kontur meget effektivt.

5.2 Videre arbeid

Videre arbeid med algoritmen bør ta for seg modellen som er benyttet, og utbedre denne. For eksempel bør man studere manuelle sporinger for å bedre estimere hvor de tre punktene som modellen er bygget opp rundt befinner seg. Modeller bør også lages for andre typer Dopplerspektrum. For øyeblikket eksisterer det kun en modell for CW Dopplerspektrum av mitralinsuffisier, men hvis man ønsker at metoden skal være robust for andre typer spektrum bør modeller lages for disse også.

Algoritmen kan også effektiviseres i forhold til tidsbruk, slik at den kan brukes i sanntid. Hvis man kan redusere tiden metoden kjører på kan algoritmen brukes i sanntid på ultralydutstyr ved å kjøre metoden på et og et stillbilde om gangen.

For å verifisere at verdiene som algoritmen kan regne ut er riktig bør algoritmen testes på Dopplerspektrum fra målinger hvor det virkelige trykket i hjertet er målt med kateter samtidig. Først da kan man vite helt sikkert hvor nøyaktig algoritmen kan regne ut verdier.

Kapittel 6

Konklusjon

I denne masteroppgaven er det foreslått en algoritme for å spore konturene av Dopplerspektrum fra hjertet automatisk. Flere verdier kan beregnes ut fra et Dopplerspektrum som man ellers måtte inn i hjertet for å måle. Å ha muligheten til å finne disse uten en invasiv undersøkelse vil gjøre slike verdier mer tilgjengelig, kan bidra til bedre diagnostisering og være tids- og ressursbesparende. Algoritmen fungerer på alle typer Dopplerspektrum, men er spesifikt tilpasset Dopplerspektrum av mitralinsuffisienser. Metoden utføres på stillbilder, men ved å fryse Dopplerspektrum ved gitte tidspunkt kan en form for sanntidsbehandling implementeres.

Metoden består av en forbehandling og en etterbehandling av spektrumet. Forbehandlingen går ut på å filtrere og terskle spektrumet, slik at man sitter igjen med et svart-hvitt bilde. Resultatet av forbehandlingen er en rekke grove kanter, som også inkluderer artifakter grunnet støy i spektrumet. Etterbehandlingen av kantene fjerner artifakter og predikerer deler av kanten som er mangelfull.

Resultatene viser at metoden effektivt finner konturene av spektrumet og fjerner artifakter og støy langs kanten. Bruk av en modell gjør at man kan predikere områder av spektrumet hvor målingene har vært dårlige. Sammenligningen mellom utvalgte verdier fra manuell og automatisk sporede spektrum viser at metoden finner riktige verdier for dP/dt og maks hastighet så lenge kvaliteten er brukbar. Resultatene viste også at det er viktig at sporingen verifiseres før man aksepterer verdiene, spesielt for dP/dt , som er høyst sensitiv for variasjoner.

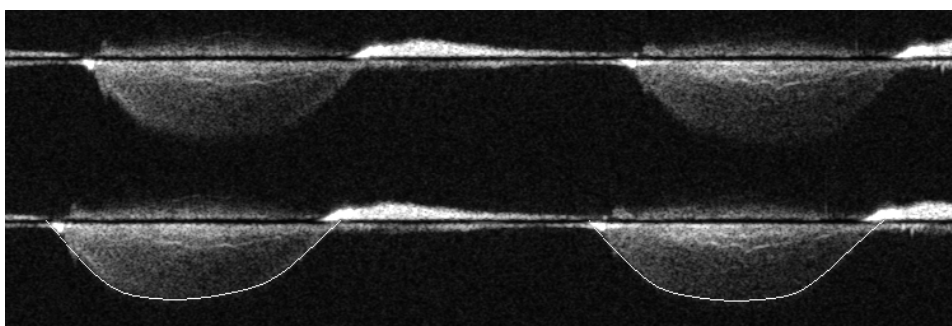
Den endelige algoritmen er en robust metode som ved bruk av en analyse av kanten kan fjerne artifakter forårsaket av klaffer og støy i bilder, og som kan predikere deler av spektrumet hvor målingene er svake. For spektrum av mitralinsuffisienser er det utviklet en modell som lærer seg formen på typiske spektrum etterhvert som algoritmen brukes. Resultatene viste at selv når 60% av Dopplerspektrumet må predikeres kan man nøyaktig finne verdier som maksimal og minimal dP/dt .

Programkode

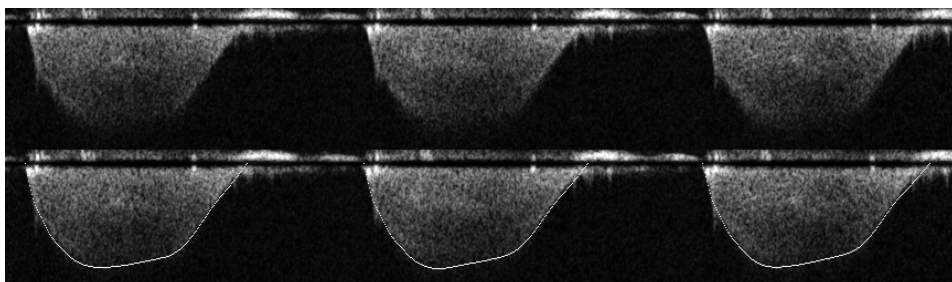
Programkoden er vedlagt masteroppgaven. Dette er en Matlabfil som inneholder metodene fra seksjon 3.6. Matlabfilen er laget som en plugin til GcLab.

Bilder

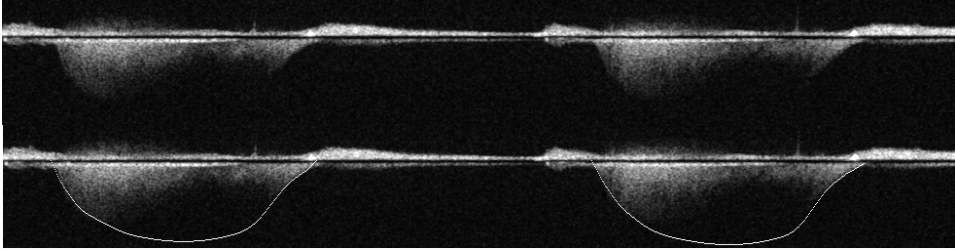
Her er det tatt med noen eksempler på Dopplerspektrum som er blitt sporet. Bildene viser Dopplerspektrumene før og etter at det er blitt sporet, og viser eksempler på spektrum hvor predikering er foretatt.



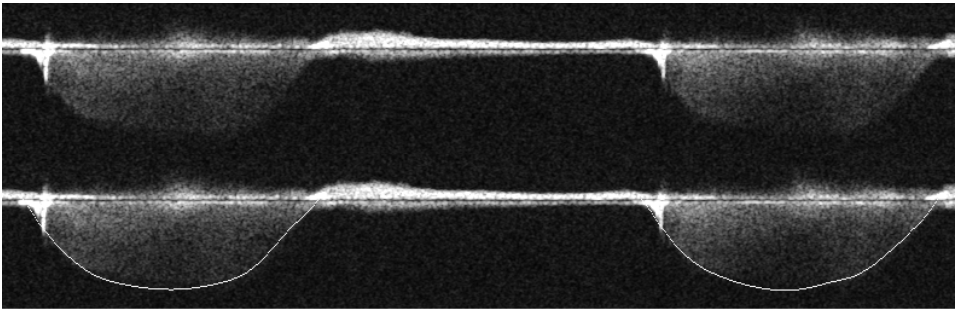
Uten predikering



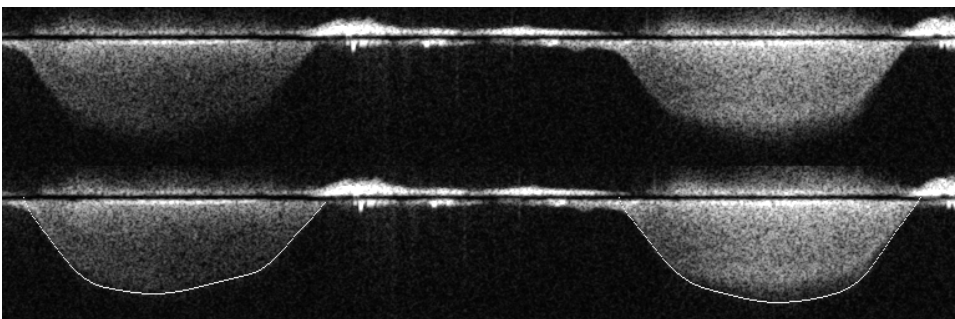
Uten predikering



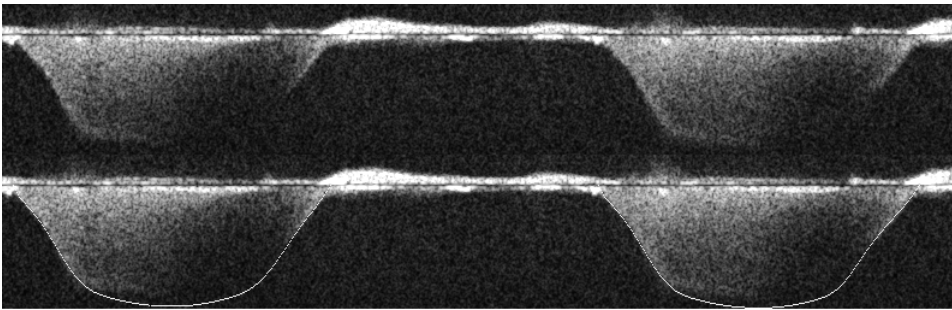
Med predikering



Uten predikering



Uten predikering



Med predikering

Bibliografi

- [1] Holen J, Aaslid R, Landmark K & Simonsen S., *Determination of pressure gradient in mitral stenosis with a non-invasive ultrasound Doppler technique*, 1976.
- [2] JinHyeong Park, S. Kevin Zhou, John Jackson & Dorin Comaniciu, *Automatic Mitral Valve Inflow Measurements from Doppler Echocardiography*, 2008.
- [3] Zhe Wang, Greg Slabaugh & Mengchu Zhou, *Automatic tracing of blood flow velocity in pulsed doppler images*, 4th IEEE conference on automation science and engineering, 2008.
- [4] Juerg Tschirren, Ronald M. Lauer & Milan Sonka, *Automated Analysis of Doppler Ultrasound Velocity Flow Diagrams*, 2001.
- [5] Hayit Greenspan, Oron Schechner, Micket Scheinowitz & Micha S. Feinberg, *Doppler echocardiography flow-velocity image analysis for patients with atrial fibrillation*, 2005.
- [6] Thomas Szabo, *Biomedical signal and image processing*, 2006.
- [7] Richard E. Klabunde PhD, *Cardiac cycle*, <http://www.cvphysiology.com/Heart%20Disease/HD002d.htm>, 13. mai, 2011.
- [8] Gregory M. Scalia, Neil L. Greenberg, Patrick M. McCarthy, James D. Thomas & Pieter M. Vandervoort, *Noninvasive assessment of the ventricular relaxation time constant(τ) in humans by doppler echocardiography*, *Circulation* Vol. 95, p.151-155, 1997.
- [9] Gunnar Smith, Stein Samstad & Helge Grendahl, *Ekko/Doppler: Kardiologisk ultralyddiagnostikk*, *Hjerteforum* suppl. 5, 2004.
- [10] DUKE university school of medicine, *Pulsed and Continuous Wave Doppler: Nyquist Limit*, http://www.echoincontext.com/doppler01/doppler01_12.asp, 1.juni, 2011.
- [11] CBSE Physics, *Bernoulli's principle*, <http://cbsephysics.blogspot.com/2010/01/bernoullis-principle.html>, 1.juni, 2011.

- [12] Sverre Holm, *Medisinsk ultralydabildning*, Fysikkens Verden, side 4-11, nr 1, 1999.
- [13] Gian Siro Bargiggia, Carla Bertucci, Franco Recusani, Arturo Raisaro, Stefano de Servi, Lilliam M. Valdes-Cruz, David J. Sahn & Luigi Tronconi, *A new method for estimating left ventricular dP/dt by continuous wave doppler-echocardiography*, Circulation Vo. 80, No 5, 1989.
- [14] Zhiming Ge, Yun Zhang, Weigiang Kang, Dongsheng Fan, Carlos Duran, *A Simultaneous Study of Doppler-Echo and Catheterization in Noninvasive Assessment of the Left Ventricular dp/dt* , Clin. Cardiol. 16,422-428, 19939.
- [15] Chunguang Chen, Leonardo Rodriguez, Jean-Paul Lethor, Robert A. Levine, Marc S. Semigran, Michael A. Fifer, Arthur E. Weyman & James D. Thomas, *Continuous wave doppler echocardiography for noninvasive assessment of left ventricular dP/dt and relaxation time constant from mitral regugigant spectra in patients*, JACC Vol. 23, No. 4, 1994.
- [16] Satoshi Nakatani, Mario J. Garcia, Michael S. Firstenberg, Leonardo Rodriguez, Richard A. Grimm, Neil L. Greenberg, Patrick M. McCarthy, Pieter M. Vandervoort, James D. Thomas, *Noninvasive Assessment of Left Atrial Maximum dP/dt by a Combination of Transmitral and Pulmonary Venous Flow*, Journal of the American College of Cardiology Vol. 34, No.3, 1999.
- [17] RSNA informatics, *The Value and Importance of an Imaging Standard*, <http://www.rsna.org/Technology/DICOM/index.cfm>, 25. mai, 2011.
- [18] Jozef Bartunek, Dirk De Bacquer, Ana Clara Rodrigues & Bernard De Bruyne, *Accuracy of aortic stenosis severity assessment by doppler echocardiography: importance of image quality*, 1995.
- [19] Sky McKinley & Megan Levine, *Cubic Spline interpolation*.
- [20] A C Hunt, C Denslow, K Carson, R Kilbey, S Murray & M Burrows, *Atrial Systolic Function in Left Heart Failure*. 2000.
- [21] R.M. Shoucri & J.G. Dumesnili, *Contribution to the theoretical study of the Doppler half-time method for the measurement of the cross-sectional area of the mitral valve* , 1989.
- [22] William C. Little, *The left ventricular dP/dt -max-end-diastolic volume relation in closed-chest dogs*, Circulation research, Vol. 56, No. 6, 1985.