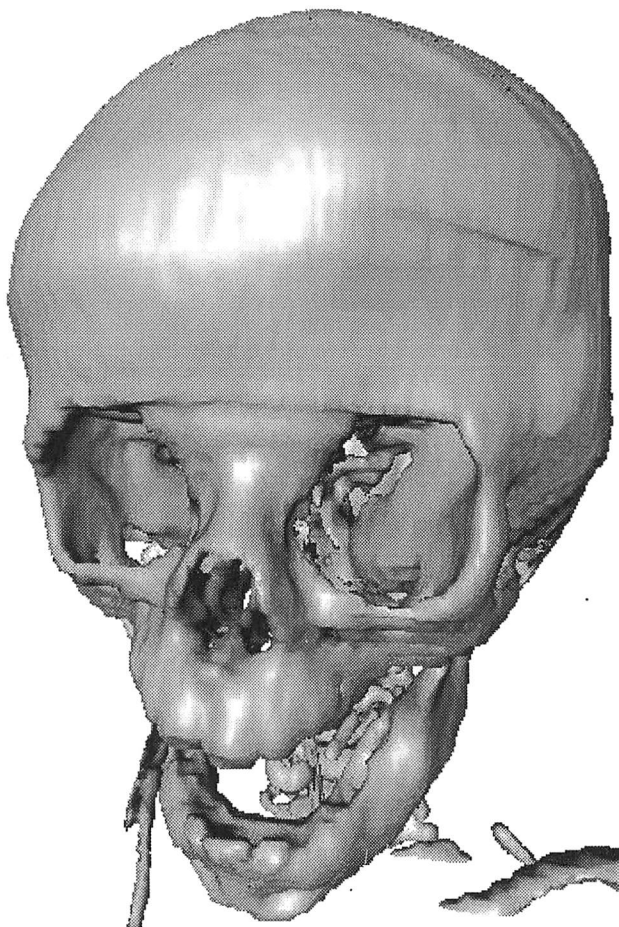


Martin Liversage

Visualisering af tredimensionale medicinske data



Speciale, Københavns Universitet
23. september 1994

Indhold

Forord	3
1 Indledning	5
1.1 Tredimensional visualisering	5
1.2 Overblik over specialets struktur	7
1.3 Terminologi og notation	8
 I Moderne medicinsk billeddannelse	 9
2 Tomografi — teoretisk baggrund	10
2.1 Historisk indledning	10
2.2 Motivation af Radon transformationen	11
2.3 Matematiske forudsætninger	12
2.4 Radon transformationen	13
2.5 Den inverse Radon transformation	14
2.6 Radon transformationen i højere dimensioner	16
2.7 Diskrete transformationer	18
2.8 Noter og litteratur	20
 3 Tomografi i praksis	 22
3.1 Medicinsk skanning	22
3.2 Røntgentomografi	25
3.3 Emissionstomografi	28
3.4 Kernemagnetisk resonans	31
3.5 Ultralydstomografi	34
3.6 Litteratur om medicinsk skanning	35
3.7 Tomografi inden for andre områder	36
 4 Visualisering af tredimensionale data	 39
4.1 Indledning	39
4.2 Visualisering af konturflader	40
4.3 Rumfangsgengivelse	42
4.4 Litteratur om rumfangsgengivelse	45
4.5 Konstruktion af konturflade	47
 II Konstruktion af kontur	 53
5 Billeder og flader	54
5.1 Digital topologi	54
5.2 Cellekomplekser	55
5.3 Digitale billeder	57
5.4 Foldning af billeder	59
5.5 Noter og litteratur	60

6 En todimensional tegl-algoritme	61
6.1 Indledning	61
6.2 Facetter	63
6.3 Flertydighed	65
6.4 Egenskaber ved tegl	67
6.5 Tegl og gruppeteori	69
6.6 Algoritmen	72
7 En tredimensional tegl-algoritme	78
7.1 Indledning	78
7.2 Facetter	79
7.3 Flertydighed	81
7.4 Tegl og gruppeteori	83
7.5 Algoritmen	86
7.6 Anvendelse af den todimensionale algoritme	89
7.7 Repræsentation af facetter	92
7.8 Implementering i praksis	94
7.9 En konkret implementering	95
8 Facetter og flader	100
8.1 Triangulering af facetter	100
8.2 Estimation af normaler	107
8.3 Forenkling af flade	112
9 Mere om tegl-algoritmer	117
9.1 Trappeeffekten	117
9.2 Anvendelse af n-topologien	122
9.3 Triangulering af tegl	125
9.4 Marching cubes	128
9.5 Noter og litteratur	129
10 Højere dimensioner	130
10.1 Generalisation til fire dimensioner	130
10.2 Anvendelse af den firedimensionale algoritme	131
10.3 Eksempler på højere dimensionale data	134
10.4 Generelle facetter	135
10.5 En generel tegl-algoritme	137
11 Afslutning	139
11.1 Resumé	139
11.2 Fordele ved tegl-algoritmer	140
11.3 Ulemper ved tegl-algoritmer	141
11.4 Konklusion	141
Bibliografi	142
Indeks	148

Forord

Jeg blev i 1988 ansat som studentermedhjælp på Tandlægeskolen (dengang Københavns Tandlægehøjskole) hos professor Sven Kreiborg på Børneafdelingen. Professor Kreiborg, som blandt andet interesserede sig for anvendelsen af computere til medicinsk forskning, anvendte nogle enkle programmer til „tredimensional“ visualisering af røntgentomogrammer. Disse programmer var udviklet hos professor Michael W. Vannier (Mallinckrodt Institute of Radiology, Washington University, St. Louis), og de kørte på en IBM AT computer. Inspireret af disse programmer begyndte jeg at udvikle mine egne programmer på professor Kreiborgs ny erhvervede Silicon Graphics 4D/20 grafiske arbejdsstation.

Samtidig med at jeg passede mit studie i fysik og matematik fortsatte mit engagement på Tandlægeskolen, blandt andet finansieret af *Schoildanns fond*, *Gangstedfonden* og *Ingeborg og Leo Dannins fond*. Jeg forbedrede de udviklede programmerne i samarbejde med professor Kreiborg der anvendte dem til praktiske medicinske formål. I efteråret 1990 tog jeg orlov fra universitetet for at arbejde fuld tid på Tandlægeskolen. I den periode var jeg så heldig at komme på et studieophold i professor Vanniers afdeling i St. Louis. Her fik jeg lejlighed til at se mange anvendelser af tredimensionale medicinske data, blandt andet ved hjælp af ANALYZE programmet fra Mayo klinikken i Rochester. Det var osse ved denne lejlighed jeg blev introduceret til „marching cubes“ algoritmen, hvilket mundede ud i en publikation om emnet [Liversage et al. 1992].

Det har været naturligt for mig at lade min faglige løbebane fortsætte ad den vej der blev udstukket på Tandlægeskolen, og jeg har derfor valgt at skrive speciale om visualisering af tredimensionale medicinske data.

Strengt taget har dette ikke meget med fysik at gøre, men erhvervelsen af disse data ved anvendelse af ioniserende stråling eller teknikker baseret på kernemagnetisk resonans har i høj grad noget med fysik at gøre. Dette speciale vil derfor osse beskrive hvorledes tredimensionale data erhverves, men hovedvægten vil stadig være lagt på hvordan de efterfølgende visualiseres.

I dette speciale optræder forskellige former for medicinsk billedmateriale. Dette materiale er venligst stillet til rådighed af følgende institutioner: *Tandlægeskolen, Københavns Universitet; Rigshospitalet; Mallinckrodt Institute of Radiology, Washington University* (St. Louis, Missouri); *Indiana University Department of Radiology* (Indianapolis, Indiana), samt *Austin Hospital* (Heidelberg, Victoria).

Flere illustrationer viser eksempler på abnorm anatomi, og det er klart at det er det medicinske studie af disse billeder der er det egentlige mål. Alligevel vil de medicinske emner af naturlige årsager ikke blive berørt i dette speciale som alene er af teknisk art. Fra illustrationerne kan man få det indtryk at det kun er kraniets anatomi som kan undersøges med teknikkerne beskrevet her, men dette er en tilfældighed som skyldes at man på Tandlægeskolen hovedsageligt har interesse i kraniets anatomi. Endelig må jeg bemærke at flere af illustrationerne desværre ikke helt kommer til deres ret da reproduktionen af dette speciale er foretaget på en almindelig laserprinter.

Undervejs i dette forløb er der mange mennesker uden hvis hjælp dette projekt ikke havde været muligt og som derfor fortjener en tak. Først og fremmest er der

professor Kreiborg og dem som i tidens løb har været tilknyttet Tandlægeskolens „3D-lab“. Blandt andre Fred Heller som har bistået med korrekturlæsning. Endvidere er der naturligvis mine to vejledere lektor Stig Steenstrup og lektor Jens Damgaard Andersen. Derudover har Peter Harremoës givet mig uvurderlig bistand med matematisk vanskelige problemer. Sidst men ikke mindst er der min kone og datter som har accepteret at betale de omkostninger specialeskrivning kan have for familielivet.

*Martin Liversage
September 1994*

Kapitel 1

Indledning

- 1.1 Tredimensional visualisering
- 1.2 Overblik over specialets struktur
- 1.3 Terminologi og notation

1.1 Tredimensional visualisering

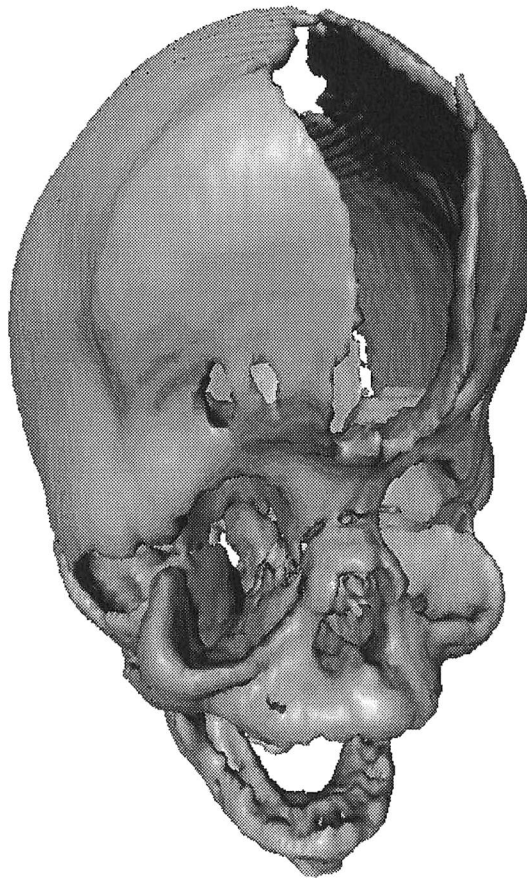
Som det fremgår af titlen af dette speciale handler det om visualisering af tredimensionale data. Den slags data optræder i mange situationer, men jeg har først og fremmest valgt at se på anvendelser inden for medicinen.

Men hvad menes egentlig med tredimensionale data? Jeg vil nøjes med at betragte tredimensionale *skalar*data. Mere præcist kræves det at dataværdierne er repræsenteret på et tredimensionalt, rektangulært *gitter*. Denne type data kan opfattes som en generalisation af et todimensionalt, digitalt billede til tre dimensioner, og hovedemnet for dette speciale er hvorledes et tredimensionalt, digitalt billede visualiseres. Og med „visualisering“ menes hvorledes man bedst præsenterer de pågældende data på en computerskærm så en interesseret bruger kan få adgang til den ønskede information.

Når et tredimensionalt billede skal visualiseres må man vælge hvad man vil studere i billedet. Dette kan for eksempel foregå ved at man forsøger at bestemme en *ækvipotentiaflade* i billedet for en bestemt *ækvipotentiafværdi*. Løst sagt vil en sådan flade gå igennem alle de punkter i det tredimensionale billede der har en billedværdi lig med ækvipotentiafværdien. Har man først en ækvipotentiaflade er det velkendt fra *computer graphics* hvorledes den visualiseres.

Det er ofte sådan at det netop er en ækvipotentiaflade som giver brugeren den ønskede information om indholdet af det tredimensionale billede. Dette er tilfældet inden for mange forskellige discipliner — specielt inden for medicinen. Med udgangspunkt i *medicinsk skanning* kan man bestemme overfladen af anatomiske strukturer, som for eksempel organer eller knogler, ved at konstruere en ækvipotentiaflade for en passende ækvipotentiafværdi. Et eksempel på dette er vist på figur 1.1, og det er klart at billeder af den slags er til stor hjælp i forbindelse med behandlingen af alvorlige lidelser. I dette speciale vil der blive beskrevet en teknik som gør det muligt at fremstille sådanne billeder af levende mennesker. Har lægen rådighed over en meget kraftig computer vil det tilmed være muligt at manipulere modellen interaktivt på skærmen. I skrivende stund er minimumsprisen for sådan en computer adskillige hundrede tusinde kroner, og derfor er interaktiv manipulation af tredimensionale medicinske visualiseringer endnu kun for de færreste.

Nu er disse metoder ikke blot interessante set ud fra et medicinsk synspunkt. Tre-



Figur 1.1: Et eksempel på tredimensional visualisering af kraniet hos et barn.

dimensionale billeder kan fås fra mange kilder, og disse kan opdeles i tre hovedgrupper:

1. *Empiriske data*, det vil sige data som kommer fra en måling af et naturligt forekommende fænomen. Et godt eksempel er medicinske data som fås ved anvendelse af *tomografi* således at en række *tomogrammer* eller tværsnitsbilleder af anatomen konstrueres. Disse tomogrammer kan kombineres til et tredimensionalt billede. Et andet eksempel på tredimensionale empiriske data er seismiske data.
2. *Simulerede data*, det vil sige data som kommer fra en simulering af et system for hvilket man har en matematisk model. Eksempler er meteorologiske simuleringer af stormvejr eller beregninger af elektronorbitaler i atomer og molekyler.
3. *Syntetiske data*. Her er formålet at konstruere en geometrisk model af et objekt der er vanskeligt at opbygge ved sædvanlige metoder, det vil for eksempel sige ved anvendelse af *CSG (Constructive Solid Geometry)*. I stedet kan man konstruere et analytisk udtryk for et felt der har den ønskede flade som ækvipotentiaflade og registrere dette på et gitter.

1.2 Overblik over specialets struktur

Dette speciale er delt op i to dele. Den første del omhandler anvendelse af tomografi, først og fremmest inden for medicinen, men andre anvendelser beskrives osse kort. Ved hjælp af tomografi erhverves sædvanlige (det vil sige todimensionale) billeder som senere kan sammensættes til tredimensionale billeder. Beskrivelsen af tomografi baserer sig udelukkende på andre værker, og der vil derfor ikke være noget væsentligt nyt at finde i disse afsnit.

Første del af specialet starter med *kapitel 2* som er en teoretisk gennemgang af tomografi. Her vil *Radon transformation* blive beskrevet.

I *kapitel 3* vil hovedprincipperne for *medicinsk skanning* baseret på tomografi blive omtalt. Anvendelse af tomografi inden for andre områder vil osse kort blive berørt.

I *kapitel 4* vil metoder til visualisering af tredimensionale data erhvervet ved for eksempel medicinsk skanning blive gennemgået.

I den anden del beskrives i detaljer en teknik hvormed tredimensionale data kan visualiseres. Mere præcist så opstilles en algoritme til konstruktion af tilnærmede ækvi-potentialflader som i dette speciale kaldes for en *tegl-algoritme*. Det er disse afsnit som er kernen i specialet og som indeholder emner der ikke er beskrevet før.

I *kapitel 5* vises hvordan *cellekomplekser* kendt fra matematikken kan anvendes som en abstraktion af såvel digitale billeder som flader. Dette giver et fundament at bygge videre på.

I *kapitel 6* opstilles en todimensional tegl-algoritme — ikke så meget fordi en sådan algoritme er nyttig, men mere fordi det bliver muligt at introducere en lang række begreber og metoder på en enkel måde.

Mange af disse ideer vil blive anvendt i *kapitel 7* hvor en tredimensional tegl-algoritme bliver beskrevet. Her vil en konkret implementering af algoritmen osse blive omtalt og nogle resultater vil blive vist.

I *kapitel 8* bliver nogle emner knyttet til de flader der konstrueres af tegl-algoritmen uddybet. Her beskrives *triangulering* af fladen, estimation af *normaler* for fladen og metoder til *forenkling* af fladen berøres.

I *kapitel 9* samles der op på nogle af de emner vedrørende tegl-algoritmer som ikke hører til de foregående kapitler. Nogle af de problemer der kan være med *aliasing* i forbindelse med tegl-algoritmer vil blive omtalt, og nogle ideer, der måske kan løse disse problemer, vil blive skitseret. Endvidere beskrives det hvorledes man kan gå fra en *kubisk* til en *simplicial* geometri i de underliggende data. Endelig vil nogle af problemerne med tegl-algoritmen kendt under navnet „marching cubes“ blive diskuteret.

I *kapitel 10* beskrives det hvordan tegl-algoritmer kan generaliseres til vilkårlige dimensioner. Fokus vil først og fremmest være på fire dimensioner, men det n -dimensionale problem vil osse blive berørt.

Endelig vil jeg i *kapitel 11* forsøge at runde specialet af, dels ved at se tilbage på hvad jeg har opnået, og dels ved at diskutere anvendeligheden af tegl-algoritmer til tredimensional medicinsk visualisering.

1.3 Terminologi og notation

Da de emner der beskrives i anden del endnu er forholdsvis nye findes der endnu ingen fælles vedtaget terminologi, og de få fælles begreber der trods alt findes har ikke nogen fælles vedtaget dansk oversættelse. På den anden side er det vigtigt at begreberne „har et navn“ for så er man i stand til at håndtere abstrakte emner sprogligt. Det er ikke tilfældigt at en matematisk tekst ofte indeholder en lang række definitioner, for på den måde bliver det abstrakte gjort håndgribeligt.

Derfor vil der undervejs i dette speciale blive introduceret en omfattende terminologi. Jeg har prøvet at gøre mit bedste for gøre denne terminologi forståelig og konsistent, men der vil måske alligevel være læsere som finder mine valg dårlige eller upassende sprogligt set. I det tilfælde håber jeg at de vil bære over med mig og læse videre uden for megen irritation.

I forbindelse med angivelse af størrelser, for eksempel hvor lang tid det tager at udføre et bestemt program, vil jeg ofte anvende *størrelsesordner*. Her er størrelsen „afrundet“ til nærmeste potens af ti. En proces der tager *størrelsesordnen* 100 sekunder varer derfor noget mere end ti sekunder, men mindre en 1.000 sekunder. Mere præcist så interesserer jeg mig kun for titals-logaritmen til tiden afrundet til nærmeste hele tal.

Forfatternavne og årstal som i teksten er skrevet i firkantede parenteser („[...]“) henviser til bibliografien bagerst i specialet.

Den matematiske notation som benyttes i dette speciale vil svare til den standard der benyttes i undervisningen på Københavns Universitet. Der er dog nogle få specielle udtryk som det er passende at definere her.

For $i, m, n \in \mathbb{Z}$ med $m < n$ sættes

$$i|_m^n = (i - m) \bmod (n - m) + m.$$

Dette udtryk kan forstås således at i „normaliseres“ til et helt tal $j = i|_m^n$ som opfylder $m \leq j < n$. Hvis $m = 0$ er $i|_0^n = i \bmod n$. Ofte er $m = 1$ og så kan $i|_1^n$ benyttes til for eksempel at beregne indeks i „cykliske“ tabeller der er indekseret fra 1 til og med n .

For at fjerne enhver tvivl kan jeg nævne at notationen $n|m$ er prædikatet „ n går op i m “ og $f|_X$ er restriktionen af afbildningen f til mængden X .

Algoritmer vil blive beskrevet i en form for pseudokode som er ligetil at forstå hvis man tidligere har beskæftiget sig med programmering af computere. Et særligt kendetegn for den anvendte pseudokode er at blokstruktur vises ved anvendelse af indrykning. To „datastrukturer“ vil flittigt blive anvendt: Det drejer sig om *mængder* og *sekvenser*. For eksempel kan mængden X tilføjes elementet x ved tilskrivningen

$$X := X \cup \{x\}$$

og sekvensen Y kan tilføjes elementet y i slutningen ved tilskrivningen

$$Y := (Y, y).$$

I

Moderne medicinsk billedannelse

Kapitel 2

Tomografi — teoretisk baggrund

- 2.1 Historisk indledning
- 2.2 Motivation af Radon transformationen
- 2.3 Matematiske forudsætninger
- 2.4 Radon transformationen
- 2.5 Den inverse Radon transformation
- 2.6 Radon transformationen i højere dimensioner
- 2.7 Diskrete transformationer
- 2.8 Noter og litteratur

I dette kapitel vil det teoretiske fundament for medicinsk tomografi blive beskrevet. Efter en historisk indledning og et lille eksempel som tjener til at motivere matematikken, vil grundpillen i det teoretiske fundament for tomografi, den såkaldte *Radon transformation*, blive beskrevet i to dimensioner. Osse højere dimensioner vil kort blive omtalt. Endelig vil nogle af de matematiske problemer der opstår når Radon transformationen skal anvendes i praksis blive omtalt.

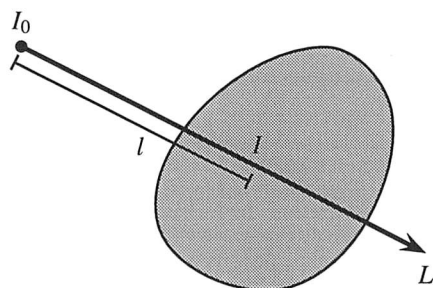
En del af materialet har tidligere været anvendt til et *kollokvium* afholdt i forbindelse med min hovedfagseksamen i fysik. Fremstillingen vil i niveau derfor typisk henvende sig til en (yngre) fysikstuderende. Er man interesseret i en mere fyldig gennemgang er der i slutningen af kapitlet en lang række henvisninger til litteratur om emnet.

2.1 Historisk indledning

Opdagelsen af Røntgenstrålingen i 1895 havde stor betydning ikke bare i fysikken, men osse i medicinen — det blev pludseligt muligt at foretage ikke-invasive undersøgelser af menneskets indre. Selv om en røntgenundersøgelse ikke er helt ufarlig er den dog mere behagelig end en tilsvarende undersøgelse foretaget med kanyler, skalpeller og knive.

Igennem tiden er metoden blevet forbedret på mange punkter, men introduktion af *røntgentomografi* osse kaldet *CT-skanning* er måske den største revolution inden for røntgenteknologien og den medicinsk billedbehandling i vort århundrede, og i 1979 fik Allan Mecleod Cormack og Godfrey Newbold Hounsfield nobelprisen i medicin for deres indsats inden for området [Cormack 1980, Hounsfield 1980].

Cormack opstillede i halvtredserne det teoretiske fundament for en metode hvor man ved at tage en række røntgenbilleder fra forskellige vinkler af et legeme kunne rekonstruere et tværsnitsbillede, osse kaldet et *tomogram*, der beskrev den indre struktur af legemet [Cormack 1963, Cormack 1964]. Senere i tresserne udviklede Hounsfield uafhængigt af Cormack på baggrund af grundige teoretiske overvejelser flere prototyper



Figur 2.1: En røntgenstråle sendes gennem et medie.

på CT-skannere [Hounsfield 1973]. I 1971 kunne man installere den såkaldte *EMI brain scanner* på *Atkinson Morleys Hospital* i England, og allerede få år efter var der kommercielle skannere tilgængelige.

Det matematiske princip bag tomografi var allerede kendt af Johann Radon [Radon 1983, Deans 1983] (østrigsk matematiker, 1887–1956) og er siden blevet „genopdaget“ adskillige gange [Cormack 1983].

2.2 Motivation af Radon transformationen

Inden de mere formelle matematiske udredninger vedrørende Radon transformationen påbegyndes er det på sin plads med en motivation af denne teori, og den hentes fra røntgentomografien.

2.2.1 Røntgenabsorption. Betragt en tynd, monokromatisk røntgenstråle L der sendes igennem et medie (se figur 2.1). Hvis I betegner intensiteten, l distancen tilbagelagt og $\mu(l)$ den brøkdel af intensiteten der absorberes eller spredes så fås følgende differentiaalligning

$$dI = -\mu(l)I dl$$

der kan løses ved

$$\int_{I_0}^{I_l} \frac{1}{I} dI = - \int_0^l \mu(l') dl'$$

hvor I_0 er den oprindelige intensitet af strålen og I_l er den endelige intensitet af strålen. Da intensiteten I altid er positiv kan (2.1) løses og man får

$$\log \frac{I}{I_0} = - \int_L \mu(l) dl. \quad (2.1)$$

2.2.2 Svækkelseskoefficienten er det velklingende navn man har givet μ . Den er afhængig af tætheden og sammensætningen af det materiale strålen passerer igennem. Man kan se at man direkte kan få linieintegralet over svækkelseskoefficienten ud fra en måling af intensiteten.

2.2.3 Røntgentomografi er en teknik hvor man måler en række linieintegraler af stråler indeholdt i et plan, og $\mu = \mu(x, y)$ bliver da en funktion af de cartesiske koordinater x og y i planet. Formålet er at rekonstruere $\mu(x, y)$ ud fra kendskabet til linieintegralerne og her kan man benytte Radon transformationen (eller mere præcist den inverse Radon transformation).

Hvis planet går gennem et levende menneske giver kendskabet til $\mu(x, y)$ en masse oplysninger om hvordan denne person „ser ud“ i det plan (se figur 3.2.B på side 24). Svækkelseskoefficienten for de forskellige vævstyper i kroppen er forskellige, og man bliver i stand til at studere anatomen af den pågældende person.

2.3 Matematiske forudsætninger

2.3.1 Transformation. Ved en *transformation* forstås en lineær afbildning fra et normeret vektorrum til et andet (eventuelt det samme) vektorrum. Vektorrummet vil typisk være funktionsrum, for eksempel af kvadratisk integrable funktioner på en passende mængde. En transformation kaldes osse for en *operator* eller *funktional*.

2.3.2 Bemærkning. En fuldstændig fremstilling af stoffet ville kræve at man omhyggeligt angav hvilket funktionsrum man betragtede. Endvidere ville det være nødvendigt i detaljer at studere på hvilken delmængde af dette rum en given transformation er veldefineret og invertibel.

Alt dette vil jeg undlade at bekymre mig om. Jeg vil i stedet antage at der i konkrete tilfælde ikke vil opstå problemer med invertibilitet, konvergens, integrabilitet og så videre. Denne antagelse er acceptabel da jeg først og fremmest interesserer mig for praktiske anvendelser. De matematiske detaljer overlades til matematikerne [Hertle 1981].

Specielt kan det nævnes at man må regne med at de udsagn om lighed der bringes her skal opfattes som lighed *næsten overalt* da det er integral-transformationer der betragtes.

Til senere brug opstilles nu Fourier transformationen og dens inverse.

2.3.3 Definition. Givet en kompleks funktion $f: \mathbb{R} \rightarrow \mathbb{C}$ på den reelle akse defineres dens Fourier transformerende $\mathcal{F}_1 f$ som

$$F(\xi) = (\mathcal{F}_1 f)(\xi) = \int_{-\infty}^{\infty} f(x) e^{-2\pi i \xi x} dx.$$

Dette er på ny en kompleks funktion $F: \mathbb{R} \rightarrow \mathbb{C}$ på den reelle akse.

2.3.4 Definition. Fourier transformation har en invers givet ved

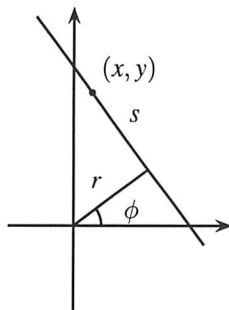
$$f(x) = (\mathcal{F}_1^{-1} F)(x) = \int_{-\infty}^{\infty} F(\xi) e^{2\pi i \xi x} d\xi.$$

For $F = \mathcal{F}_1 f$ gælder der at $f = \mathcal{F}_1^{-1} F$ med de forbehold om „lighed“ der blev taget ovenfor.

Fourier transformation kan generaliseres til n dimensioner. Her angives den todimensionale udgave. Udvidelse til højere dimensioner følger nemt.

2.3.5 Definition. For $f: \mathbb{R}^2 \rightarrow \mathbb{C}$ defineres dens Fourier transformerende som

$$F(\xi, \eta) = (\mathcal{F}_2 f)(\xi, \eta) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-2\pi i (\xi x + \eta y)} dx dy.$$



Figur 2.2: Linie i planet.

2.3.6 Definition. Den inverse Fourier transformation af $F: \mathbb{R}^2 \rightarrow \mathbb{C}$ er

$$f(x, y) = (\mathcal{F}_2^{-1}F)(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(\xi, \eta) e^{2\pi i(\xi x + \eta y)} d\xi d\eta. \quad (2.2)$$

Der gælder som i det éndimensionale tilfælde at $f = \mathcal{F}_2^{-1}F$ når $F = \mathcal{F}_2 f$.

2.4 Radon transformationen

Husk at man med en røntgenstråle er i stand til at måle et linieintegral over svækkelses-koefficient μ . Denne proces kan formelt beskrives ved hjælp af *Radon transformationen*. Lad der være givet en reel funktion $\mu: \mathbb{R}^2 \rightarrow \mathbb{C}$ i planet. Her anvendes navnet μ så man bliver mindet om at funktionen for eksempel kunne være svækkelseskoefficienten af vævet i et snitplan gennem et levende menneske.

2.4.1 Linie i planet kan specificeres ved to parametre r og ϕ hvor linien løber igennem punktet $(r \cos \phi, r \sin \phi)$ og har normalvektoren $(-\sin \phi, \cos \phi)$. Linien kan parametriseres ved s

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} r \cos \phi - s \sin \phi \\ r \sin \phi + s \cos \phi \end{pmatrix}, \quad s \in \mathbb{R}$$

som vist på figur 2.2.

Alle linier i planet kan fås hvis man kræver $r \in \mathbb{R}$ og $0 \leq \phi < \pi$. Dette fører til...

2.4.2 Definition. (Det todimensionale Radon rum.)

$$\mathbb{Y}_2 = \mathbb{R} \times [0; \pi[.$$

Den *todimensionale Radon transformation* af μ er linieintegralet af μ langs linien givet ved r og ϕ

2.4.3 Definition. (Radon transformationen.)

$$\lambda(r, \phi) = (\mathcal{R}_2 \mu)(r, \phi) = \int_{-\infty}^{\infty} \mu(r \cos \phi - s \sin \phi, r \sin \phi + s \cos \phi) ds.$$

Der gælder $\lambda: \mathbb{Y}_2 \rightarrow \mathbb{R}$.

Fremover vil λ være betegnelsen for den Radon transformerende af μ .

2.4.4 Bemærkning. Hvis man ser nærmere på λ 's definitions­mængde $\mathbb{Y}_2$ så ses det at den kan ændres til $[0; \infty[\times [0; 2\pi[$. Dette minder om polære koordinater, *men er det ikke* da der for $\phi_1 \neq \phi_2$ i almindelighed gælder at $\lambda(0, \phi_1) \neq \lambda(0, \phi_2)$.

2.4.5 Definition. (Tilbageprojektion.) For $\lambda: \mathbb{Y}_2 \rightarrow \mathbb{R}$ sættes

$$\tilde{\lambda}(x, y) = (\mathcal{B}_2 \lambda)(x, y) = \int_0^\pi \lambda(x \cos \phi + y \sin \phi, \phi) d\phi, \quad (x, y) \in \mathbb{R}^2. \quad (2.3)$$

Der gælder at $\tilde{\cdot}: \mathbb{R}^2 \rightarrow \mathbb{R}$, og man kan vise at $2\mathcal{B}_2 = \mathcal{R}_2^*$ hvor $\mathcal{R}_2^*$ er den adjungerede transformation af $\mathcal{R}_2$.

For $\lambda = \mathcal{R}_2 \mu$ er $\tilde{\lambda} = \mathcal{B}_2 \lambda$ i et bestemt punkt (x_0, y_0) gennemsnittet af alle linieintegraler gennem dette punkt (for at være helt korrekt skal der divideres med π for at få gennemsnittet). Man kan anvende tilbageprojektion til at foretage en grov rekonstruktion, men det er bedre at invertere Radon transformationen.

2.5 Den inverse Radon transformation

Den Radon transformerende λ af μ er den størrelse man er i stand til at måle og målet er derfor at invertere Radon transformationen så μ kan bestemmes.

2.5.1 Fourier transformationen. Det viser sig at der er en overraskende sammenhæng mellem Fourier transformationen og Radon transformationen. For et givet $\phi \in [0; \pi[$ gælder

$$\begin{aligned} (\mathcal{F}_1 \lambda)(\rho, \phi) &= \int_{-\infty}^{\infty} \lambda(r, \phi) e^{-2\pi i \rho r} dr \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \mu(r \cos \phi - s \sin \phi, r \sin \phi + s \cos \phi) e^{-2\pi i \rho r} ds dr. \end{aligned} \quad (2.4)$$

2.5.2 Bemærkning. Her benyttes den éndimensionale Fourier transformation på en funktion defineret på $\mathbb{Y}_2$. Det er underforstået at den anden variabel, ϕ er fast under transformationen. Dette giver en hel skare af transformerende funktioner — én for hver værdi af ϕ , og resultatet er endnu engang en funktion defineret på $\mathbb{Y}_2$.

2.5.3 Koordinattransformation. For at forenkle (2.4) roteres koordinatsystemet ved at vælge nye x - og y -koordinater givet ved

$$x = r \cos \phi - s \sin \phi, \quad y = r \sin \phi + s \cos \phi$$

som kan inverteres til

$$r = x \cos \phi + y \sin \phi, \quad s = -x \sin \phi + y \cos \phi.$$

Jacobianten af rotationen er identiteten og (2.4) får i det nye koordinatsystem formen

$$(\mathcal{F}_1 \lambda)(\rho, \phi) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \mu(x, y) e^{-2\pi i (x\rho \cos \phi + y\rho \sin \phi)} dx dy. \quad (2.5)$$

Hvis man sætter

$$\xi = \rho \cos \phi, \quad \eta = \rho \sin \phi \quad (2.6)$$

kan (2.5) omskrives til

$$(\mathcal{F}_1\lambda)(\xi, \eta) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \mu(x, y) e^{-2\pi i(x\xi + y\eta)} dx dy,$$

men det er netop $(\mathcal{F}_2\mu)(\xi, \eta)$. Derfor kan man formulere

2.5.4 Sætning. (Projektionssætningen.)

$$\mathcal{F}_2\mu = \mathcal{F}_1\lambda. \quad (2.7)$$

Man kan bestemme μ ved at anvende $\mathcal{F}_2^{-1}$ på $\mathcal{F}_1\lambda$. Da $\lambda = \mathcal{R}_2\mu$ fås følgende udtryk for den inverse Radon transformation:

2.5.5 Sætning.

$$\mathcal{R}_2^{-1} = \mathcal{F}_2^{-1}\mathcal{F}_1. \quad (2.8)$$

2.5.6 EksPLICIT INVERSIONSFORMEL. Udtrykket (2.8) giver den inverse Radon transformation, men der mangler stadig et eksPLICIT udtryk som kan anvendes i praktiske beregninger. Den komplikation man straks støder på er at $\mathcal{F}_1\lambda$ er en funktion med definitionsmængde $\mathbb{Y}_2$, mens den udgave af $\mathcal{F}_2^{-1}$ der er angivet i (2.2) transformerer en funktion defineret på $\mathbb{R}^2$. Derfor opskrives...

2.5.7 Identitet

$$\begin{aligned} \mu(x, y) &= (\mathcal{F}_2^{-1}\mathcal{F}_2\mu)(x, y) \\ &= \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} (\mathcal{F}_2\mu)(\xi, \eta) e^{2\pi i(x\xi + y\eta)} d\xi d\eta. \end{aligned} \quad (2.9)$$

Det sidste integral opskrives i polære koordinater givet ved (2.6). Man skal huske at Jacobianten under denne koordinattransformation er ρ og (2.9) kan derfor omskrives til

$$\mu(x, y) = \int_0^{2\pi} \int_0^{\infty} (\mathcal{F}_2\mu)(\rho \cos \phi, \rho \sin \phi) e^{2\pi i(x\rho \cos \phi + y\rho \sin \phi)} \rho d\rho d\phi. \quad (2.10)$$

I stedet for at integrere hele cirklen rundt over halvlinier fra 0 til ∞ kan man integrere over en halvcirkel og over linier fra $-\infty$ til ∞ , blot man husker at erstatte ρ med $|\rho|$ i integranden. Dermed opnås at integralet bliver over $\mathbb{Y}_2$.

2.5.8 Inversionsformel. Ved at indsætte (2.7) i (2.10) fås

$$\mu(x, y) = \int_0^{\pi} \int_{-\infty}^{\infty} (\mathcal{F}_1\lambda)(\rho, \phi) e^{2\pi i(x\rho \cos \phi + y\rho \sin \phi)} |\rho| d\rho d\phi. \quad (2.11)$$

For bedre at forstå denne inversionsformel foretages endnu et par omskrivninger.

2.5.9 Definition. Lad funktionen $A: \mathbb{Y}_2 \rightarrow [0; \infty[$ være givet ved

$$A(\rho, \phi) = |\rho|. \quad (2.12)$$

Integralet over ρ i (2.11) identificeres som $\mathcal{F}_1^{-1}$, men det ses nemmest ved at indsætte A . Man får

$$\begin{aligned}\mu(x, y) &= \int_0^\pi \int_{-\infty}^\infty (A \mathcal{F}_1 \lambda)(\rho, \phi) e^{2\pi i \rho(x \cos \phi + y \sin \phi)} d\rho d\phi \\ &= \int_0^\pi \mathcal{F}_1^{-1}(A \mathcal{F}_1 \lambda)(x \cos \phi + y \sin \phi, \phi) d\phi.\end{aligned}\quad (2.13)$$

Tilbageprojektion defineret i (2.3) genkendes og (2.13) forenkles til

2.5.10 Sætning. (Filtreret tilbageprojektion.)

$$\mu(x, y) = (\mathcal{B}_2 \mathcal{F}_1^{-1}(A \mathcal{F}_1 \lambda))(x, y). \quad (2.14)$$

Denne inversionsformel er den man oftest benytter i konkrete tilfælde da den er velegnet til implementering på computer. Det lidt besynderlige navn *filtreret tilbageprojektion* skal forstås på følgende baggrund:

2.5.11 Signalteori. For fast ϕ kan man tænke på den Radon transformerende $\lambda(r, \phi)$ som et tidsvarierende signal hvor r svarer til tiden. Ved at Fourier transformere dette signal fås $(\mathcal{F}_1 \lambda)(\rho, \phi)$ som er et frekvensspektrum af signalet. Frekvensvariablen kaldes i dette tilfælde for ρ . Når man multiplicerer dette spektrum med $A = |\rho|$ filtreres spektret og funktionen A kaldes for *filter respons-funktionen*. Filtringen er det man kalder en *højpas filtrering* hvor de numerisk høje frekvenser forstærkes mens de numerisk lave frekvenser dæmpes. Man får et nyt spektrum $(A \mathcal{F}_1 \lambda)(\rho, \phi)$ som transformeres tilbage til et tidsvarierende signal $(\mathcal{F}_1^{-1}(A \mathcal{F}_1 \lambda))(r, \phi)$. Dette signal tilbageprojiceres med $\mathcal{B}_2$.

Med andre ord så filtreres den Radon transformerende før den tilbageprojiceres. Man kunne i stedet have tilbageprojiceret direkte uden filtrering, men den resulterede rekonstruktion ville være meget dårlig uden detaljer, netop som hvis man havde foretaget en *lavpas filtrering* af den oprindelige funktion μ [Holm 1988].

2.6 Radon transformationen i højere dimensioner

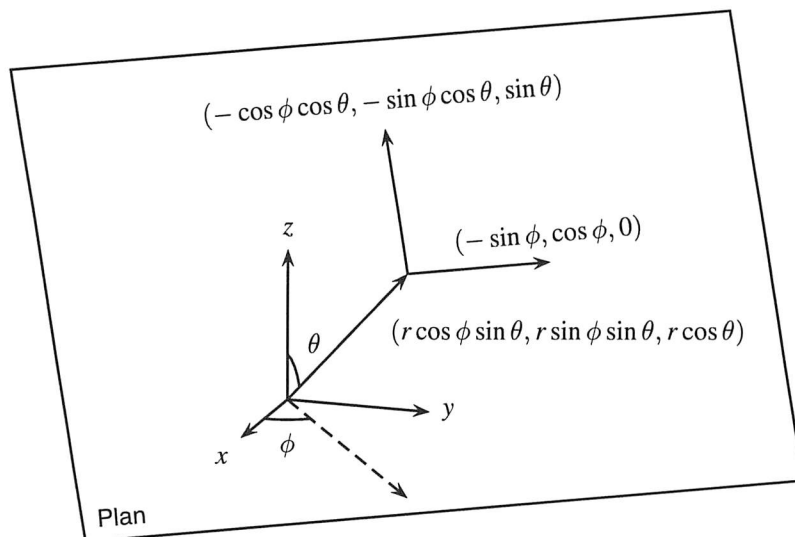
Det kan osse lade sig gøre at definere den tredimensionale Radon transformation og dens inverse. Udledninger minder meget om det tilsvarende todimensionale tilfælde så jeg vil være noget mere kortfattet her. Se i øvrigt [Barrett 1984].

2.6.1 Plan i rummet. Den tredimensionale Radon transformation er et planintegral. Som illustreret på figur 2.3 kan planet specificeres ved tre parametre r , ϕ og θ og parametriseres ved s og t

$$\begin{aligned}(x, y, z) &= (\cos \phi \sin \theta, \sin \phi \sin \theta, \cos \theta)r \\ &\quad + (-\sin \phi, \cos \phi, 0)s \\ &\quad + (-\cos \phi \cos \theta, -\sin \phi \cos \theta, \sin \theta)t \quad (s, t) \in \mathbb{R}^2.\end{aligned}\quad (2.15)$$

2.6.2 Definition. (Tredimensionalt Radon rum.)

$$\mathbb{Y}_3 = \mathbb{R} \times [0; \pi[\times [0; \pi[. \quad (2.16)$$



Figur 2.3: Plan i rummet.

Alle planer i rummet fås hvis man kræver

$$(r, \phi, \theta) \in \mathbb{Y}_3.$$

2.6.3 Definition. (Tredimensional Radon transformation.) Givet $\mu: \mathbb{R}^3 \rightarrow \mathbb{C}$ defineres Radon transformationen som

$$\lambda(r, \phi, \theta) = (\mathcal{R}_3 \mu)(r, \phi, \theta) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \mu(x(s, t), y(s, t), z(s, t)) ds dt$$

hvor (2.15) definerer $(x(s, t), y(s, t), z(s, t))$. Der gælder $\lambda: \mathbb{Y}_3 \rightarrow \mathbb{C}$.

2.6.4 Definition. (Tredimensional tilbageprojektion.) Tilbageprojektion defineres som

$$(\mathcal{B}_3 \lambda)(x, y, z) = \int_0^\pi \int_0^\pi \lambda(x \cos \phi \sin \theta + y \sin \phi \sin \theta + z \cos \theta, \phi, \theta) d\phi d\theta.$$

Man har følgende tredimensionale varianter af de todimensionale sætninger.

2.6.5 Sætning. (Projektionssætning.)

$$\mathcal{R}_3^{-1} = \mathcal{F}_3^{-1} \mathcal{F}_1. \quad (2.17)$$

2.6.6 Sætning. (Filtreret tilbageprojektion.)

$$\mathcal{R}_3^{-1} = \mathcal{B}_3 \mathcal{F}_1^{-1} (A^2 \mathcal{F}_1)$$

hvor $A: \mathbb{Y}_3 \rightarrow \mathbb{C}$ fås ved at generalisere definitionen (2.12) til $A(\rho, \phi, \theta) = |\rho|$.

2.6.7 Filter respons-funktionen. Her skal man især bemærke at filter respons-funktionen i det tredimensionale tilfælde er A^2 som er differentiabel på hele $\mathbb{R}$. I det todimensionale tilfælde er filter respons-funktionen A , og den er ikke differentiabel i 0.

2.6.8 Højere dimensioner. Det hele kan osse generaliseres til n dimensioner ved passende steder at erstatte tal der angiver dimensioner med et n . For eksempel bliver udtrykket for filtreret tilbageprojektion

$$\mathcal{R}_n^{-1} = \mathcal{B}_n \mathcal{F}_1^{-1} (A^{n-1} \mathcal{F}_1).$$

For $n > 3$ har problemet dog hovedsageligt interesse for matematikere.

2.7 Diskrete transformationer

2.7.1 Inversion i praksis. Indtil nu er det blevet vist hvordan en ukendt funktion kan bestemmes når dens Radon transformation er kendt. Dette fungerer fint i teorien, men når det kommer til praksis dukker der straks problemer op. Man skal huske på at den Radon transformerende af en todimensional funktion selv er en todimensional funktion. Med andre ord så skal man kende *alle* linieintegralerne $\lambda(r, \phi)$ for entydigt at bestemme $\mu(x, y)$.

I det praktiske tilfælde kendes kun λ for et endeligt antal værdier af r og ϕ og man kan derfor ikke entydigt bestemme μ . Hvis Y betegner mængden af par $(r, \phi) \in \mathbb{Y}_2$ hvor λ er målt så findes mange forskellige μ som opfylder $\lambda = (\mathcal{R}_2 \mu)|_Y$.

Det er nødvendigt at opstille en algoritme der er i stand til at finde en god approksimation til μ givet en diskret måling af λ . Man må forvente at kriteriet om „godhed“ er forskelligt alt efter hvilken konkret anvendelse man betragter. For eksempel vil det være overraskende hvis en radioastronom, der ser på stjerner, og en læge, der ser på anatomi, definerer „godhed“ på samme måde.

Man kan opdele de almindelige rekonstruktionsalgoritmer i to klasser: *operator metoder* og *iterative metoder* [Herman 1985].

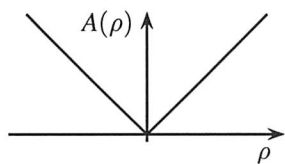
2.7.2 Operator metoden er et forsøg på at implementere den filtrerede tilbageprojektion i en diskret udgave. På samme måde som man i Fourier analyse benytter diskrete transformationer har transformationerne som optræder i (2.14) diskrete analoge. De svarer til de kontinuerte med integrationerne erstattet af summer over et endeligt antal punkter. Man kan opstille sammenhængende matematisk teori for disse transformationer, og det er blandt andet gjort af Rowland [1979] som der henvises til for detaljer.

2.7.3 Filter respons-funktionen er et særligt kritisk punkt. Funktionen A , som optræder i (2.14), forstærker de høje frekvenser, men støj i målingen er i reglen højfrekvent. I praksis må man derfor dæmpe filtrets forstærkning for høje frekvenser for at få et rimeligt signal-støj forhold. Dette gøres ved at foretage en *bånd begrænsning* hvorved alle frekvenser der er numerisk over en vis grænse ρ_0 skæres bort. Endvidere kan man mindske forstærkningen for frekvenser nær den øvre grænse så filtret for ρ nær 0 stadig går som $|\rho|$, mens det for $|\rho|$ nær ρ_0 går imod 0 (se figur 2.4).

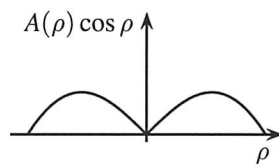
2.7.4 Den iterative metode er et forsøg på at udvikle μ på et sæt af basisfunktioner. Udviklingskoefficienterne bestemmes ved en iterativ algoritme, deraf navnet.

2.7.5 Udvikling. Sættet af basisfunktioner betegnes med

$$e_i, \quad i = 1, \dots, n,$$



A) Teoretisk funktion.



B) Praktisk funktion hvor filtret er „rullet af“ for at dæmpe forstærkningen af høje frekvenser.

Figur 2.4: Filter respons-funktioner.

og lad $\hat{\mu}$ være et estimat for μ udviklet på sættet af basisfunktioner

$$\hat{\mu} = \sum_{i=1}^n a_i e_i \quad (2.18)$$

hvor a_i , $i = 1, \dots, n$ er koefficienterne i udviklingen.

Hvis man ønsker slutresultatet repræsenteret som et billede med for eksempel 256×256 billedpunkter så kan man som basisfunktioner e_i vælge indikatorfunktionen på hver af de enkelte billedpunkter. Udviklingskoefficienten a_i er så billedværdien i punkt i .

Ved at Radon transformere (2.18) fås et estimat for $\lambda = \mathcal{R}_2 \mu$

$$\hat{\lambda} = \mathcal{R}_2 \hat{\mu} = \mathcal{R}_2 \sum_{i=1}^n a_i e_i = \sum_{i=1}^n a_i \mathcal{R}_2 e_i = \sum_{i=1}^n a_i f_i \quad (2.19)$$

hvor $f_i = \mathcal{R}_2 e_i$ indsættes.

2.7.6 Diskret udvikling. I realiteten kan de Radon transformerede basisfunktioner f_i kun repræsenteres på et endeligt antal punkter i $\mathbb{Y}_2$. Lad antallet af punkterne være m , og lad punkterne være givet ved

$$(r_j, \phi_j) \in \mathbb{Y}_2 \quad j = 1, \dots, m.$$

Man kan opskrive (2.19) i punktet indiceret ved j og får

$$\hat{\lambda}_j = \hat{\lambda}(r_j, \phi_j) = \sum_{i=1}^n a_i f_{ij}$$

hvor $f_{ij} = f_i(r_j, \phi_j)$.

Introduceres vektorerne

$$\hat{\lambda} = (\hat{\lambda}_1, \dots, \hat{\lambda}_m), \quad \mathbf{a} = (a_1, \dots, a_n)$$

og matricen

$$\mathbf{F} = \begin{pmatrix} f_{11} & \dots & f_{1n} \\ \vdots & \ddots & \vdots \\ f_{m1} & \dots & f_{mn} \end{pmatrix}$$

får (2.19) følgende diskrete form

$$\hat{\lambda} = \mathbf{aF}.$$

2.7.7 Tilpasning til måling. Antag at en konkret måling af λ foretages på punkterne (r_j, ϕ_j) med $j = 1, \dots, m$. Den målte værdi betegnes med $\tilde{\lambda}_j$. Målingen kan skrives mere hensigtsmæssigt i form af en *målevektor*

$$\tilde{\lambda} = (\tilde{\lambda}_1, \dots, \tilde{\lambda}_m).$$

Fejlvektoren ϵ defineres som forskellen mellem målingen og estimatet

$$\epsilon = \tilde{\lambda} - \hat{\lambda}.$$

Ved at indsætte (2.20) fås et udtryk for målevektoren

$$\tilde{\lambda} = \mathbf{a}F + \epsilon.$$

For en given målevektor $\tilde{\lambda}$ ønskes det at bestemme udviklingskoefficienterne $\mathbf{a}$ så fejlen ϵ minimeres. Når en optimal $\mathbf{a}$ er bestemt indsættes denne i (2.18) så estimatet $\hat{\mu}$ for μ fås.

2.7.8 Bemærkning. Undervejs er der blevet introduceret flere tilnærmelser. For det første approksimeres μ med et estimat $\hat{\mu}$ som er en udvikling på et *endeligt* antal basisfunktioner. For det andet „diskretiseres“ Radon rummet $\mathbb{Y}_2$ ved valg af målepunkter så den Radon transformerende af estimatet skrives som vektoren $\hat{\lambda}$. Endelig tilpasses $\hat{\lambda}$ til en konkret måling $\tilde{\lambda}$.

Hvis ikke valget af basisfunktioner er fornuftigt, hvis ikke valget af målepunkter er fornuftigt, og hvis ikke tilpasningen af $\hat{\lambda}$ til $\tilde{\lambda}$ er fornuftig så kan man nemt risikere at løsningen bliver meget dårlig.

2.7.9 Iterativ optimering. Ligningen (2.20) kan løses ved en serie af iterationer hvor man „skyder“ sig ind på en optimal $\mathbf{a}$ [Pierro 1990, Censor 1990]. Selv om det lyder enkelt er for eksempel F meget stor i praktiske tilfælde. Endvidere viser det sig ofte at en lille ændring i målevektoren kan give en stor ændring i løsningen. En vigtig klasse af velegnede algoritmer kaldes for *ART* som står for *Algebraic Reconstruction Technique*.

2.7.10 Konklusion. Medicinsk skanning eller tomografi er nok den mest udbredte anvendelse af de her beskrevne principper. Her benytter man næsten udelukkende operator metoder, for eksempel filtreret tilbageprojektion.

På den anden side så benyttede den første CT-skanner en ART algoritme, og da iterative metoder kan anvendes inden for ganske mange områder er det stadig et aktive felt.

2.8 Noter og litteratur

2.8.1 Radon transformationen. Hvis man ønsker at studere Radon transformationen og den matematiske teori der er knyttet til denne har Deans [1983] forfattet et værk om netop disse emner der absolut kan anbefales. Endvidere bør Radons originale artikel nævnes — den er genoptrykt i [Radon 1983] og findes oversat til engelsk i [Deans 1983] for dem der foretrækker at læse engelsk frem for tysk. Cormack fik som nævnt nobelprisen, og hans oprindelige artikler findes i [Cormack 1963, Cormack 1964]. Desuden giver Rowland [1979] en systematisk, men alligevel enkel gennemgang af teorien. Barrett [1984] præsenterer det samme matematiske indhold i en noget anderledes form. Hvis man er interesseret i dybere matematiske resultater vedrørende

Radon transformationen har, ud over Deans [1983], for eksempel Zeitler [1974], Smith et al. [1977], Natterer [1980], Hertle [1981], Helgason [1983], Boman [1990] og Maass [1990] beskæftiget sig med dette.

2.8.2 Rekonstruktionsalgoritmer er gennemgået på en teoretisk velfunderet måde af Rowland [1979]. Herman [1985] bringer en mere overfladisk introduktion til emnet. Endvidere er kendte og/eller nye rekonstruktionsalgoritmer beskrevet af Smith et al. [1977], Smith [1983], Steenstrup [1990], Censor [1990] og Pierro [1990].

2.8.3 Historien om Radons oprindelige værk fra 1917, de senere „genopdagelser“ af hans resultater, og som kronen på værket, nobelprisen i 1979, er interessant nok noget som man ofte støder på i litteraturen om emnet. Det skyldes måske ikke mindst at Cormack, efter han modtog nobelprisen, har ønsket at gøre det klart at han ikke var den eneste pioner inden for området [Cormack 1980, Cormack 1983] — han var bare den første der fandt på at bruge teknikken inden for medicinen, og det skulle føre til en „revolution“ som mange har beskrevet det. Af andre pionerer nævnes især Bracewell og Terman der har udviklet og anvendt teknikker baseret på Radon transformation inden for radioastronomien. Dette er for eksempel beskrevet af Bracewell [1979]. Osse inden for medicinen var der andre end Cormack og Hounsfield der udviklede lignende teknikker, og det er kort refereret af Deans [1983]. Endelig kommer Barrett [1984] ind på nogle tidlige anvendelser af Radons resultater.

2.8.4 Generel matematisk baggrund kan man få fra utallige matematiske lærebøger. Jeg kan henvise til [Rees et al. 1981, Cartwright 1990] der beskriver Fourier analyse.

Kapitel 3

Tomografi i praksis

- 3.1 Medicinsk skanning
- 3.2 Røntgentomografi
- 3.3 Emissionstomografi
- 3.4 Kernemagnetisk resonans
- 3.5 Ultralydstomografi
- 3.6 Litteratur om medicinsk skanning
- 3.7 Tomografi inden for andre områder

I dette kapitel beskrives hvordan tomografi kan anvendes i praksis. Hovedvægten vil ligge på en beskrivelse af *medicinsk skanning* og fire forskellige former vil blive beskrevet. Beskrivelserne er ret overfladiske og skal udelukkende tjene til at give læseren en introduktion til emnet. I afsnit 3.6 findes henvisninger til mere specialiseret litteratur. Endelig er der til slut et afsnit som kort opregner en række områder uden for medicinen hvor tomografi anvendes.

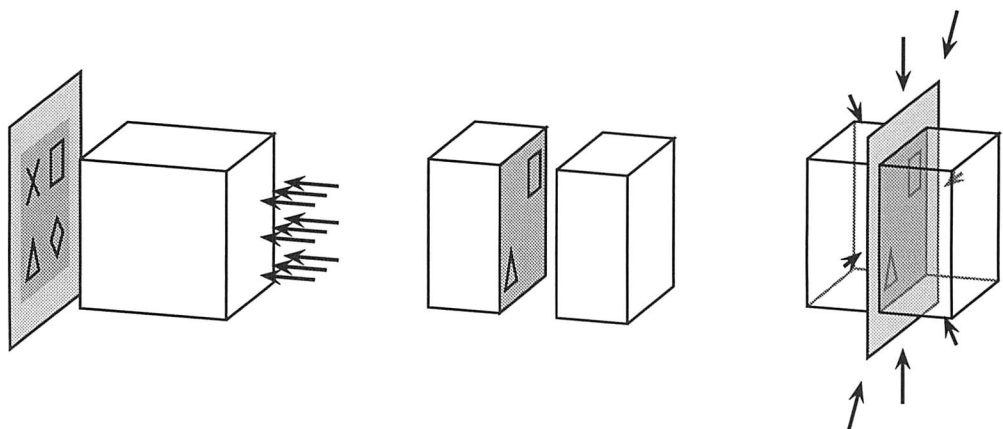
3.1 Medicinsk skanning

Her benyttes *medicinsk skanning* som en fællesbetegnelse for metoder hvor man med avanceret teknologi erhverver tværsnitsbilleder af patientens anatomi og/eller funktionalitet af organer. Et tværsnitsbillede kaldes inden for medicinen for et *tomogram* og konstruktionen af tomogrammer kaldes for *tomografi*. Tomogrammer er værdifulde kilder til information — så værdifuld at (nogle af) opfinderne er blevet belønnet med nobelprisen.

3.1.1 Illustration af tomografi. Figur 3.1 illustrerer forskellen mellem et traditionelt medicinsk billede og et tomogram. Ved et „traditionelt medicinsk billede“ tænkes her først og fremmest på et røntgenbillede, men osse andre teknikker hører til denne kategori, for eksempel *scintigrafi*.

3.1.2 Gennemlysning. Det traditionelle billede fås ved en gennemlysning af emnet som vist på figur 3.1.A hvor pilene viser hvordan strålingen trænger ind. Meget nyttig information kan fås fra sådan et billede, men der er osse mange detaljer man ikke ser. Dette skyldes at billedet er en sum eller projektion af emnet der studeres, og vigtige strukturer kan derfor være placeret „ovenpå“ hinanden i billedet. Dette ses på filmplanet til venstre som en række geometriske figurer.

3.1.3 Dissektion. Man kan i stedet skære emnet over langs et snitplan som vist på figur 3.1.B. Derved kan man direkte betragte den indre struktur i snitplanet. Man gen-



A) Gennemlysning.

B) Fysisk snit.

C) Tomogram.

Figur 3.1: Principskitse af forskellen mellem traditionelt medicinsk billede og tomogram.

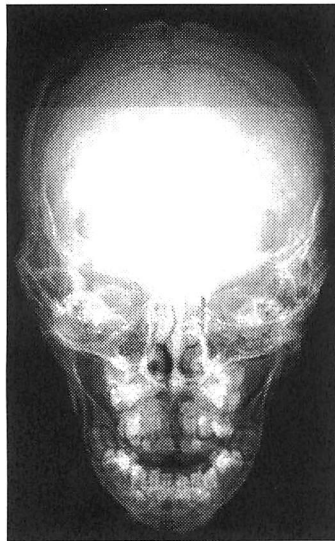
kender nogle af de geometriske former der optrådte på gennemlysningsbilledet. Uheldigvis er metoden destruktiv og derfor ikke egnet til levende mennesker.

3.1.4 Røntgentomografi. Endelig kan man konstruere et tomogram af det ønskede snitplan som vist på figur 3.1.C. Stråling (for eksempel røntgenstråling i røntgentomografi) sendes igennem emnet i snitplanet som vist med pilene. Ved anvendelse af rekonstruktionsteknikkerne beskrevet tidligere kan man konstruere et tomogram, og for nemmere at kunne se dette er emnet gjort transparent på figuren. I tomogrammet genkendes de samme struktur som sås ved dissektion langs snitplanet. Forskellen er at metoden er *ikke-invasiv* og *ikke-destruktiv*.

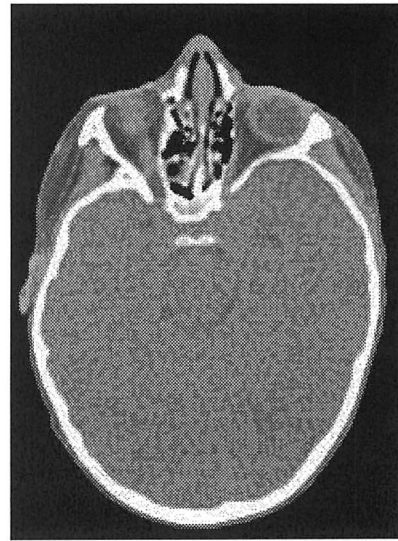
For yderligere at illustrere forskellen mellem traditionelle røntgenbilleder og røntgentomogrammer kan man på figur 3.2.A se et røntgenbillede. Det er en gennemlysning af et kranie forfra. På figur 3.2.B ser man et horisontalt røntgentomogram fra øjenregionen. På røntgenbilledet har man hele kraniet repræsenteret på filmen, mens man på tomogrammet kun har et enkelt snitplan. Til gengæld er der langt flere detaljer i tomogrammet.

3.1.5 En typisk skanner ses på figur 3.3. Patienten er placeret på bordet som kan skydes ind igennem *gantryet*. I gantryet er der monteret udstyr til måling af en eller anden form for linieintegraler. Gantryet definerer et plan vinkelret på patientens længdeakse som kaldes for *snitplanet*, og tomogrammet er en måling af den interessante størrelse i dette plan. Man har oftest mulighed for at vippe gantryet og/eller patientbordet for at få et snitplan der ikke nødvendigvis er vinkelret på patientens længdeakse.

3.1.6 Skannebilledet. Hele skanneren styres af en tilsluttet computer som osse tager sig af konstruktionen af tomogrammet. Når dette er klart kan det betragtes på en skærm eller gemmes til senere brug eller udskrift. Tomogrammet er i realiteten et digitalt billede bestående af billedpunkter (*pixels*). Hvert billedpunkt har en værdi som afbildes på en computerskærm med en bestemt grå- eller farvetone. Hvis man benytter gråtoner svarer høje værdier i reglen til lyse toner. Anvendes farvetoner ses det ofte at høje værdier svarer til „varme“ toner, mens lave værdier svarer til „kolde“ toner, men der er naturligvis mange forskellige måder at gøre dette på.

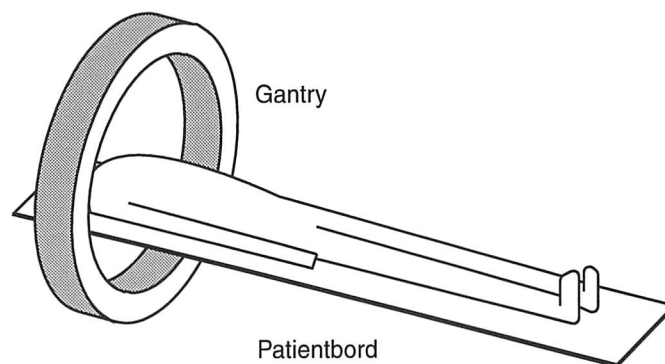


A) Et traditionelt røntgenbillede af kraniet set forfra (frontal projektion).



B) Et horisontalt røntgentomogram fra øjenregionen (aksialt).

Figur 3.2: To typer røntgenbilleder.



Figur 3.3: Skematisk tegning af en medicinsk skanner.

3.1.7 Opløsningen af billedet afhænger meget af skanneteknikken og typiske værdier er fra 32×32 til 512×512 billedpunkter. Den virkelige opløsning som måleteknikken tilbyder behøver ikke at svare til opløsningen på billedet. Man kan sænke opløsningen af billedet hvorved man får et mere „udtværet“ billede med mindre støj. I stedet kan man hæve opløsningen for at give det indtryk at opløsningen er højere end den egentlig er. Endelig er der mulighed for at filtrere og manipulere billedet inden det præsenteres for på den måde at gøre billedet „pænere“. Her skal man blot passe på at man ikke taber information eller tilføjer falsk information i denne proces.

Det er som regel umuligt at få oplyst hvilke processer et skannebillede fra en kommerciel skanner har været igennem, da der er en hård konkurrence mellem de forskellige producenter. Derfor er man i praksis nødt til at stole på at billedet er „korrekt“.

3.1.8 Skanneformer. I det efterfølgende vil de mest almindelige skanneformer kort blive beskrevet. Det drejer sig om:

1. Røntgentomografi eller CT-skanning.
2. Emissionstomografi som findes i to former: SPECT og PET.
3. Kernemagnetisk resonans osse kaldet MRI.
4. Ultralydstomografi.

De første to metoder anvender ioniserende stråling og er derfor ikke helt ufarlige for patienten. De to sidste former regnes derimod i dag for ufarlige. Om dette osse er helt korrekt vil jeg ikke komme ind på her, men de er helt sikkert meget *mindre* skadelige.

CT og MRI giver billeder af en høj kvalitet med et stort anatomisk indhold: CT fremhæver især knoglestrukturen, mens MRI er i stand til skelne mellem mange forskellige typer *blødt væv* osse kaldet *bløddele*. Med MRI kan man osse til en vis grad studere de biokemiske processer der foregår i vævet.

Med SPECT og PET opnår man ikke en så høj billedmæssig kvalitet, men man er i stand til at studere de biokemiske processer der foregår i det levende menneske.

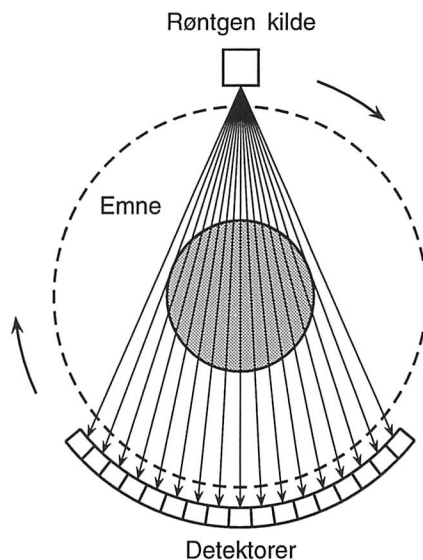
Ultralydstomografi giver mulighed for at studere anatomien, men har, lissom SPECT og PET, en ringere billedkvalitet end CT og MRI. Enkelheden af udstyret og de minimale risici involveret gør dog at denne teknik er attraktiv.

3.2 Røntgentomografi

Den første form for tomografi, som blev introduceret i 1971, og den til dato mest anvendte form er baseret på røntgenstråling. Da konstruktionen af tomogrammer kræver et meget stort antal beregninger som i realiteten kun kan udføres på en computer, så benytter man betegnelsen *Computed Tomography* eller i daglig tale *CT-skanning* om røntgentomografi. Det er lidt forvirrende da alle andre former for tomografi med lige så stor ret kan kaldes for CT. En mere dækkende betegnelse ville for eksempel være *XCT* for *X-ray Computed Tomography*. En anden vej ud af dilemmaet er at lade betegnelsen CT hænge ved røntgentomografien. Så kan man kalde den generelle teknik for *RT* for *Reconstructive Tomography* hvilket dog ikke har slået an [Deans 1983]. Og når jeg nu alligevel er ved det kan jeg nævne den ældre betegnelse *CAT* som angiveligt skulle stå for *Computer Assisted Tomography* eller *Computerized Axial Tomography*.

3.2.1 Anvendelse. Røntgentomografi er meget anvendelig til detaljerede anatomiske studier da opløsningsevnen er god og støjen lav. Da knogler absorberer røntgenstråling meget kraftigere end andre former for væv er især disse tydelige på røntgentomogrammer. Det kan være en del sværere at skelne mellem forskellige typer blødt væv da svækkelseskoefficienten ikke varierer særligt meget for disse typer væv. Ved at indgive patienten forskellige former for kontraststof kan man dog råde bod på dette.

Det er klart at røntgentomografi vil vinde mere og mere frem som et dagligdags redskab for lægen. Et af problemerne er dog de forholdsvis høje stråledoser patienten bliver udsat for.



Figur 3.4: Skematisk tegning af gantryet på en CT-skanner.

3.2.2 Forenklet virkemåde. På figur 3.4 ses en principskitse af gantryet på en typisk CT-skanner. I midten ses emnet der skannes. Dette emne gennemlyses af røntgenstråler udsendt fra kilden foroven med en typisk energi på 90–120 keV [Budinger et al. 1979]. Mængden af stråling der trænger igennem emnet registreres af en række detektorer nederst, og da begyndelsesintensiteten er kendt måler man i realiteten linieintegraler over svækkelseskoefficienten af emnet jævnfør udtrykket (2.1). Her skal man bemærke at strålebundtet er kollimeret så det er begrænset til planet som figuren repræsenterer.

Hele arrangementet kan drejes så tilstrækkeligt mange linieintegraler kan måles. Typisk måler man for hver halve grad hvilket med gængse skannere giver en samlet procestid for ét tomogram på cirka et halvt minut. Patientbordet kan derefter flyttes til en ny position og et nyt tomogram kan konstrueres. Skannebilledet har som regel en opløsning på 256×256 eller 512×512 billedpunkter og størrelsen af et billedpunkt er af størrelsesordenen $1 \text{ mm} \times 1 \text{ mm}$. Afstanden mellem to efterfølgende tomogrammer kan være så lille som 1 mm.

3.2.3 Hounsfield enheder. Billedværdierne i et tomogram måles ofte i *Hounsfield Units* (HU) navngivet således til ære for opfinderen af CT-skanneren. Denne skala er en lineær skalering af svækkelseskoefficient μ , og den er kalibreret således at vand har en værdi på 0 HU og luft -1.000 HU. Hvis svækkelseskoefficienten af vand betegnes med μ_{H_2O} så fås følgende udtryk for den skalerede svækkelseskoefficienten

$$\mu_{HU} = 1.000 \cdot \frac{\mu - \mu_{H_2O}}{\mu_{H_2O}}$$

idet lufts svækkelseskoefficient er nul sammenlignet med vands.

I tabel 3.5 kan man se et repræsentativt udsnit af værdier for forskellige typer væv.

3.2.4 Skannetid. I visse tilfælde ønsker man at tage af størrelsesordenen hundrede tomogrammer af patienten, for eksempel når man skal indsamle data til konstruktion af

Væv	Interval (HU)
Luft	~ -1.000
Lunger	$-100 - -750$
Fedt	$-100 - -70$
Vand	~ 0
Nyre	$-30 - 20$
Hjerte	$-20 - 10$
Muskler	$-10 - 80$
Hjerne	$-10 - 20$
Blod	$0 - 90$
Lever	$40 - 100$
Blød knogle	$100 - 250$
Hård knogle	$200 - 4.000$

Tabel 3.5: Intervaller for forskellige vævstyper målt i Hounsfield enheder.

en model af patientens kranie (hvilket netop er et af hovedtemaerne i dette speciale). Med en skannetid på cirka et halvt minut tager dette omkring én time — og i al den tid må patienten *ikke* røre sig, hvilket naturligvis er et problem.

3.2.5 Nye skannere. Siden 1971 hvor dem såkaldte *EMI brain scanner* blev taget i brug har CT-skanneren udviklet sig igennem flere såkaldte generationer [Robb 1985d]. Den nyeste generation af skannere er i stand til inden for meget kort tid at indsamle data nok til at konstruere adskillige tomogrammer. På Rigshospitalet har man for nylig fået installeret en såkaldt *spiral skanner* eller *helisk skanner* der på omkring fem minutter kan konstruere halvtreds tomogrammer uden at patienten flyttes gennem gantryet [Siemens 1991, Kalender 1993]. Med den såkaldte *DSR-skanner* er man med multiple kilder og kameraer i stand til at „filme“ hjertets slag på flere tomogrammer samtidig [Wood et al. 1979, Robb 1985c].

3.2.6 Hærdning. Beskrivelsen af CT-skannerens virkemåde har været meget forenklet. Blandt andet har jeg ikke omtalt hvorledes geometrien af kilden og detektoren modificerer beregningen af tilbageprojektion. Endvidere var en af forudsætningerne i udledningen af udtrykket (2.1) for røntgenabsorptionen at strålen var monokromatisk. Svækkelseskoefficienten μ varierer med frekvensen og forskellige frekvenser giver derfor forskellige målinger. I praksis er strålingen *ikke* monokromatisk, og man må tage højde for at spektret ændres efterhånden som strålen passerer gennem stoffet på grund af μ 's afhængighed af frekvensen. Denne effekt kaldes for *hærdning* af strålen.

Et bestemt område af emnet som skannes rammes fra forskellige retninger af stråler der er hærdet forskelligt, og bidraget til de forskellige linieintegraler bliver dermed ikke nødvendigvis det samme. Herman [1985] beskriver hvordan man med en iterativ proces kan korrigere for denne effekt. Det antydes at denne teknik *ikke* anvendes i kommercielle skannere, men om dette osse er korrekt i dag ved jeg ikke.

3.2.7 Snittykkelse. Som nævnt er strålebundtet kollimeret så det ligger i et plan. I praksis er planet ikke uendelig tynd og man kan regulere tykkelsen af planet ved at

ændre kollimationen. Dette kalder man for *snittykkelsen* og hvis den er relativt stor får man et bedre signal-støj forhold. På den anden side bliver tomogrammet en midling af svækkelseskoefficient over hele snittykkelsen og præcise billeder kræver derfor en lille snittykkelse. Man arbejder typisk med snittykkelser på 1–10 mm.

3.2.8 Del-rumfangs effekten (på engelsk *partial volume effect*) er en betegnelse der dækker over at de enkelte billedpunkter på tomogrammet ikke repræsenterer et matematisk punkt i snitplanet, men i stedet svarer til en lille terning i rummet. Størrelsen af terningen afhænger af opløsningen i snitplanet samt snittykkelsen. Værdien af billedpunktet er gennemsnittet af svækkelseskoefficienten over hele terningen.

3.2.9 Voxel er den betegnelse man hæfter på terningen. Det skal ses i analogi til betegnelsen *pixel* der står for *picture element*, det vil sige billedpunkt, i et todimensionalt billede. *Voxel* står for *volume element* for at fremhæve den tredimensionale udstrækning.

3.3 Emissionstomografi

Emissionstomografi er en betegnelse der dækker over flere former for tomografi hvor patienten indgives en radioaktiv isotop der forårsager udsendelse af γ -stråling. Denne stråling måles og tomogrammer konstrueres. Lissom med røntgentomografi er der her et problem i forbindelse med stråledosis.

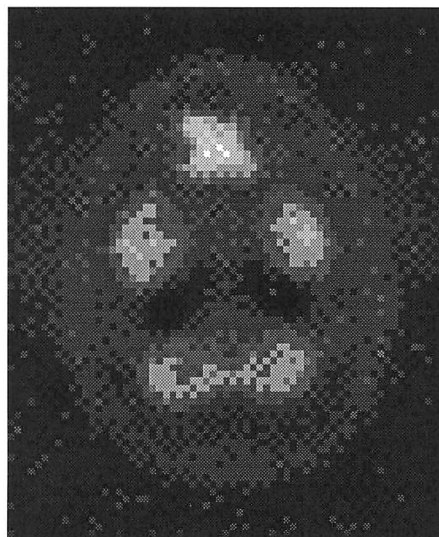
3.3.1 Emission og transmission. I modsætning til røntgentomografi hvor strålerne sendes fra en kilde uden for patienten gennem patientens krop, benytter man i emissionstomografi kilder der er inden i patientens krop. Denne form for tomografi kaldes derfor for *ECT* der står for *Emission Computed Tomography*. Så kan man kalde røntgentomografi for *TCT* der står for *Transmission Computed Tomography*.

3.3.2 Passiv og aktiv tomografi. En anden måde at skelne mellem ECT og TCT er følgende: ECT er en passiv form for tomografi der måler stråling udsendt fra et emne. TCT derimod er en aktiv form for tomografi hvor man undersøger emnet ved at sende stråling igennem det.

3.3.3 Den eksponentielle Radon transformation. I Røntgentomografi kender man kildens placering og dens styrke, og man studerer hvordan strålen svækkes af det mellemliggende stof. I emissionstomografi er hverken kildens placering eller dens styrke kendt, og man kender heller ikke svækkelseskoefficienten af det mellemliggende stof. Man kan derfor ikke benytte de rekonstruktionsmetoder beskrevet ovenfor direkte, men ved hjælp af den *eksponentielle Radon transformation* kan problemet løses [Budinger et al. 1979, Tretiak 1983, Barrett 1984].

3.3.4 Datamængden til rådighed ved emissionstomografi er typisk 1.000–10.000 gange mindre end ved røntgentomografi, og man må osse regne med et forholdsvis dårligt signal-støj forhold i disse data sammenlignet med røntgentomografi [Budinger et al. 1979]. Den rumlige opløsning af emissionstomografi bliver derfor af størrelsesordenen cm i stedet for mm som er muligt med røntgentomografi.

3.3.5 Anvendelse af andre skanneformer. Man kan forbedre kvaliteten af rekonstruktionen ved at anvende anatomiske informationer erhvervet med for eksempel CT eller MRI som er langt mere detaljerede. Man er ved at udvikle systemer der kombinerer de



Figur 3.6: PET-tomogram som viser glucose-metabolismen horisontalt gennem hjernen (aksialt). Isotopen anvendt er ^{18}F .

medicinsk set nyttige informationer man kan få fra emissionstomografi med de teknisk set detaljerede informationer man kan få fra CT og MRI [Gindi et al. 1991, Leahy og Yan 1991].

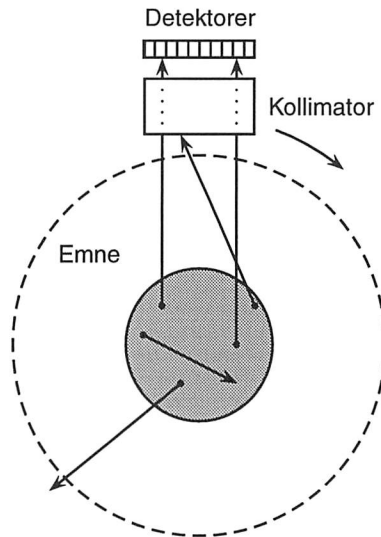
3.3.6 Statisk billeddannelse. Patienten indgives et sporstof som er „mærket“ med en radioaktiv isotop. Man kan foretage en *statisk billeddannelse* hvor man lissom i røntgentomografi studerer anatomen. Ved at vælge et passende sporstof er man i stand til at studere den del af anatomen som er relevant i det pågældende tilfælde. Det kan for eksempel være et spørgsmål om at lokalisere en svulst, og til dette benyttes et sporstof som koncentrerer i svulsten.

Statisk billeddannelsen kan osse give information om aktivitet i en bestemt vævstype. Dette kan lade sig gøre hvis aktiviteten er proportional til koncentrationen af et bestemt stof i vævet som man er i stand til at spore. Dette giver for eksempel mulighed for at studere glucose-metabolismen i hjernen.

3.3.7 Dynamisk billeddannelse. Hvis sporstoffet borttransporteres fra det område der studeres kan man få indblik i denne proces ved at måle hvordan aktiviteten ændres. Dette kan for eksempel give indsigt i blodgennemstrømning og teknikken kaldes for *dynamisk billeddannelse*.

3.3.8 Emissionstomografi former. Man kan inddele emissionstomografi i to kategorier: *Single Photon Emission Computed Tomography* og *Positron Emission Tomography*, ofte forkortet *SPECT* og *PET*. På figur 3.6 ses et eksempel på et PET-tomogram lodret gennem hovedets symmetriplan. Tonerne i billedet afslører glucose-metabolismen i hjernen — lyse toner svarer til høj metabolisme.

3.3.9 SPECT benytter isotoper der henfalder under udsendelse af en γ -foton med en energi som typisk ligger i området 80–500 keV. Isotoperne kan for eksempel være ^{123}I , ^{201}Tl , ^{99m}Tc og ^{133}Xe . For at tage et eksempel så kan den sidstnævnte isotop



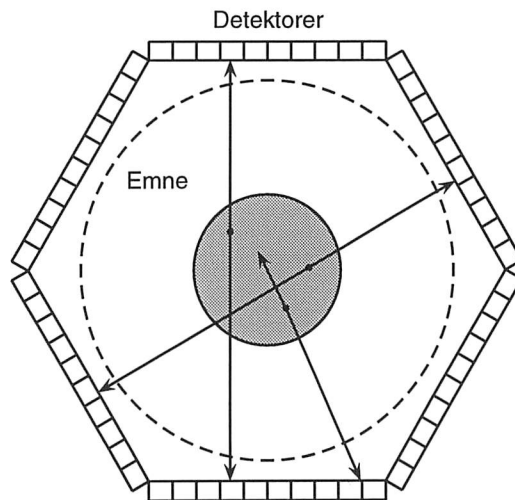
Figur 3.7: Skematisk tegning af gantryet på en SPECT-skanner.

indåndes hvorved den optages i blodet. Derved kan man studere blodgennemstrømningen af et udvalgt organ, for eksempel hjernen [Holm 1988].

3.3.10 Forenklet virkemåde. På figur 3.7 ses en principskitse af gantryet på en typisk SPECT-skanner. I midten ses emnet der skannes. Herfra udgår strålingen der udsendes i alle mulige retninger (osse ud af planet der er afbildet). For at blive i stand til at måle linieintegraler er der foran detektorerne monteret en kollimator der typisk er en blyklods der er tyk nok til at stoppe strålingen, men som samtidig er gennemboret af huller. En γ -foton kan kun trænge igennem hvis den rammer et hul vinkelret på kollimatoren. Dette har den uheldige effekt at en stor del af de udsendte γ -fotoner aldrig registreres — typisk vil kun én ud af 10.000 nå frem til detektoren [Budinger et al. 1979].

3.3.11 PET anvender for eksempel isotoperne ^{11}C , ^{13}N , ^{15}O , ^{18}F og ^{82}Rb der har det til fælles at de henfalder under udsendelse af en positron. Positronen annihileres med en nærliggende elektron og der udsendes to γ -fotoner hver med en energi på 511 keV i modsatte retninger langs en linie gennem punktet for annihilationen. De involverede isotoper gør at man for eksempel kan studere metabolisme i modsætning til SPECT: her har man endnu ikke haft held til at binde en SPECT-isotop til et stof der kan anvendes til måling af metabolismen [Gustafson 1985].

3.3.12 Forenklet virkemåde. På figur 3.8 ses en principskitse af gantryet på en typisk PET-skanner. I midten ses emnet der skannes. Herfra udgår strålingen der udsendes i alle mulige retninger, men altid som par af fotoner i modsat retning. Bevæger to sådanne parrede fotoner sig i planet defineret af gantryet, og når de begge frem til detektorerne, vil de blive registreret samtidig. Detektorerne er forbundet så samtidige registreringer måles, og hvert par af detektorer måler derved et linieintegral over forbindelseslinien mellem dem, og dette benyttes til konstruktion af et tomogram. Man behøver ikke nogen reel kollimation, men opnår i stedet noget tilsvarende ved en elek-



Figur 3.8: Skematisk tegning af gantryet på en PET-skanner.

tronisk teknik som dog, skal det tilføjes, i praksis er ganske kompliceret. Fordelen er at der ikke, som ved SPECT, mistes signal i en blykollimator.

Skanneren vist på figuren er af den „moderne“ type hvor man har detektorer hele vejen rundt. En enklere udgave vil blot have to rækker af detektorer overfor hinanden. Dette system vil så kunne roteres på samme måde som ved CT og SPECT.

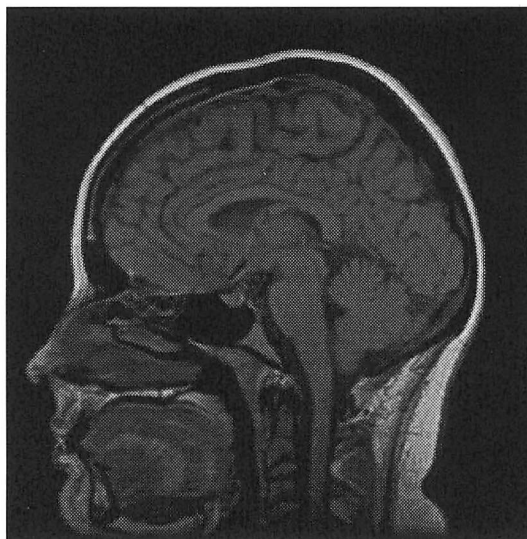
3.3.13 Tilgængelighed af isotoper. Isotoperne anvendt i PET har til forskel for isotoperne anvendt i SPECT en meget kort halveringstid af størrelsesordenen 1–10 minutter. Man må derfor fremstille isotoperne ved skanneren, og dette kræver en cyclotron samt avanceret udstyr til oprensning af de fremstillede isotoper så de er ugiftige for patienten. Alt dette er meget omkostningskrævende.

3.4 Kernemagnetisk resonans

Inden for kemien har man længe benyttet *Nuclear Magnetic Resonance spectroscopy* og kaldet *NMR-spektroskopi* til at bestemme sammensætningen af en ukendt blanding. Man kan tænke sig at patienten er den ukendte blanding, og ved at anvende tomografi bliver man i stand til opnå viden om den rumlige fordeling af de forskellige typer væv mennesket er opbygget af. Man regner teknikken for ganske ufarlig, og da kvaliteten af tomogrammerne er overordentlig høj har teknikken et stort potentiale.

3.4.1 MRI er den betegnelse man normalt benytter om denne teknik. Det er en forkortelse af *Magnetic Resonance Imaging*. Ordet „nuclear“ er faldet ud af hensyn til patienterne idet man tror de er „bange“ for det ord.

3.4.2 Anvendelse. MRI er velegnet til at skelne mellem forskellige typer af blødt væv. På figur 3.9 ses et eksempel på et MRI-skan gennem hovedets symmetriplan. Man ser mange tydelige strukturer i det bløde væv. Knoglen er derimod ikke så veldefineret som på et tilsvarende røntgentomogram.



Figur 3.9: Et T_1 -vægtet MRI-tomogram lodret gennem hovedets symmetriplan (koronalt).

3.4.3 Kernespin. Lissom elektronen har protonen et spin. Man kan tænke på protonen som en lille magnet. Befinder protonen sig i et magnetfelt kan den orientere sig på to måder der svarer til „spin op“ og „spin ned“. Energiforskellen mellem disse to tilstande er

$$\Delta E = g\mu_N B_0$$

hvor μ_N er kerne magnetonen, g det gyromagnetiske forhold og B_0 størrelsen af magnetfeltet. For protonen gælder der at

$$\mu_N = \frac{e\hbar}{2m_p} = 5,05082 \times 10^{-27} \text{ JT}^{-1},$$

$$g = 5,588.$$

Her betegner e elementarladningen, $\hbar$ Plancks konstant divideret med 2π og m_p protonens masse.

3.4.4 Resonansfrekvens. Energiforskellen mellem de to tilstande svarer til en bestemt frekvens kaldet *Larmor frekvensen* givet ved

$$\nu_L = \frac{\Delta E}{2\pi\hbar} = \frac{g\mu_N}{2\pi\hbar} B_0 = 42,59 \times 10^6 \text{ T}^{-1} \text{ s}^{-1} B_0.$$

Hvis magnetfeltets styrke er 1 T så er Larmor frekvensen $42,59 \text{ MHz}$ hvilket er i radioområdet. Anbringes protonen i et strålingsfelt med denne frekvens er der en stor sandsynlighed for at den vil skifte tilstand — populært sagt foretage et „spin-flip“. Ved et passende forsøg hvor man varierer feltets frekvens ν vil man ved $\nu = \nu_L$ observere resonans og frekvensen kaldes osse for *resonansfrekvensen*.

3.4.5 Ligevægt. Et stort antal hydrogenatomer i termisk ligevægt vil have næsten lige mange kerner i hver tilstand med en lille overvægt i tilstanden med lavest energi. Man kan bringe atomerne ud af ligevægt ved at påtrykke en elektromagnetisk impuls med resonansfrekvensen. Når pulsen ophører vil systemet igen søge imod ligevægt under udsendelse af stråling med en frekvens der netop svarer til resonansfrekvensen. En del af energien der er tilført med pulsen afgives dog på anden måde gennem *spin-gitter* og *spin-spin* relaksation osse kaldet *longitudinal* og *transversal* relaksation.

3.4.6 Lokale variationer. Hydrogenatomerne er bundet kemisk i et molekyle og derfor vil resonansfrekvensen afhænge af det *lokale* magnetfelt omkring hydrogenatomet. Dette felt er sammensat af det (kraftige) ydre felt man påtrykker samt af feltet fra nærliggende atomer. Denne effekt er målelig, og forskellige kemiske substanser får karakteristiske „fingeraftryk“. Det er denne effekt man benytter i traditionel NMR-spektroskopi. Hvordan hydrogenatomet er bundet påvirker osse hvordan relaksationen forløber, og det er især det man forsøger at måle i MRI.

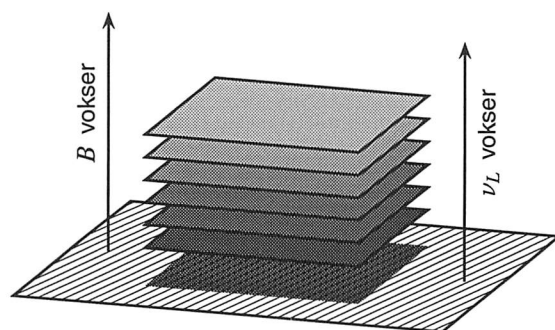
3.4.7 Spin-ekko. I MRI sendes en radioimpuls gennem det materiale man ønsker at undersøge. Det signal man ønsker at måle er meget svagt sammenlignet med pulsen (typisk en faktor 10^{-18}) så derfor stoppes pulsen og man lytter efter et signal — et såkaldt *spin-ekko*. Den energi der er tilført fra pulsen afgives i en efterfølgende relaksationsperiode. Detaljerne for hvordan denne relaksation foregår er komplicerede, men hovedsagen er at man kan måle tætheden af hydrogenatomerne n og to karakteristiske tider T_1 og T_2 som svarer til longitudinal henholdsvis transversal relaksation. For mennesket gælder der at T_1 og T_2 er karakteristiske størrelser for forskellige typer væv. Derimod er n stort set konstant igennem legemet.

3.4.8 T_1 - og T_2 -vægtning. Den størrelse der måles er en blanding af størrelserne T_1 , T_2 og n . Ved at varierer hvordan impulserne påtrykkes kan man få forskellige kombinationer af disse størrelser. Da n normalt ikke er så interessant koncentrerer man sig derfor om at lave såkaldte T_1 -vægtede og T_2 -vægtede målinger hvor billedpunkterne i skannebilledet hovedsageligt afhænger af T_1 henholdsvis T_2 relaksationstiderne.

Går man lidt mere i detaljer viser det sig osse at man har mulighed for at undersøge de processer der foregår i vævet, for eksempel kan man studere *diffusion*.

3.4.9 Planintegraler. Man ønsker at bestemme billedværdien som funktioner af stedet r . Dette gøres ved at tilføje et svagt „gradientfelt“ $\vec{B}$ til det kraftige felt B_0 så feltet bliver inhomogent. Påtrykker man en impuls med en veldefineret frekvens ν (det vil sige en lang impuls) er det kun hydrogenatomerne i et bestemt plan der vil resonere, og det signal man måler bliver derfor et planintegral over dette plan. Dette er illustreret i figur 3.10. Ved at variere frekvensen kan man måle forskellige integraler over forskellig planer. Der er dog det problem at man skal vente på at termisk ligevægt indfinder sig før man kan måle over et nyt plan, og denne tid kan være op til en størrelsesorden længere end den tid det tager at udføre målingen. Man kan i stedet påtrykke en impuls som indeholder et bredt spektrum af frekvenser (det vil sige en kort impuls), og man kan dermed måle integralerne over mange planer samtidig.

3.4.10 Radon rekonstruktion. Et enkelt planintegral er et punkt i det tredimensionale Radon rum $\mathbb{Y}_3$. Måler man samtidig over flere planer får man en diskret måling af en hel linie i $\mathbb{Y}_3$. Ved at ændre retningen af „gradientfeltet“ kan man få målinger



Figur 3.10: Illustration af hvorledes man måler forskellige planintegraler i MRI ved at variere resonansfrekvensen ν_L . Feltet B er sammensat af det konstante felt B_0 og „gradientfeltet“ γB som vokser langs én akse.

af forskellige linier og på den måde kan man registrere Radon transformationen på hele $\mathbb{Y}_3$. Man kan nu anvende tredimensionalt filtreret tilbageprojektion til at konstruere et tredimensionalt billede af det skannede.

Alternativt kan man ved at manipulere med „gradientfeltet“ under og efter impulsen opnå at måle en række linieintegraler i et udvalgt plan. Herefter kan man anvende todimensional tilbageprojektion til at konstruere et tomogram af det udvalgte plan.

3.4.11 Skannetid. Da man som sagt skal vente på at termisk ligevægt indfinder sig mellem hver måling er det ikke muligt at „filme“ med MRI, men den nyeste udvikling går i retning af meget kortere skannetider hvorved man måske en dag kan registrere en række øjebliksbilleder af anatomen.

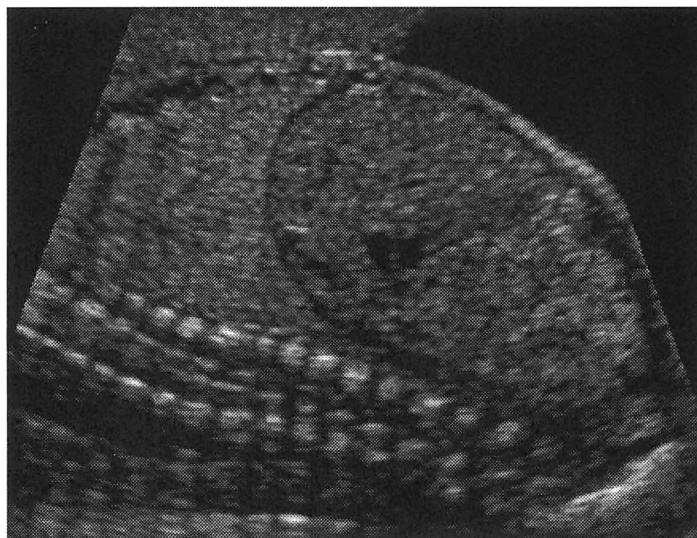
3.5 Ultralydstomografi

Ultralyd benyttes rutinemæssigt til studier af fostre hos gravide kvinder. Tidligere benyttede man såkaldte *A-* og *B-billeder* som var ganske vanskelige at fortolke med mindre man var særligt uddannet. Anvendelsen af tomografi giver mulighed for at konstruere tværsnitsbilleder, og dette er i høj grad ønskeligt da metoden er billig og regnes for ufarlig. På figur 3.11 ses et eksempel på et ultralydstomogram der viser kroppen af et foster.

3.5.1 Fordele og ulemper. Ud over et dårligt signal-støj forholdet og problemer med rekonstruktionen så er det største problem nok at knogle og luft ikke transmitterer ultralyd. Det er derfor nødvendigt at apparaturet placeres direkte på huden, og man kan ikke undersøge organer „skjult“ bag knogler.

Til gengæld kan man frit flytte den kombinerede sender og modtager som nemt kan holdes i en hånd, og det giver mulighed for at konstruere snit i mange planer. Desuden kan billedet opdateres i sand tid hvilket dem der har set en skanning af en gravid kvinde nok har haft stor fornøjelse af.

3.5.2 Fremtiden. Strammere krav fra myndighedernes side om mindske af stråledosis gør at man har brug for ikke-invasive metoder der ikke benytter ioniserende stråling.



Figur 3.11: Et ultralydstomogram der viser kroppen af et foster. Nederst ses rygraden.

MRI er en sådan metode, men er temmelig omkostningskrævende. Ultralydstomografi ser derfor ud til at være en metode der har et stort potentiale da den opfylder kravet om ufarlighed samtidig med at omkostningerne er relativt lave.

3.5.3 Princip. Den umiddelbare lighed mellem lydbølger og lysbølger (røntgenstråling) gør det tænkeligt at det er muligt at anvende tomografi i forbindelse med ultralyd — og dog. Absorptionen af røntgenstråler er i princippet givet ved det enkle udtryk (2.1), men så nemt slipper man ikke når det drejer sig om lydbølger. Man kan opstille en generel bølgeligning, men den er helt ubrugelig til praktiske formål. I stedet må man foretage visse forenklinger og approksimationer. Hvis man blandt andet antager at tætheden af det stof man undersøger er stort set konstant, at lydbølger tilnærmelsesvis udbreder sig i rette linier, og at de objekter man studerer er meget større end den anvendte bølgelængde, så kan man vise at den målte lydintensitet fremkommer som et linieintegral over en størrelse der involverer det refraktive indeks af mediet, og denne størrelse er karakteristisk for stoffet [Robinson og Greenleaf 1985].

3.5.4 Diffraktionstomografi. En ny type tomografi er ved at nå fra teoretikernes skrivebord til praktikernes værksteder. I *DT* for *Diffraktions-Tomografi* antager man ikke, som i sædvanlig tomografi, at bølgerne udbreder sig langs rette linier. I stedet tillader man alle former for diffraktionsfænomener. Teknikken er ideel til ultralyd, men man har det problem at man har svært ved at knytte en direkte sammenhæng mellem den størrelse man rekonstruerer og den faktiske anatomiske information man er interesseret i [Grassin et al. 1990].

3.6 Litteratur om medicinsk skanning

Alle de former for medicinsk tomografi som er beskrevet her er kort introduceret af Deans [1983].

Den tilgængelige litteratur om røntgentomografi er omfattende og mange værker om anvendelserne af Radon transformationen tager netop udgangspunkt i denne form for tomografi. Herman [1985] og Robb [1985d] beskæftiger sig specifikt med røntgentomografi.

Budinger et al. [1979] og Ell og Holman [1982] bringer grundige oversigter over emissionstomografi. Gustafson [1985] beskriver SPECT lissom Barrett [1984] bringer en kortfattet gennemgang. Holm [1988] beskriver SPECT-skanneren på Rigshospitalet indgående. PET beskrives af Ter-Pogossian [1985] og kortfattet af Shepp [1983b]. Den eksponentielle Radon transformation er beskrevet af Tretiak [1983] og Barrett [1984].

Den ovenstående beskrivelse af MRI er nødvendigvis noget overfladisk. Marr et al. [1981], Shepp [1983b], Barrett [1984] og Hill og Hinshaw [1985] giver flere detaljer. Den kvantemekaniske baggrund for kernemagnetisk resonans er kort beskrevet af Bransden og Joachain [1983].

Ultralydstomografi er tilsyneladende ikke så „populært“ som de øvrige former for tomografi hvis man skal vurdere det ud fra mængden af tilgængelige publikationer om emnet. Jeg kan dog henvise til [Schomberg 1981, Robinson og Greenleaf 1985, Grassin et al. 1990].

For alle disse emner gælder det at de er i en rivende udvikling, og der publiceres derfor hele tiden nyt materiale om de seneste fremskridt, lissom der produceres nye og mere avancerede kommercielle systemer.

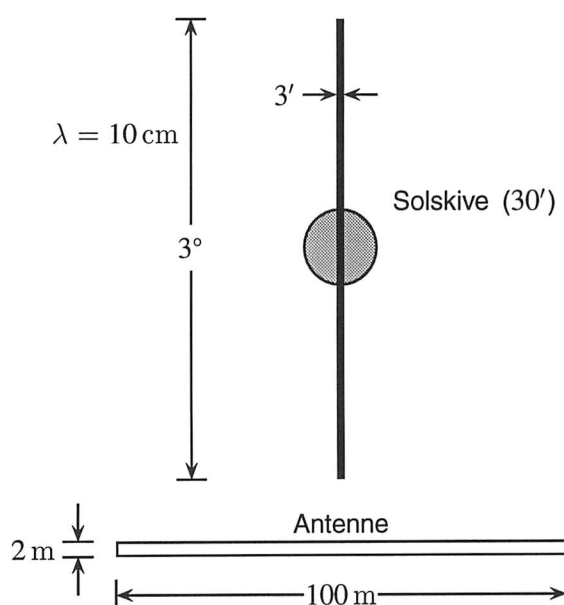
3.7 Tomografi inden for andre områder

I dette afsnit vil nogle af de talrige anvendelser af teknikker kendt fra tomografi som ikke vedrører medicin blive opsummeret.

3.7.1 Stjerners hastighedsfordeling. En af de tidligste anvendelse af tomografi var i 1936 hvor den armenske astronom V. A. Ambartsumian forsøgte at bestemme stjernernes hastighedsfordeling i rummet. Man kan for de fleste stjerners vedkommende kun bestemme deres radiale hastighed, men ved at se i forskellige retninger i rummet kan man betragte problemet som et tredimensionalt rekonstruktionsproblem og bestemme stjernernes hastighedsfordeling. Ambartsumian løste problemet og nåede til de samme resultater som Radon. Han lavede tilmed nogle praktiske beregninger ud fra astronomiske data og var dermed den første som benyttede tomografi i praksis (længe før computerens indtog) [Cormack 1983, Barrett 1984].

3.7.2 Radioastronomi. Inden for radioastronomien har man altid søgt mod større og større opløsningsevne. Uheldigvis hænger opløsningsevnen sammen med forholdet mellem antennestørrelse og bølgelængde, og radiobølger er langbølgede sammenlignet med lys. Inden for radioastronomien står man derfor ofte i den situation at man ønsker sig et radioteleskop med en antennediameter på 200 m, og den slags er ikke så lette at opdrive.

I stedet for en stor cirkulær antenne kan man benytte en aflang antenne (se figur 3.12). Den aflange antenne har den egenskab at den har en stor opløsningsevne på langs, mens den på tværs stort set ingen opløsningsevne har. Retter man antennen mod for eksempel solen vil det signal antennen modtager svare til et linieintegral hen over solskiven. Som dagen går vil solen bevæge sig over himlen, og man kan få mange



Figur 3.12: Principskitse for opmåling af solen med en aflang antenne ved en bølgelængde på $\lambda = 10 \text{ cm}$.

linieintegraler langs en bestemt retning uden at bevæge antennen. Dernæst kan man dreje hele arrangementet og få en ny serie af linieintegraler langs en ny retning. Gøres dette tilstrækkelig mange gange har man til sidst nok data til at benytte metoder kendt fra tomografi til at bestemme fordelingen af radiointensitet hen over solens overflade.

I praksis realiseres den aflange antenne ved en serie af mindre antenner som opstilles på linie. Dette er kun ét eksempel på hvordan man inden for radioastronomien, uafhængigt af udviklingen af røntgentomografi, har udviklet teknikker baseret på Radon transformationen. Disse metoder forbedrer opløsningsevnen af radioobservationer med minimale omkostninger bortset fra en forøget observationstid [Bracewell 1979, Deans 1983].

3.7.3 Modellering af solens korona. Man har igennem de sidste tyve år indsamlet en stor mængde observationer af solens korona blandt andet fra *Skylab* rumstationen. Man har nu ved anvendelse af tredimensionale rekonstruktionsteknikker været i stand til at modellere og rekonstruere magnetfeltet og elektrontætheden i koronaen [Altschuler 1979].

3.7.4 Elektronmikroskopi. Parallelt med udviklingen af medicinsk tomografi har man udviklet en tilsvarende form for tomografi inden for elektronmikroskopi [Deans 1983].

3.7.5 Optik. Hovedparten af de anvendelser af tomografi beskrevet her har et slægtskab med optik idet man observerer hvordan en form for stråling passerer gennem et medie. Inden for holografisk interferometri dukker Radon transformationen derfor naturligt op [Deans 1983]. Osse Barrett [1984] nævner anvendelser inden for bølgeudbredelse og mikrobølgespredning.

3.7.6 Spredningsforsøg. Foruden mikrobølgespredning beskriver Barrett [1984] osse sammenhængen mellem Radon transformationen og Compton spredning, og Hejzmanek

[1981] beskriver sammenhængen mellem lineær transportteori, matematisk spredningsteori og Radon transformationen.

3.7.7 Seismologi. En anden form for „optik“ er seismologi. Her er lysbølgerne erstattet med lydbølger der bevæger sig gennem undergrunden. Tomografi har været anvendt til at konstruere såkaldte *seismiske profiler* [Deans 1983].

3.7.8 Elektrostatik. Den inverse Radon transformation kan skrives i en form der involverer Laplace operatoren. Det er derfor ikke overraskende at Radon transformationen dukker op inden for elektrostatikken [Barrett 1984].

3.7.9 Plasmafysik. Inden for plasmafysik har man nogle problemer med at montere detektorer i eksperimentområdet og man har her set nytten af tomografi [Niland 1982]. Specifikt er man nu i stand til at bestemme hastighedsspektret af pulser af partikler når partikelkilden varierer i tiden. Dette kaldes for *time of flight analysis* [Niland og Schmidt-Harms 1985].

3.7.10 Tekniske anvendelser af tomografi. Anvendelse af røntgentomografi på metal-komponenter kan afsløre små sprækker og lignende inden i komponenten. Dette kan bruges til en ikke-invasiv kontrol af vitale maskindele, for eksempel fra jetmotorer eller turbiner. På den måde kan ulykker forebygges uden at man behøver at udskifte dele unødigt ofte.

3.7.11 Raketter. Specielt kan nævnes at man har benyttet en særlig teknik relateret til tomografi til at foretage materialekontrol på raketsnuder og -motorer [Quinto 1990].

3.7.12 Stressanalyse. Deans [1983] nævner at Radon transformationen osse optræder når man forsøger at modellere små sprækker inden for elastostatikken.

3.7.13 Matematik. Inden for matematikken har man anvendt Radon transformationen til så forskellige ting som teorien for partielle differentialligninger og statistik [Cormack 1983, Deans 1983].

3.7.14 Fremtidsvisioner. Der har været mange forslag om potentielle anvendelser af Radon transformationen [Deans 1983]: Lige fra overvågning af luftforureningen med lasere, over billedtransmission og mønstergenkendelse i billeder, til måling af snedybder med henblik på at forudsige laviner [Cormack 1980].

Den mest visionære idé er måske at studere det indre af planeter ved at observere hvorledes neutrinoer fra solen transmitteres [Deans 1983] — en direkte udvidelse af røntgentomografi. Find selv på flere anvendelser!

Kapitel 4

Visualisering af tredimensionale data

- 4.1 Indledning
- 4.2 Visualisering af konturflader
- 4.3 Rumfangsgengivelse
- 4.4 Litteratur om rumfangsgengivelse
- 4.5 Konstruktion af konturflade

I dette kapitel bringes der en oversigt over forskellige metoder hentet fra litteraturen til visualisering af tredimensionale data. Udgangspunkt vil være medicinske data erhvervet ved tomografi.

Først introduceres en *konturflade* som er en tilnærmelse til en *ækvipotentiaflade*. Denne konturflade kan visualiseres på to måder: ved *rumfangsgengivelse* og ved at konstruere konturfladen som derefter visualiseres. De næste kapitler fokuserer på hvordan en konturflade kan konstrueres så her benyttes plads til at beskrive rumfangsgengivelse. Derefter fortsættes med en beskrivelse af metoder hentet fra litteraturen til konstruktion af en konturflade. Dette tjener som en optakt til anden del af dette speciale som handler om en bestemt type af disse algoritmer.

4.1 Indledning

4.1.1 Todimensional visualisering er ligetil idet man har en måling af for eksempel lysintensitetsværdier registreret regelmæssigt på et rektangulært område. Ved at tilordne gråtoner til intensitetsværdierne — lyse ved høj intensitet og mørke ved lav intensitet — kan man visualisere målingen ved at vise de tilhørende gråtoner på en computerskærm. I stedet for gråtoner kan farvetoner tilordnes. Hvis målingen er en temperaturmåling, for eksempel foretaget af en satellit, kan kolde områder afbildes som blå og varme områder som røde.

4.1.2 Tredimensionale data. Antag at anatomien af en patients hoved ønskes studeret med røntgen. Et enkelt vandret røntgentomogram giver et tværsnit af hovedet og det kan studeres som beskrevet ovenfor. Ved at skanne flere gange er man i stand til at bestemme en række tværsnit fra hagespids til isse ved at variere placeringen af snitplanet. Dette giver en hel serie af tværsnit som til sammen giver et godt indtryk af hovedets anatomi. Dette kræver dog at afstanden mellem de enkelte tomogrammer ikke er meget større end opløsningen i hvert tomogram.

4.1.3 Rumfang. Denne form for data kaldes for et *rumfang* eller på engelsk *volume*. Man kan tænke på et rumfang som en slags tredimensionalt billede som består af billedpunkter kaldet *voxels* (i stil med *pixels* — *picture elements* — i almindeli-

ge, todimensionale billeder). Men i modsætning til det todimensionale billede kan det tredimensionale billede ikke direkte visualiseres på en computerskærm.

4.1.4 Tærskelværdi. Betragtes det enkelte røntgentomogram i rumfanget ses det at knoglevæv fremtræder som „lyse“ områder. Det betyder at man skal nøjes med at studere de „lyse“ områder hvis man vil studere kraniet. Dette gøres med en *tærskelværdi* — erfaringen viser at billedpunkter med en værdi på over cirka 150 HU svarer til knoglevæv.

4.1.5 Grænseflade. Betragtes alle tværsnittene under ét, det vil sige hele rumfanget, gælder der at man kan lokalisere overfladen af kraniet som de områder i rumfanget hvor røntgenabsorptionen skifter fra under til over tærskelværdien. Et af de væsentlige emner for dette speciale er hvordan man konstruerer denne flade fra et givet rumfang og en givet tærskelværdi. Den konstruerede flade svarer netop til overfladen af knoglen og er derfor betydningsfuld i studiet af hovedets anatomi. Ved at ændre tærskelværdien kan man bestemme grænseflader mellem andre vævstype og organer.

4.1.6 Kantdetektion og klassifikation. Mere generelt må man foretage en tredimensional kantdetektion og klassifikation af områder. At benytte en tærskelværdi til dette er nok den enkleste metode og den virker godt til detektion af knogler i røntgentomogrammer. Derimod må man ofte i MRI benytte mere avancerede metoder. Dette er dog ikke det centrale emne for dette speciale så jeg vil for enkelhedens skyld i den følgende fremstilling tage udgangspunkt i at en tærskelværdi benyttes.

4.1.7 Datamængden i medicinske skanninger gør at arbejdet med disse er tidskrævende med mindre man råder over meget hurtige computere. Et „typisk“ rumfang kan bestå af cirka $200 \times 200 \times 100 = 4.000.000$ billedpunkter. De fleste rumfang jeg har arbejdet med har omtrent denne størrelse. Skanner man hovedet af et barn i en lidt ældre CT-skanner af den type som står på mange danske sygehuse vil man typisk få data med denne opløsning. Mere moderne skannere er ofte i stand til at fordoble den lineære opløsning således at datamængden forøges otte gange.

4.2 Visualisering af konturflader

4.2.1 Ækvipotentiaflade. Man kan betragte rumfanget som en måling af et tredimensionalt felt på et rektangulært gitter, og det som ønskes studeret er en ækvipotentiaflade defineret ved en tærskelværdi. Dette problem dukker op inden for mange områder af naturvidenskaben. For eksempel kan feltet beskrive elektrontætheder i et atom og den ønskede ækvipotentiaflade kan være en udvalgt orbital.

4.2.2 Konturflade. Da feltet er registreret på et diskret antal billedpunkter så vil feltet næsten aldrig antage tærskelværdien. I stedet vil der være nogle billedpunkter hvor feltværdierne er under tærskelværdien og nogle billedpunkter hvor feltværdien er over tærskelværdien. Det vil derfor kun være muligt at bestemme en flade der adskiller billedpunkter som er *over* tærskelværdien med billedpunkter der er *under*. Den type flade vil her blive betegnet som en *kontur*, eller en *konturflade* når det skal fremhæves at det er en flade. Tilsvarende kan man så benytte betegnelsen *konturkurve* om en kontur i et todimensionalt billede.

Der gælder at en konturflade for en given tærskelværdi er en god tilnærmelse til ækvipotentialfladen defineret af samme tærskelværdi.

4.2.3 Metoder. Der findes mange måder hvorved man kan lokalisere og visualisere konturflader. De kan deles i to kategorier:

1. Direkte skankonvertering (afbildning på computerskærm) af data osse kaldet *rumfangsgengivelse* eller *volume rendering* på engelsk.
2. Konstruktion af en model af konturfladen. Denne flade kan senere skankonverteres eller benyttes til konstruktion af en fysisk model i for eksempel kunststof.

4.2.4 Rumfangsgengivelse er den metode hvor det endelige billede af konturfladen konstrueres direkte fra de oprindelige data. Der findes algoritmer som er ganske hurtige — på en „typisk“ arbejdsstation (for eksempel med en ydelse på mindst 10 MIPS) kan et billede konstrueres på størrelsesordenen 10 sekunder. Kvaliteten af billedet kan naturligvis forbedres ved at anvende mere avancerede og tidskrævende algoritmer.

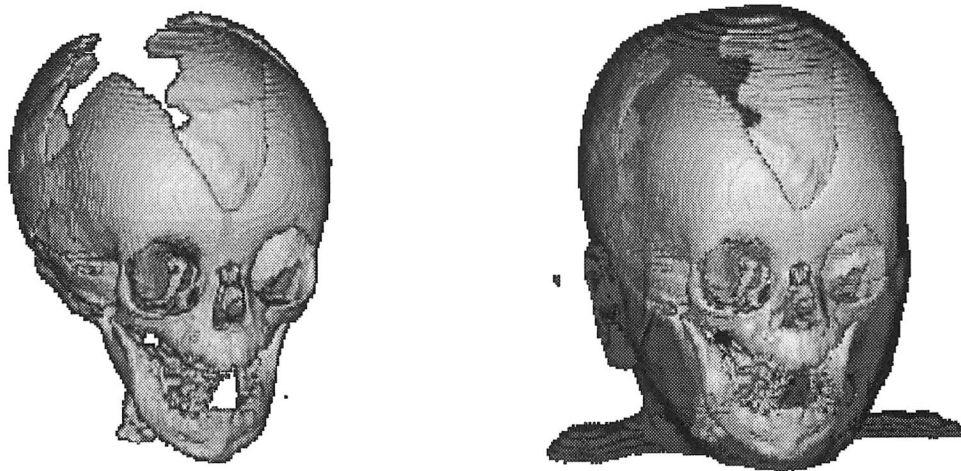
Ønsker man at konstruere et nyt billede set fra en anden vinkel er det i reglen ikke muligt at genanvende resultater fra tidligere beregninger, og hvert nyt billede vil tage samme tid at konstruere (der findes algoritmer der genanvender resultater fra tidligere beregninger, men det er ofte på bekostning af et stort forbrug af lager). Metoden er derfor endnu ikke så velegnet til visualisering og manipulation af rumfang i sand tid med mindre man har temmelig kraftige computere til rådighed.

4.2.5 Konstruktion af konturflade. I stedet for direkte at danne billedet af konturfladen kan man konstruere en mellemliggende repræsentation af fladen som så efterfølgende kan visualiseres. Fladen skal repræsenteres af en form for geometrisk primitiv og her benytter man oftest trekanter (der altid er plane). Der findes talrige teknikker til at visualisere sådanne flader, og det er for eksempel ganske enkelt at foretage en ændring af betragterens placering [Foley et al. 1990, Mortenson 1989]. Man kan osse introducere mere avancerede effekter som, for eksempel, perspektiv, egenskaber ved materialet, lyskilder og så videre.

Antallet af geometriske primitiver, der benyttes til at beskrive fladen, er dog i reglen så stort at beregningen af et enkelt billede på en kraftig grafisk arbejdsstation tager af størrelsesordenen 10–100 sekunder, og sand tids manipulation er derfor ikke praktisk muligt. Men med den rivende udvikling der sker inden for computerindustrien vil sand tids eller nær sand tids manipulation sikkert snart være mulige.

For eksempel er *Reality Engine*² fra *Silicon Graphics* angiveligt det hurtigste serieproducerede system til tredimensional grafik. Ydeevnen er omkring 900.000 tredimensionale trekanter per sekund inklusiv *tekstur* og *anti-aliasing*. Derved skulle det være muligt at opnå adskillige opdateringer per sekund af skærbilledet for typiske flader fra medicinske data.

4.2.6 Animation. Da almindeligt tilgængelige arbejdsstationer i dag endnu ikke er så effektive at man med de ovenfor beskrevne metoder kan opdatere det viste billede i sand tid, kan man i stedet forberede en animation ved at lave en serie billeder hvor man for eksempel flytter betragterens placering. Disse billeder kan gemmes til senere brug hvor de kan vises i hurtig rækkefølge for at give et indtryk af bevægelse. Brugeren af systemet, for eksempel en læge, vil ofte få en bedre rumlig opfattelse af de givne data når den slags „tricks“ benyttes [Robb og Barillot 1989].



A) Knogle.

B) Knogle og transparente bløddede.

Figur 4.1: Eksempler på rumfangsgengivelse lavet med ANALYZE programmet.

4.2.7 Fysisk model. En flade repræsenteret på en computer kan anvendes til styring af fræseværktøj og lignende så det bliver muligt at få en model i kunststof af det objekt man ønsker at modellere. Især *stereolitografi* ser ud til at være en lovende metode til konstruktion af selv komplicerede modeller af for eksempel knoglestrukturer. Sådanne modeller kan være meget nyttige til fremstilling af proteser og som hjælp til kirurgisk behandlingsplanlægning [Kreiborg et al. 1992b, Swaelens 1992, Ruud 1992].

4.2.8 Stereolitografi er kort fortalt en relativ ny teknik hvor man ved hjælp af to skærende laserstråler er i stand til selektivt at hærde en særlig kunstharpiks i et kar. Dette gør det muligt at konstruere vilkårligt komplicerede modeller som tilmed kan indeholde hulrum. Fem-akse fræsere, som traditionelt har været anvendt til nogle af de samme arbejdsopgaver, har mange begrænsninger sammenlignet med denne metode. Man arbejder for tiden på at designe bio-kompatible kunstharpikser som kan anvendes til direkte at fremstille særligt tilpassede proteser og implantater fra medicinske data.

4.3 Rumfangsgengivelse

4.3.1 Skankonvertering er den disciplin inden for *computer graphics* hvor man for en given geometrisk model og en given betragter af denne model beregner et billede af modellen set af betragteren. Man beskæftiger sig med hvordan man mest effektivt kan konvertere et *geometrisk primitiv* til intensitetsværdier på en computerskærm. De geometriske primitiver kan være ganske enkle, så som punkter og trekanter, eller mere komplicerede objekter, for eksempel såkaldte *NURBS flader*. Rumfangsgengivelse er den disciplin der beskæftiger sig med hvordan en voxel som geometrisk primitiv kan konverteres til en intensitetsværdi på computerskærmen. På figur 4.1 ses eksempler på rumfangsgengivelse. Den anvendte metode må karakteriseres som værende relativt hurtig — dette opnås naturligvis på bekostning af kvaliteten af billedet.

4.3.2 Metoder. Der findes mange forskellige algoritmer beskrevet i litteraturen, og alle disse har som regel hver for sig et særligt kendetegn der adskiller den pågældende algoritme fra de øvrige. Disse algoritmer kan inddeles i følgende grupper:

1. Voxel projektion hvor de enkelte voxels projiceres på skærmen. Dette kan gøres på to måder:
 - (a) Bagfra (*back to front*) hvor man starter med at projicere den fjerneste voxel. Efterhånden vil foran liggende voxels dække over de bagved liggende.
 - (b) Forfra (*front to back*) hvor man starter med at projicere den nærmeste voxel. Efterhånden som bagved liggende voxels skjules af allerede projicerede voxels ses der helt bort fra disse.
2. Strålekastning (*ray tracing*) hvor „synsstråler“ udgår fra en betragter og kastes ind i rumfanget indtil de enten rammer et objekt eller fortsætter ud af rumfanget på den anden side.

4.3.3 Klassifikation. Fælles for disse metoder er at de enkelte voxels klassificeres i flere kategorier. Den enkleste klassifikation fås, som tidligere nævnt, ved at benytte en tærskelværdi hvorved hver voxel kan klassificeres som hørende med til objektet defineret af tærskelværdien eller som ikke hørende med. Kun de voxels der hører med til objektet bidrager til det endelige billede.

4.3.4 Transparente voxels. I stedet kan værdien af hver enkelt voxel benyttes til at bestemme dens farve og opacitet, og ved at integrere langs synsstråler kan man gengive de forskellige dele af rumfanget i forskellige farver, nogle dele transparent foran andre (se figur 4.1.B). Man kan tilmed lade voxels med bestemte værdier udsende lys hvorved rumfanget oplyses indefra.

Set fra en brugers synsvinkel er en af de største forskelle mellem rumfangsgengivelse og visualisering af flademodeller netop muligheden for at anvende transparente voxels. Dette giver mulighed for at studere samspillet mellem forskellige vævstyper i stedet for at man skal begrænse sig til én type væv ved at fastlægge en tærskelværdi.

Derfor kan man med en vis ret sige at „rigtig“ rumfangsgengivelse skal give mulighed for at anvende transparente voxels. Forenklet kan man hævde at set fra brugerens synspunkt, så er rumfangsgengivelse, der ikke har den funktionalitet, ikke til at skelne fra algoritmer der konstruerer en konturflade som derefter visualiseres.

4.3.5 Voxel projektion bagfra. Alle voxels i rumfanget gennemløbes startende med den voxel der er længst væk fra betragteren og sluttende med den voxel der er nærmest betragteren. Hver voxel som hører til objektet projiceres på skærmen, og eftersom de gennemløbes på den netop beskrevne måde vil foran liggende voxels dække for bagved liggende voxels som ønsket. Metoden kræver dog at samtlige voxels gennemløbes selv om kun de voxels i objektet som er nærmest betragteren bidrager til det endelige billede.

4.3.6 Voxel projektion forfra. I stedet kan man gennemløbe alle voxels startende med den voxel der er nærmest betragteren. Efterhånden som man møder voxels som hører til objektet vil deres projektion på skærmen skjule bagved liggende voxels som man ikke behøver at undersøge. Dette giver en mere effektiv algoritme som dog er mere

komplikeret at implementere. Ønsker man at implementere transparente voxels vil opaciteten langs synsstrålerne efterhånden nå den værdi der svarer til ugenomsigtig hvorved bagved liggende voxels skjules og ikke derfor ikke behøver at blive undersøgt.

4.3.7 Strålekastning. Man kan i stedet tage udgangspunkt i betragteren. Fra dennes raster af billedpunkter kastes stråler ind i rumfanget og man bestemmer hvilke voxels de rammer. Benyttes en tærskelværdi til klassifikation vil strålen enten ramme en voxel over tærskelværdien hvorved den stopper, eller osse vil den fortsætte ud af rumfanget på den anden side.

Ønsker man at inddrage transparente voxels så forløber beregningen af farve- og intensitetsværdien af hver stråle som en integration over strålen hvorved de enkelte voxels bidrager til den endelige værdi. Integration stopper når opaciteten er nået op på ugenomsigtig eller når strålen forlader rumfanget på den anden side.

4.3.8 Rotation af rumfang. De netop beskrevne metoder tillader at betragteren befinder sig i vilkårligt „skæve“ vinkler i forhold til rumfanget, men det er klart at algoritmerne forenkles væsentligt hvis synslinierne er parallelle med en af de tre principale akser. I stedet for at udvikle en algoritme der kan håndtere vilkårlige placeringer af betragteren kan man i stedet rotere rumfanget ved at interpolere et nyt rumfang på et nyt gitter der er roteret i forhold til det oprindelige. Dette kan være en tidskrævende proces, men visse muligheder for optimering er mulige [Hanrahan 1990]. En integration langs syns-linier som anvendes i forbindelse med strålekastning involverer i realiteten en implicit interpolation af rumfanget på et gitter defineret af synslinierne.

4.3.9 Interpolation af rumfang. Mange algoritmer forudsætter at voxels er kubiske, det vil sige har ens dimensioner langs alle tre akser. Ofte er dette ikke tilfældet da data som regel stammer fra tomogrammer hvor afstanden mellem hvert tomogram er større end opløsningen i tomogrammet. Man kan da interpolere et nyt rumfang med kubiske voxels, og man benytter næsten altid lineær interpolation da det er det enkleste.

Antagelsen om lineær variation mellem de enkelte tomogrammer strider dog imod den observation at et tomogram ofte kan inddeles i forskellige områder med stort set samme intensitetsværdierne svarende til for eksempel forskellige organer. Hvert område er karakteriseret ved en rimelig konstant intensitetsværdi, mens områder indbyrdes er differentieret ved en forskel i intensitetsværdi. Herman og Bucholtz [1991] beskriver en ny interpolationsteknik baseret på *chamfer* afstanden der giver bedre resultater end når simpel lineær interpolation anvendes.

4.3.10 Aliasing. Da medicinske data som nævnt har en ulineær karakter vil værdierne i rumfanget have en tendens til at springe i stedet for at variere glat inden for rumfanget. Endvidere er data registreret på et gitter som ofte ikke engang er kubisk, men med voxels der er „aflange“, og de strukturer man ønsker at afbilde er af samme størrelsesorden som de enkelte voxels i gitteret.

Alt dette er med til introducere såkaldte *artefakter*, eller *aliasing* i det endelig billede. For eksempel vil man på grund af den manglende opløsning på tværs af tværnittene ofte se falske konturlinier svarende til hvert tværsnitsbillede med mindre man tager særlige hensyn. For at afhjælpe dette kan man lavpas filtrere rumfanget inden rumfangsgengivelsen foretages [Drebin et al. 1988]

4.3.11 Toning. Det er ikke nok at projicere de enkelte voxels på skærmen — de skal osse tilordnes en tone, det vil sige *tones* (på engelsk *shading*). Man kan benytte en række forskellige metoder og den enkleste er...

4.3.12 Afstandstoning. Afstanden fra betragteren benyttes til at beregne en gråtone — mørke toner for voxels langt væk og lyse toner for voxels tæt på. Det resulterende billede bliver et slags „højdekort“ og man benytter betegnelsen *afstandstoning* (på engelsk *depth shading*) om denne metode.

4.3.13 Belysning. I stedet kan fladen illumineres af en lyskilde for at afsløre detaljer i fladen som ikke kan ses tydeligt på „højdekortet“. Kendskab til lyskildens placering og fladens orientering i forhold til denne kan give en gråtone, for eksempel efter den såkaldte *cosinus lov*. Vektoren der beskriver lyset projiceres på fladens normalvektor og længden af denne projektion er den mængde lys der reflekteres. Man kan osse anvende mere avancerede *belysningsmodeller* som for eksempel *Gouraud* eller *Phong* modellen [Foley et al. 1990].

4.3.14 Fladenormaler. Men hvordan beregner man konturfladens normal som benyttes til toningen? Man kan bestemme konturfladen i en omegn af den voxel man ønsker at tone og så bestemme en normal til denne flade. Dette kaldes somme tider for *afstands-gradient toning* (på engelsk *depth-gradient shading*) fordi man benytter de beregnede dybdeværdier til at bestemme konturfladen og dermed til at estimere normalen. I stedet kan man bemærke at gradienten af et tredimensionalt skalarfelt beregnet i et punkt vil være vinkelret på *ækvipotentialfladen* gennem punktet. Med andre ord så kan skalarfeltets gradient normaliseres i et punkt på konturfladen hvorved den ønskede normalvektor fås. Dette er osse grunden til den lidt forvirrende betegnelse „afstands-gradient toning“ anvendt ovenfor. Egentlig skulle man sige *afstands-normal toning* da man på intet tidspunkt blander gradienten af feltet ind i beregningen.

Men som sagt så kan man få et estimat for normalen ved at beregne gradienten, for eksempel ved i hver voxel at studere værdien af de seks nærmeste nabo-voxels. Benyttes denne lokale gradient til at estimere normalen af konturfladen opnås et bedre resultat end med afstands-gradient metoden [Höhne og Bernstein 1986].

4.3.15 Binær klassifikation af voxels er et problem ved disse metoder. Enten er voxels inden for fladen eller osse er de udenfor. I forvejen kan det være svært at klassificere voxels fra medicinske skanninger med en tærskelværdi blandt andet på grund af den tidligere omtalte del-rumfangs effekt (se afsnit 3.2.8). Som antydnet kan man anvende mere avancerede algoritmer til rumfangsgengivelse hvorved man får mulighed for at afbilde nogle typer væv transparent. På figur 4.1.B er for eksempel vist et kranie overlejret med transparente bløddele. Man undgår derved at fjerne for meget information ved kun at vise én konturflade givet ved en tærskelværdi.

4.4 Litteratur om rumfangsgengivelse

Som tidligere antydnet er mangfoldigheden iblandt algoritmerne til rumfangsgengivelse stor. Ovenfor har jeg forsøgt at fremhæve nogle fundamentale fælles træk hos disse algoritmer, men har på den måde osse udeladt nogen af de interessante træk der gør hver algoritme til noget særligt. Dette vil jeg råde bod på her ved at give en lille

oversigt over nogle af de algoritmer der er beskrevet i litteraturen. Selv om oversigten omfatter adskillige referencer er den dog langt fra komplet.

4.4.1 Transparente voxels. Levoy [1988] beskriver en strålekastningsalgoritme der implementerer transparente voxels. De enkelte voxels klassificeres i for eksempel tre grupper: luft, bløddeler og knogle. Hver gruppe tildeles en farve og en opacitet. For eksempel kan luft være gennemsigtigt, mens bløddeler kan være røde og transparente. Endelig kan knogle være ugennemsigtigt, hvidt. I grænseområdet mellem for eksempel knogle og bløddeler kan voxels tildeles farve- og opacitetsværdier der er en passende kombination af værdierne for begge grupper. For at dette skal fungere må man antage at der kun er grænseområder mellem luft og bløddeler og mellem bløddeler og knogle — ikke mellem luft og knogle, men denne antagelse er jo osse ganske fornuftig. De enkelte voxels toner efter Phong modellen og farve- og opacitetsværdier af voxels langs samme synsline kombineres ved α -blanding. De resulterende billeder skulle være af en overordentlig god kvalitet da man undgår en binær klassifikation af voxels og i stedet er i stand til at vise samspillet mellem forskellige typer væv.

Upson og Keeler [1988] beskriver den såkaldte *V-BUFFER* algoritme som i realiteten er en form for voxel projektion forfra. Strålekastningsmetoder beskrives dog osse. Værdien af den enkelte voxel benyttes til at tildele den en farve- og en opacitetsværdi. I modsætning til Levoy [1988] gøres der her ikke noget særligt ud at beskrive hvordan dette gøres så forskellige vævstyper kan adskilles.

Drebin et al. [1988] beskriver en algoritme der benytter strålekastning langs en af de principale akser. Det betyder at rumfanget kun kan vises fra „skæve“ vinkler hvis det først roteres og interpoleres på et nyt gitter, men samtidig er algoritmen lavet således at en del af beregningerne kan laves før rotationerne foretages hvorved der spares tid. Forfatterne gør sig i den forbindelse nogle signalteoretiske overvejelser om gyldigheden af en sådan rotation eller reinterpolation. Til klassifikation af de enkelte voxels benyttes en fremgangsmåde meget lig den beskrevet af Levoy [1988], og der beskrives hvordan man med udgangspunkt i et histogram over voxel værdierne kan konstruere funktioner der tildeler farve- og opacitetsværdier til de enkelte voxels. Firmaet for hvilket forfatterne arbejder (PIXAR) har konstrueret en computer specielt beregnet til rumfangsgengivelse, og hardwaren er opbygget således at for eksempel den meget tidskrævende reinterpolation kan forløbe med en rimelig hastighed.

4.4.2 Partikelsystemer. Sabella [1988] beskriver en algoritme der tager sit udgangspunkt i en model for spredning af (lys)partikler som for eksempel kan benyttes til at modellere tåge og skyer. De enkelte voxels udsender partikler som spredes, og et tæthedsfelt af partiklerne defineres. Stråler kastes gennem rumfanget og forskellige størrelser fra feltet integreres over strålerne. De resulterende integraler afbildes i et HSV farvesystem. Det kan lyde kompliceret, men det væsentlige er at voxels både kan være transparente og udsende lys. Disse oplysninger kombineres til et farvebillede som skulle indeholde stærke visuelle elementer som hjælper til at beskrive det underliggende rumfangs struktur.

Krueger [1990] beskriver en algoritme i stil med den ovenfor beskrevne baseret på transport af „virtuelle“ partikler gennem rumfanget. Hver voxel bidrager som kilde og med absorption, spredning og farveskift af partiklerne. Det demonstreres hvordan begreber fra fysikkens verden hvor transportteori har mange anvendelser, kan benyttes

til at afsløre interessante ting ved rumfanget. For eksempel kan man placere en „ladet“ testpartikel i rumfanget hvorved symmetrier i rumfanget vil optræde som symmetriske mønstre der opstår på grund af tunneleffekten.

4.4.3 Optimeringer. De ovenfor beskrevne algoritmer (på nær V-BUFFER algoritmen) har alle benyttet en form for strålekastning, men dette kan være en ganske proces-søkrævende operation. Wilhelms og Van Gelder [1991] beskriver en algoritme særligt egnet til rumfangsgengivelse i sand tid som benytter projektion af voxels enten bagfra eller forfra. Den høje hastighed opnås ved at observere at alle voxels projiceres ens på skærmen og at der kun er tretten væsentligt forskellige måder hvorpå dette kan gøres. Derved kan man foretage et antal beregninger på forhånd inden samtlige voxels gennemløbes. Algoritmen giver mulighed for at afbilde selvlysende og transparente voxels på linie med andre avancerede algoritmer.

4.4.4 Hierarkisk model. Laur og Hanrahan [1991] beskriver en model hvor rumfanget er repræsenteret af et oktræ. Oktræet visualiseres ved at hver knude i træet afbildes som en „splat“ af den rette størrelse svarende til niveauet i oktræet. Jo dybere man bevæger sig ned i træet jo mindre „splats“. De enkelte „splats“ er repræsenteret ved en tæthedsfunktion, for eksempel en todimensional Gauß kurve, der giver den enkelte „splat“ et „fodafttryk“. Knuderne i oktræet gennemløbes bagfra og de enkelte „fodafttryk“ projiceres på skærmen. Til hver knude i oktræet knyttes en middelfejl som afhænger af forskellen i værdi mellem de voxels der indgår i den pågældende knude. Denne middelfejl benyttes som kriterie for hvor langt op eller ned i hierarkiet af „splats“ man skal gå. Et forholdsvis homogent område af rumfanget vil blive beskrevet med „store“ „splats“, mens mere varierende områder vil blive beskrevet med mindre „splats“.

4.4.5 Oversigter. Coatrieux og Barillot [1990] og Höhne et al. [1990a] forsøger at give en oversigt over tilgængelige metoder til såvel konstruktion af konturflader som til rumfangsgengivelse. Höhne et al. [1990a] anvender nogle forskellige algoritmer på medicinske data. Konklusionen er — ikke overraskende — at forskellige anvendelser kræver forskellige algoritmer. Der findes ingen „universal“ algoritme.

4.5 Konstruktion af konturflade

4.5.1 Metoder. Der eksisterer adskillige algoritmer til konstruktion af en konturflade og disse kan deles i to hovedklasser:

1. *Kurvebaserede* algoritmer tager udgangspunkt i at data foreligger i form af tvær-snitbilleder fra for eksempel medicinske tomogrammer. Når opløsningen af det enkelte tvær-snitbillede er stor i forhold til afstanden mellem billederne er det naturligt at konstruere en konturkurve, det vil sige en kurve der tilnærmer en ækvipotentialekurve, i hvert tvær-snitbillede. Så stykkes konturerne i hvert tvær-snit sammen på tværs af tvær-snittene til en flade. Denne type algoritme består af en del der håndterer de todimensionale problemer vedrørende konstruktion af konturerne og en del der håndterer de tredimensionale problemer vedrørende sammensætningen af konturerne til en flade.
2. *Isotropiske* algoritmer har ikke nogen udvalgt retning som den ovenfor beskrevne type tydeligvis har.

Først vil nogle kurvebaserede algoritmer kort blive omtalt hvorefter de isotropiske algoritmer vil blive beskrevet.

4.5.2 Bestemmelse af konturkurve foretages i hvert tværsnit. Forskellige kantdetektionsmetoder kan anvendes i hvert billede [Rosenfeld og Kak 1982], men det enkleste er at benytte en tærskelværdi. Det endelige resultat er en eller flere lukkede konturer i hvert tværsnit, for eksempel beskrevet som sammensætningen af en række liniestykker.

4.5.3 Fladekonstruktion. Næste trin er at forbinde konturerne i nabo-tværsnit således at der dannes en lukket flade. Konturerne kan forbindes med trekantede fladesegmenter. Af alle de mulige tesseleringer med trekanter skal man vælge én som er optimal. Man kan for eksempel maksimere det indesluttede volumen [Kepple 1975] eller minimere arealet af tesseleringen [Fuchs et al. 1977].

4.5.4 Forgreningsproblemet. To nabo-tværsnit kan have et forskelligt antal lukkede konturer. Det er et tegn på at de strukturer der er indeholdt i rumfanget forgrener sig mellem de to tværsnit, og det er af fundamental betydning for algoritmens anvendelighed at den er i stand til at forbinde konturerne korrekt. Adskillige beskrevne algoritmer er slet ikke i stand til at konstruere flader der forgrener sig og de er derfor kun anvendelige til helt enkle problemstillinger.

Boissonnat [1988] beskriver en algoritme som på basis af en særlig Delaunay triangulering er i stand til at håndtere flader der forgrener sig. Uheldigvis er der en tendens til at fladens opbygning omkring forgreningerne kan være temmelig „grov“ [Lin et al. 1989].

4.5.5 Interpolation af konturer mellem eksisterende konturer kan afhjælpe denne grovhed. Xu og Lu [1988] beskriver en algoritme der interpolerer nye konturer ud fra et kriterie om minimal krumning af den udspændte flade. Uheldigvis er algoritmen ikke i stand til at håndtere forgreninger. Lin et al. [1989] løser problemet ved at betragte interpolation af konturer som en elastisk deformation af et sæt af konturer over i et andet sæt hvorved forgrening tillades.

4.5.6 Sammenligning med isotropiske algoritmer. Det er klart at jo større afstanden mellem de enkelte tværsnit bliver (set i forhold til opløsningen af hvert enkelt tværsnit) desto vanskeligere bliver det at håndtere forgreningsproblemet. På den anden side: jo mindre afstanden mellem de enkelte tværsnit bliver jo mindre indlysende bliver det at man skal benytte en kurvebaseret algoritme frem for en isotropisk algoritme. Dog har kurvebaserede algoritmer den fordel at de er nemme at forstå da processen er delt op i to trin: konstruktion af kurver og sammensætning af kurver til flader.

4.5.7 Reduceret fladekompleksitet. En anden fordel ved kurvebaserede algoritmer er muligheden for at reducere kompleksiteten af de konstruerede flader. Chang et al. [1991] viser hvordan man kan konstruere en flade opbygget af såkaldte *Hermite flader* fra konturer i de enkelte tværsnit. Fladen vil i reglen indeholde et færre antal geometriske primitiver end en tilsvarende flade konstrueret med en isotropisk algoritme uden at man på den måde har mistet betydningsfuld information. Uheldigvis er algoritmen ikke i stand til at håndtere forgreningsproblemet.

4.5.8 Isotropiske algoritmer kommer i blandt andet disse to varianter:

1. Såkaldte *tegl-algoritmer* som vil blive beskrevet nedenfor.
2. Algoritmer der sammensætter en flade af grænseflader mellem voxels ved en *flade-sporingsproces*.

4.5.9 Tegl-algoritmer er betegnelsen for en type algoritme der betragter rumfanget som en samling af kubuser eller terninger. En kubus udspændes af otte voxels som er placeret i hvert hjørne af kubus (bemærk at voxels i denne sammenhæng betragtes som punktformige). Betegnelsen *tegl* vil blive anvendt om disse kubuser da de i en vis forstand overdækker hele rumfanget lissom tegl dækker et tag.

En konturflade vil skære en del af disse tegl og disse siges at være *kritiske*. Et kritisk tegl er udspændt både af voxels med en værdi over tærskelværdien og af voxels med en værdi under. Hvert kritisk tegl kan betragtes for sig, og der kan konstrueres et fladesegment eller en *facet* som beskriver konturfladen gennem teglet. Dette kan gøres således at facetterne i to kritiske nabo-tegl „passer sammen“ så facetterne kan sættes sammen til en sammenhængende flade. Algoritmer bygget over denne læst vil blive betegnet som *tegl-algoritmer*. Det der adskiller de forskellige tegl-algoritmer er først og fremmest hvordan konstruktionen af facetten i det kritiske tegl foregår.

4.5.10 „Marching cubes“ er en af de mest kendte og hyppigst implementerede tegl-algoritmer [Lorensen og Cline 1987, Cline et al. 1988]. Den er dog ikke den første af sin slags idet allerede Wyvill et al. [1986] beskriver en tegl-algoritme. Det der adskiller de to algoritmer er først og fremmest implementeringen, mens fladerne der konstrueres er (næsten) ens. Grunden til at „marching cubes“ er blevet den mest kendte og udbredte algoritme er måske at Lorensen og Cline [1987] og Cline et al. [1988] anvendte algoritmen på medicinske data fra røntgentomogrammer, mens Wyvill et al. [1986] modellerede syntetiske „bløde“ objekter. Den medicinske anvendelse har formodentlig haft større gennemslagskraft.

Uheldigvis er der i „marching cubes“ algoritmen en fejl som gør at de flader der konstrueres somme tider indeholder „huller“ eller „topologiske fejl“. Dette problem vil der blive redegjort for i afsnit 9.4, og i afsnit 9.5 kan man finde referencer til forskellige artikler der beskriver problemet.

Med tiden er der kommet flere algoritmer til. Baker [1989] har beskrevet en algoritme kaldet „weaving wall“ beregnet til at lave sand tids sporing af konturer i forbindelse med styring af en robot, men han har osse erkendt de mulige anvendelser inden for medicinen. Denne algoritme er fri for de fejl som plager „marching cubes“ algoritmen.

4.5.11 Flade-sporing. I stedet for at hver voxels opfattes som punktformig kan man tænke på dem som små terninger. Hele rumfanget udgøres af terninger som er „stabled“ side mod side. Mellem to nabo-voxels er der en grænseflade og disse fladesegmenter kan benyttes som byggesten for en konturflade. Værdierne i rumfanget antages at være binære (dette kan opnås ved at anvende en tærskelværdi på et ikke-binært rumfang), og en flade der skiller 1-voxels fra 0-voxels konstrueres ved en sporingsproces. Hvorledes dette gøres er beskrevet af [Artzy et al. 1981, Herman og Webster 1981, Gordon og Udupa 1989, Shu og Krueger 1991] hvor teknikken gradvis er blevet forbedret.

Ud over at det kræves at rumfanget er binært hvorved man ofte må bortkaste mere detaljeret information vil den færdige flade ofte osse have et „takket“ udseende da det geometriske primitiv der benyttes til konstruktion af fladen er fladesegmenter med en normal parallel med en af de tre principale akser.

4.5.12 Forenkling af flade. De konstruerede flader er ofte overordentligt „store“ forstået således at antallet af geometriske primitiver der medgår til at beskrive for eksempel et kranie er af størrelsesordenen 100.000–1.000.000. Da disse størrelser stadig er i overkanten af hvad der er praktisk muligt at håndtere på en grafisk arbejdsstation så er der behov for at reducere størrelsen eller kompleksiteten af de konstruerede flader. Et nærmere studie af en flade vil ofte vise at der er „jævne“ områder på fladen der med fordel kan repræsenteres med et færre antal primitiver. Denne proces kan man kalde for *forenkling* af fladen. Det væsentlige ved „forenklingen“ er at kun områder med lille detaljerighed forenkles. Man kan tænke sig at man ved justering af en (brugervalgt) parameter kan „forenkle“ fladen mere eller mindre. Ved at lade „forenklingsparameteren“ gennemløbe et diskret antal punkter i et interval får man *hierarkisk model* med flere niveauer af detaljerighed.

4.5.13 Prismetræet er en datastruktur beskrevet af Ponce og Faugeras [1987] som giver en hierarkisk opbygning af en polyhedral flade hvor mindre polygoner i fladebeskrivelsen smelter sammen til større polygoner efterhånden som man bevæger sig „op“ i træet og detaljerigheden mindskes. Uheldigvis kræver algoritmen at polyhedronen har genus nul (det vil sige er uden huller som det der er i en vanilliekrans), og dette er sjældent tilfældet for polyhedrale flader konstrueret fra medicinske data med tegl-algoritmer. Det er derfor nødvendigt at snitte disse flader indtil genus bliver nul, og dette er kompliceret og kræver ofte interaktion.

4.5.14 Fjernelse af skæringspunkter. Schroeder et al. [1992] beskriver en algoritme specielt designet til at forenkle flader konstrueret med „marching cubes“ algoritmen. Hvert skæringspunkt i den polyhedrale model opbygget af trekanter (det vil sige hvert hjørne i en af trekanterne) vurderes og fjernes hvis et nærmere specificeret kriterie er opfyldt. Dette kriterie vil for de fleste skæringspunkter være at skæringspunktets afstand til det plan der udspændes af de omgivende skæringspunkter er mindre end en given tærskelværdi. Når et skæringspunkt fjernes konstrueres en ny triangulering af det hul der opstår i fladen. Da algoritmen baserer sig på flader konstrueret med „marching cubes“ må den osse være i stand til at håndtere de „huller“ eller „topologiske fejl“ som algoritmen somme tider konstruerer.

4.5.15 Retriangulering. Turk [1992] beskriver en algoritme der på basis af en polyhedral flade konstruerer en helt ny forenklet model. Nye skæringspunkter til triangulering af modellen distribueres tilfældigt over den eksisterende flade og ved en gensidig frastødningsproces fordeles de jævnt. Ved at inddrage information om krumningen af den originale flade kan punkttætheden forøges i områder med stor krumning. Ved at vælge et forskelligt antal skæringspunkter i den nye flade kan man konstruerer flader med forskelligt niveau af detaljer, og i artiklen demonstreres det hvordan man kan interpolere en ny flade mellem to eksisterende flader konstrueret med et forskelligt niveau af detaljer.

4.5.16 Underinddeling af gitter. Især ved modellering af „bløde“ objekter hvor data ikke foreligger i form af empiriske målinger på et givet gitter, men i stedet er en til lejligheden konstrueret analytisk funktion der kan registreres på et vilkårligt gitter, har man algoritmer der forsøger at tilpasse gitteret således at områder med stor detaljerigdom registreres med en større frekvens. Det svarer til at der benyttes „mindre“ tegl i områder med stor detaljerigdom. Et af hovedproblemerne med den type algoritme er at undgå „sprækker“ i fladen langs grænsefladen mellem et „stort“ tegl og en række „mindre“ tegl.

Syntetiske flader som konstrueres fra analytiske felter kaldes ofte for *implicitte flader* og de kan være nyttige i for eksempel *computer aided geometric design* og *computer animation* da objekter som ellers kan være meget svære at modellere „traditionelt“ kan beskrives på en enkel måde.

Bloomenthal [1988] har arbejdet videre på resultaterne fra Wyvill et al. [1986] og Lorensen og Cline [1987], og de har opstillet algoritmer der konstruerer en konturflade på basis af et gitter der underinddeles indtil den ønskede detaljerigdom er opnået. Forskellen fra denne type algoritme og de tidligere beskrevne algoritmer til forenkling af flader er at de tidligere beskrevne algoritmer i en vis forstand har arbejdet „*nedefra og op*“ hvorved små geometriske primitiver samles til større. Den her beskrevne type algoritme arbejder „*oppefra og ned*“ indtil den nødvendige detaljerigdom er opnået.

Resultaterne fra Bloomenthal [1988] er senere blevet anvendt af Hall og Warren [1990] som har opstillet en algoritme som benytter tetraedre som tegl i stedet for kubuser. Dette giver en række forbedringer i forhold til den oprindelige algoritme. „Cutting cubes“ algoritmen beskrevet af Schmidt [1993] er osse baseret på resultaterne fra Bloomenthal [1988], men denne algoritme er i stand til at beskrive „singulære“ flader med selvgennemskæringer.

4.5.17 „Splitting-box“ algoritmen er ligeledes en algoritme baseret på Wyvill et al. [1986] og Lorensen og Cline [1987] der konstruerer forenklede flader *oppefra og ned*. Mens udgangspunktet for anvendelsen af algoritmerne beskrevet ovenfor har været modellering af syntetiske objekter så tager Müller og Stark [1993] udgangspunkt i anvendelse på empiriske data som, for eksempel, medicinske data. Ved den ovenfor beskrevne modellering af syntetiske objekter benyttes en række kriterier for om en facet er „jævn“ nok — er den ikke det, underinddeles det tegl som facetten tilhører indtil kriteriet om „jævnhed“ er opfyldt. Man kan i princippet blive ved med at underinddele da skalarfeltet findes på analytisk form og derfor kan beregnes i et vilkårligt punkt, men processen stopper naturligvis når en vis nedre grænse for teglenes størrelse er nået.

„Splitting-box“ algoritmen benytter i stedet „store“ tegl. Til at begynde med betragtes hele rumfanget som ét, stort tegl. Dette tegl kan deles i to mindre tegl ved en deling på tværs af teglets største dimension. Denne proces kan gentages indtil man er nået til det mindste niveau af tegl som svarer til de tegl som benyttes af for eksempel „marching cubes“ algoritmen. Algoritmen undersøger om et „stort“ tegl med en „stor“ facet kan anvendes i stedet for en underinddeling af det „store“ tegl i mindre tegl. Kriteriet er løst sagt at den „store“ facet ikke må afvige væsentligt fra de mindre facetter som tilsammen udgør den „store“ facet. Bemærk at det ikke er „jævnheden“ af facetten der afgør om et „stort“ tegl skal underinddeles.

4.5.18 Forenkling af billede. Et alternativ til at forenkle fladen er at forenkle det billede som fladen konstrueres fra. En simpel metode er anvendelse af *oktræer* som på forskellige måder kan integreres i mange af de ovenfor beskrevne algoritmer [Wilhelms og Van Gelder 1990a, Globus 1990, Coatrieux og Barillot 1990]. Mere interessant er måske algoritmen beskrevet af Bergtin et al. [1993]. Et rumfang forenkles ved at nærliggende, „ens“ billedpunkter kombineres. Resultatet er en Delaunay triangulering af rumfanget med langt færre billedpunkter end oprindeligt. En modificeret tegl-algoritme som anvender tetraedere som tegl kan så anvendes på dette rumfang. Det vil være interessant at undersøge om den konstruerede flade bliver enklere end en tilsvarende flade konstrueret med en sædvanlig tegl-algoritme.

II

Konstruktion af kontur

Kapitel 5

Billeder og flader

- 5.1 Digital topologi
- 5.2 Cellekomplekser
- 5.3 Digitale billeder
- 5.4 Foldning af billeder
- 5.5 Noter og litteratur

Før tegl-algoritmerne kan beskrives i detaljer er det nødvendigt at opstille et minimum af formalisme og terminologi. Målet er at opstille en matematisk abstraktion af såvel billeder som flader. Her benyttes *cellekomplekser* til dette formål hvorved der opnås at definitionerne kan generaliseres til vilkårlige dimensioner.

Først argumenteres for anvendeligheden af cellekomplekser som grundlag for digitale billeder og flader. Derefter bringes en formel definition af cellekomplekser og der opstilles en definition af *digitale billeder*. Der vil osse blive plads til at omtale hvordan man kan operere på disse billeder ved hjælp af *foldninger*. Endelig er der til slut et afsnit med henvisninger til mere grundig litteratur om emnet.

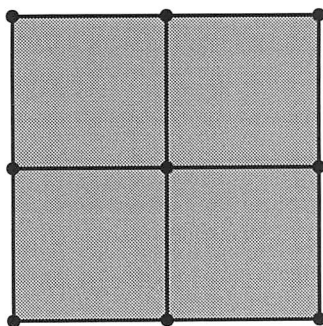
5.1 Digital topologi

5.1.1 Billedanalyse. I todimensional billedanalyse ønsker man at formalisere begreber som sammenhæng, rande, naboskab og så videre, således at de udsagn man kommer med kan underkastes matematikkens minutiøse logik. Dette er især aktuelt når billedanalysen skal generaliseres til højere dimensioner. Disse begreber hører under den matematiske disciplin topologi, og spørgsmålet er derfor hvordan man definerer en *digital topologi* som kan benyttes i billedanalyse.

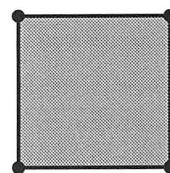
Det har vist sig at være vanskeligt at udstyre et billede opbygget af billedpunkter med en topologi som på en gang er enkel og elegant uden at den samtidig indeholder inkonsistenser.

I den forbindelse kommer man til at beskæftige sig med *omegn af et billedpunkt*, og *sammenhængsgrafen* i et billede. Man støder her på *sammenhængsparadokset* som har været genstand for megen diskussion. Diskussionen og definitionen af disse begreber er dog ikke specielt relevant i denne sammenhæng, men ved at udstyre et digitalt billede med en topologi fås en nyttig og veldefineret terminologi som vil blive anvendt flittigt.

5.1.2 Cellekomplekser. Som udgangspunkt for definitionen af en digital topologi benyttes her *cellekomplekser* som er velkendte fra den kombinatoriske og algebraiske topologi. Kovalovsky [1989] argumenterer for at enhver (matematisk konsistent) definition af en topologi på et digitalt billede nødvendigvis må være et cellekompleks. Dette udsagn kritiseres dog af Herman [1990], men den diskussion vil jeg ikke komme ind på her.



A) n -topologi.



B) 0-topologi.

Figur 5.1: Mulige cellestrukturer på et digitalt billede med fire billedpunkter. De sorte og grå toner tjener alene til at skelne mellem celler. Billedpunkternes værdier er ikke vist.

I stedet vil det blive demonstreret at cellekomplekser som fundament for en digital topologi kan give nyttige resultater — om der osse findes andre digitale topologier skal jeg lade være usagt.

5.1.3 Cellestruktur på digitalt billede. Nedenfor bringes en formel definition af et cellekompleks, men inden da vil det uformelt blive vist hvordan man på naturlig måde kan udstyre et digitalt billede med en cellestruktur. I den forbindelse skal man huske at i topologi er *morfierne*, det vil sige de afbildninger mellem topologiske rum der „bevarer egenskaber“, de *kontinuerte* afbildninger, og *isomorfierne* er de *bijektive* kontinuerte afbildninger, osse kaldet *homeomorfier*.

Ved en n -celle forstås (uformelt) en mængde homeomorf med den åbne n -dimensionale kugle. En 0-celle er et punkt; en 1-celle er et liniestykke; 2-celle er et fladeselement og så videre.

Billedet er opbygget af billedpunkter som antager forskellige værdier. Billedet tænkes placeret i planet $\mathbb{R}^2$, og det antages at hvert billedpunkt har sidelængden 1. Hvert billedpunkt er et enhedskvadrat, og definerer en 2-celle. Hver 2-celle har fire kanter, og hver kant definerer en 1-celle. Bemærk at et billedpunkts fire nærmeste naboer hver definerer én af de fire 1-celler således at 1-cellerne er „delt“ mellem to billedpunkter. En 2-celle har osse fire hjørner, og hvert hjørne definerer en 0-celle. Igen bliver den samme 0-celle defineret af flere billedpunkter. Med andre ord så opfattes det digitale billede som et billede opbygget af kvadrater (2-celler), af linier mellem to kvadrater (1-cellerne), og af punkter mellem fire kvadrater (0-celler).

På figur 5.1.A er der vist et eksempel på en cellestruktur på et lille digitalt billede bestående af fire billedpunkter. Cellestrukturen indeholder fire 2-celler, tolv 1-celler og ni 0-celler.

5.2 Cellekomplekser

Det er nu blevet antydnet hvordan et digitalt billede kan repræsenteres som en cellekompleks. Senere vil cellekomplekser osse blive anvendt til at repræsentere n -dimensionale flader. Derfor må jeg starte med at definere hvad et cellekompleks er, og her må jeg beklage at definitionerne kan virke alt for omstændelige og indviklede. Grunden til at

jeg ønsker at komme med disse generelle definitioner er at jeg bliver i stand til at generalisere begreber fra tre dimensioner til n dimensioner. Disse generelle begreber vil først blive anvendt i senere kapitler, men de bringes her fordi de så er samlet ét sted.

5.2.1 Forudsætninger. De topologiske rum der betragtes vil altid være det euklidiske rum $\mathbb{R}^n$ eller delmængder heraf. $\mathbb{R}^n$ vil være udstyret med den sædvanlige topologi induceret af det indre produkt som igen inducerer en norm og dermed en topologi, og delmængder af $\mathbb{R}^n$ vil være udstyret med den relative topologi.

5.2.2 Definition. Lad $n \in \mathbb{N}$ og definer n -kuglen i $\mathbb{R}^n$ som mængden

$$\mathbb{B}^n = \{x \in \mathbb{R}^n \mid \|x\| \leq 1\}.$$

Den topologiske rand i $\mathbb{R}^n$ af denne mængde er $(n-1)$ -sfæren

$$\mathbb{S}^{n-1} = \partial\mathbb{B}^n = \{x \in \mathbb{R}^n \mid \|x\| = 1\}$$

og forskellen mellem disse

$$\mathbb{B}^n = \mathbb{B}^n - \mathbb{S}^{n-1}$$

er den åbne n -kugle som er homeomorf med $\mathbb{R}^n$. Jeg sætter $\mathbb{B}^0 = \{0\}$.

For $m > n$ kan man indlejre disse mængder i $\mathbb{R}^m$ ved inklusionen $\iota: \mathbb{R}^n \rightarrow \mathbb{R}^m$ givet ved $\iota(x) = (x, 0, \dots, 0)$.

5.2.3 Definition. (Celle.) Lad X være topologisk rum. En delmængde $x \in X$ udstyret med den relative topologi er en n -celle hvis den er homeomorf med $\mathbb{B}^n$ og hvis homeomorfien kan udvides til en homeomorfi af $\mathbb{S}^{n-1}$ på randen af x (i X). Dimensionen af cellen skrives som $|x| = n$. Homeomorfien kaldes for den karakteristiske afbildning for x og betegnes med γ_x .

5.2.4 Bemærkning. Denne definition af en celle er en snæver definition som svarer til den sædvanlige definition af en *åben* og *regulær* celle. Det andet krav i definition sikrer at cellen bliver regulær. Et eksempel på en celle der ikke er regulær er $\mathbb{S}^n - (1, 0, \dots, 0) \in \mathbb{R}^{n+1}$ for $n \geq 1$. Cellen er homeomorf med $\mathbb{B}^n$, men randen er homeomorf med $\mathbb{B}^0$ og ikke $\mathbb{S}^{n-1}$.

5.2.5 Definition. Et cellekompleks på mængden X er en endelig mængde af delmængder $\mathcal{X}$ kaldet cellerne som opfylder:

1. $X = \bigcup_{x \in \mathcal{X}} x$ med $x \neq x' \Rightarrow x \cap x' = \emptyset$. Det vil sige at X er en disjunkt forening af cellerne.
2. Hver celle $x \in \mathcal{X}$ er en n -celle i X . Foreningen $\circ^m(X) = \bigcup_{|x| \leq m} x$ kaldes for m -skelettet og består af alle celler hvis dimension ikke er højere end m . Mængden af m -celler $\circ^m(X) - \circ^{m-1}(X)$ betegnes osse med $\overline{\circ^m(X)}$. Den højeste værdi af m for hvilken der gælder at $\overline{\circ^m(X)} \neq \emptyset$ er cellekompleksets dimension.
3. For hver celle $x \in X$ med karakteristiske afbildning γ_x skal restriktion $\gamma_x|_{\mathbb{S}^{n-1}}$ være en afbildning på $\circ^{n-1}(X)$.

5.2.6 Bemærkning. Man kan give flere definitioner på et cellekompleks. Den der er bragt er en definition af et *endeligt*, *regulært* CW-kompleks. Definition kan udvides til et *uendeligt* CW-kompleks ved at tilføje to ekstra punkter til definitionen, men det er ikke nødvendigt her.

Ønskes en matematisk abstraktion af flader anvendt inden for *computer graphics* opbygget af for eksempel trekanter så er todimensionale cellekomplekser velegnede. Denne abstraktion er nyttig da den lader sig generalisere til højere dimensioner som det vil blive demonstreret i kapitel 10.

5.2.7 Definition. (Siderelation.) Givet et cellekompleks $(X, \mathcal{X})$ med karakteristiske afbildninger γ_x for $x \in \mathcal{X}$ defineres for $x, y \in \mathcal{X}$ relationen $\preceq$

$$x \preceq y \quad \text{netop hvis} \quad \gamma_y(\mathbb{B}^{|y|}) \subseteq x.$$

5.3 Digitale billeder

Det er nu muligt at give en formel definition af et digitalt billede som vil blive benyttet fremover. Den underliggende struktur er et cellekompleks som tilføjes værdier og koordinater.

5.3.1 Den topologiske struktur der skal danne basis for et digitalt billede er den type cellekompleks som er illustreret på figur 5.1. Denne struktur kan man kalde for et *gitter*.

5.3.2 Definition. (Endeligt kubisk gitter) i $\mathbb{R}^n$ med dimensioner $\{m_1, \dots, m_n\}$ er et cellekompleks på $X = [0; m_1 - 1] \times \dots \times [0; m_n - 1]$ med k -cellerne

$$x_{j_1 \dots j_n}^{i_1 \dots i_k} = \left\{ \sum_{\nu=1}^n j_\nu e_\nu + \sum_{\kappa=1}^k t_\kappa e_{i_\kappa} \mid 0 < t_\kappa < 1 \right\} \quad (5.1)$$

hvor $e_1, \dots, e_n$ er basisvektorerne der beskriver det cartesiske koordinatsystem på $\mathbb{R}^n$. Cellerne i komplekset er mængden

$$\mathcal{X} = \{ x_{j_1 \dots j_n}^{i_1 \dots i_k} \mid j_\nu \in \{0, 1, \dots, m_\nu - 1\}, i_\kappa \in \{1, 2, \dots, n\}, i_1 < i_2 < \dots < i_k \}.$$

Den karakteristiske afbildning $\gamma_{j_1 \dots j_n}^{i_1 \dots i_k}$ for cellen $x_{j_1 \dots j_n}^{i_1 \dots i_k}$ er defineret på $[0; 1]^k$ som er homeomorf med $\mathbb{B}^k$ og givet ved

$$\gamma_{j_1 \dots j_n}^{i_1 \dots i_k}(t_1, \dots, t_k) = \sum_{\nu=1}^n j_\nu e_\nu + \sum_{\kappa=1}^k t_\kappa e_{i_\kappa}.$$

Det er klart at et endeligt kubisk gitter er et cellekompleks. Dimensionen af gitteret er dimensionen af cellekomplekset som er n , og betegnelsen n -gitter benyttes for at understrege dimensionaliteten af gitteret.

Uformelt kan man beskrive den underliggende mængde for et gitter som en „klods“ i $\mathbb{R}^n$. Det er somme tider nødvendigt at henvise til randen af denne „klods“ og de celler som ligger i randen. Ved randen af den underliggende mængde X forstås den topologiske rand ∂X i $\mathbb{R}^n$. Det er klart at en celle som skærer randen osse er helt indeholdt i randen, og der gælder tilmed at cellerne i randen udgør et cellekompleks.

5.3.3 Værdifunktion. Et gitter giver en topologisk struktur på billedets definitions- mængde, men billedets „indhold“ fås ikke fra denne struktur alene. Det der gør billedet til et „billede“ er at de enkelte billedpunkter (eller celler) har forskellige værdier. Dette formaliseres ved at definere en *værdifunktion* V på cellestrukturen.

Det ville være naturligt at kræve at værdifunktionen V var defineret på alle $x \in \mathcal{X}$ og dermed på hele X , men i praksis vil V kun være defineret på en delmængde af $\mathcal{X}$.

5.3.4 Samspil mellem værdifunktion og topologi. I en traditionel opfattelse af et digitalt billede har man en række billedpunkter som antager forskellige værdier. Spørgsmålet er: hvilke celler i gitteret skal svare til de oprindelige billedpunkter? Og der er to naturlige svar: 2-cellerne som illustreret på figur 5.1.A og 0-cellerne som vist på figur 5.1.B. I det n -dimensionale tilfælde vil det første svar i stedet være n -cellerne. Man kan derfor opstille to forskellige topologiske strukturerer til det samme billede.

5.3.5 Definition. Der er to måder at knytte sammenhæng mellem et kubisk n -gitter $(X, \mathcal{X})$ og den fysiske virkelighed gitteret repræsenterer.

1. 0-topologi er det tilfælde hvor V er defineret på 0-cellerne i $(X, \mathcal{X})$.

2. n -topologi er det tilfælde hvor V er defineret på n -cellerne i $(X, \mathcal{X})$.

I det tilfælde hvor 0-topologien anvendes vil de „informationsbærende“ celler (det vil sige 0-cellerne) have formen $x_{j_1 \dots j_n}$ jævnfør udtrykket (5.1). Ved anvendelse af n -topologien vil disse celler i stedet have formen $x_{j_1 \dots j_n}^{1 \dots n}$. Man kan se at det er værdien af $j_1 \dots j_n$ der entydigt fastlægger cellen i begge tilfælde. Når tvivl ikke er mulig vil betegnelsen $x_{j_1 \dots j_n}$ fremover blive benyttet om de celler som er definitionsområde for værdifunktionen hvorved en række udsagn forenkles da det ikke er nødvendigt at skelne mellem de to topologier.

5.3.6 Værdimængde af V . Værdifunktionen V kan have forskellige værdimængder. Hvis billedet er binært er værdimængden $\{0, 1\}$, hvis billedet er et 8-bit gråtone billede er værdimængden $\{0, 1, \dots, 255\}$ og hvis billedet er et 24-bit RGB-billede så er værdimængden tripler af 8-bit værdier. Der er talrige andre muligheder, men oftest antages det at værdimængden for V er $\mathbb{R}$. Derved bliver man i stand til, for eksempel, at midle værdier og så videre. De diskrete eksempler beskrevet ovenfor kan så betragtes som tilnærmelser til den kontinuerte virkelighed.

5.3.7 Koordinatfunktion. Det er ikke altid muligt at „presse“ et kubisk 2-gitter ned over „virkeligheden“. De enkelte billedpunkter kan for eksempel være $2 \text{ mm} \times 3 \text{ mm}$ store og dermed forskellige fra de enhedskvadrater gitteret er opbygget af. En *koordinatfunktion* der knytter en sammenhæng mellem n -gitteret $(X, \mathcal{X})$ og $\mathbb{R}^n$ introduceres.

5.3.8 Definition. En gyldig koordinatfunktion på n -gitteret $(X, \mathcal{X})$ er en funktion defineret på de samme celler som værdifunktionen V som har formen

$$C = (C_1, \dots, C_n): {}^{\circ m}(X) \rightarrow \mathbb{R}^n$$

hvor m er 0 eller n og hvor m -cellerne kort skrives som $x_{j_1 \dots j_n}$ jævnfør bemærkningen ovenfor om „informationsbærende“ celler. Hver C_k skal opfylde at $C_k(x_{j_1 \dots j_k \dots j_n})$ er konstant for fast j_k og monotont voksende som funktion af j_k .

Ovenstående definition er en meget generel definition af en koordinatfunktion. Ofte vil en koordinatfunktion høre til en mindre klasse givet ved følgende...

5.3.9 Definition. En koordinatfunktion $C = (C_1, \dots, C_n)$ er regulær i den k 'te koordinat hvis der findes en konstant $c_k > 0$ som opfylder

$$C_k(x_{j_1 \dots j_k+1 \dots j_n}) - C_k(x_{j_1 \dots j_k \dots j_n}) = c_k.$$

Et todimensionalt billede som har billedpunkter med dimensionen $2 \text{ mm} \times 3 \text{ mm}$ vil have en koordinatfunktion der er regulær i begge koordinater med $c_1 = 2$ og $c_2 = 3$ hvor koordinater måles i enheder af mm.

5.3.10 Identisk koordinatfunktion. I nogle tilfælde er koordinatfunktionen uden betydning og da kaldes koordinatfunktionen for *identiteten*. Dette skal forstås således at $C(x_{j_1 \dots j_n}) = x_{j_1 \dots j_n}$ hvor $x_{j_1 \dots j_n}$ i 0-topologien opfattes som en vektor i $X \subset \mathbb{R}^n$ og i n -topologien opfattes som centret af n -cellen $x_{j_1 \dots j_n}$. Det svarer til at alle billedpunkter har dimensionen $1 \times 1 \times \dots \times 1$.

5.3.11 Digitalt billede. Det er nu muligt at opstille en fyldestgørende definition af et digitalt billede.

5.3.12 Definition. (Digitalt billede.) Et digitalt n -billede er en tripel $((X, \mathcal{X}), V, C)$. Cellekomplekset $(X, \mathcal{X})$ er det kubiske n -gitter hvorpå billedet er defineret, V er værdifunktionen og C er koordinatfunktionen.

Billedets dimension er dimensionen af det underliggende gitter. Gitterets underliggende mængde X kaldes for billedets *definitions-mængde*.

5.3.13 Komplementaritet. De to forskellige slags topologier der er beskrevet her er i en vis forstand komplementære, og de svarer til hvert sit *paradigme*. I 0-topologi er de celler der „bærer“ værdifunktionen punkter og man kan se en parallel til den abstraktion der ofte anvendes i klassisk mekanik hvor man studerer punktformige legemers bevægelse. I modsætning til dette har man n -topologi som „spreader“ værdifunktionen ud over de største celler. Dette svarer meget godt til en måleproces hvor man måler middelværdien af en kontinuert funktion over et vist areal fordi man ikke kan gøre det bedre. Dette optræder som del-rumfangs effekten i forbindelse med røntgentomografi som beskrevet i afsnit 3.2.8.

Ud fra denne diskussion skulle man måske tro at n -topologi er den mest „korrekte“ måde at betragte de fleste problemer fra virkelighedens verden, men de algoritmer der beskrives i de næste kapitler anvender 0-topologien uden at jeg dog dermed mener at resultaterne er uanvendelige.

5.4 Foldning af billeder

En ofte anvendt operation inden for digital billedbehandling er en *foldning* af billedet (kaldet *convolution* på engelsk). Her vil den matematiske baggrund for den type operation ikke blive beskrevet, men forenklet kan en foldning af et digitalt billede beskrives som en operation, der for hvert billedpunkt i billedet beregner en ny værdi i dette billedpunkt ved at inddrage værdierne af billedpunkterne i en lille omegn omkring det oprindelige billedpunkt. Værdierne af de omgivende billedpunkter kombineres ved en linearkombination hvor koefficienterne er givet ved foldningens *kerne*. Med den her bragte definition af et digitalt billede fås følgende forskrift for hvorledes en foldning udføres.

5.4.1 Definition. (Foldning.) Lad $P = ((X, \mathcal{X}), V, C)$ være et n -billede og lad $[a_{l_1 \dots l_n}] = \{a_{l_1 \dots l_n} \in \mathbb{R} \mid l_i = -d_i, \dots, d_i, i = 1, \dots, n\}$ være en n -dimensional kerne med dimension $(2d_1 + 1) \times \dots \times (2d_n + 1)$. Foldningen P' af P med kernen $[a_{l_1 \dots l_n}]$ er på ny et billede som fås fra P ved at erstatte værdifunktionen V med værdifunktionen V' givet ved

$$V'(x_{j_1 \dots j_n}) = \sum_{l_1 = -d_1}^{d_1} \dots \sum_{l_n = -d_n}^{d_n} a_{l_1 \dots l_n} V(x_{(j_1 + l_1) \dots (j_n + l_n)}).$$

5.4.2 Bemærkning. Der kan optræde celler med ugyldige indeks i summen hvis $j_i + l_i < 0$ eller $j_i + l_i \geq m_i$. Løst sagt kan man sige at problemet er hvorledes man bedst udvider værdifunktionen til celler uden for billedet. Der er traditionelt to måder at håndtere dette problem på: Den ene lader værdifunktionen være „cyklisk“ sådan at forstå at et ugyldigt indeks $j_i + l_i$ erstattes med indekset $j_i + l_i \bmod m_i$. I stedet kan man „udvide“ værdifunktionen ved at erstatte et ugyldigt indeks $j_i + l_i$ som er negativt med 0 og ugyldigt indeks som er for stort med $m_i - 1$.

Den form for billedbehandling der bliver anvendt her giver ikke nogen særlig anledning til at anvende den „cykliske“ metode så her vil „udvidelsen“ altid blive anvendt, selv om om denne metode i praksis ofte er den langsommeste af de to.

5.4.3 Repræsentation af kerne. I to dimensioner kan kernen bekvemt opskrives som en matrice. Til at repræsentere koefficienterne $a_{l_1 l_2}$ benyttes systemet givet ved

$$[a_{l_1 l_2}] = \begin{bmatrix} a_{(l_1-d_1)(l_2+d_2)} & a_{(l_1-d_1+1)(l_2+d_2)} & \cdots & a_{(l_1+d_1)(l_2+d_2)} \\ a_{(l_1-d_1)(l_2+d_2-1)} & a_{(l_1-d_1+1)(l_2+d_2-1)} & \cdots & a_{(l_1+d_1)(l_2+d_2-1)} \\ \vdots & \vdots & \ddots & \vdots \\ a_{(l_1-d_1)(l_2-d_2)} & a_{(l_1-d_1+1)(l_2-d_2)} & \cdots & a_{(l_1+d_1)(l_2-d_2)} \end{bmatrix}$$

Denne opskrivning svarer til hvordan man normalt ville organisere punkterne i et todimensionalt gitter i et cartesisk koordinatsystem med sædvanlig orientering.

Tredimensionale kerner kan skrives som et sæt af matricer som beskrevet ovenfor efter følgende system

$$[a_{l_1 l_2 l_3}] = [[a_{l_1 l_2}]_{d_3}, [a_{l_1 l_2}]_{d_3-1}, \dots, [a_{l_1 l_2}]_{-d_3}].$$

hvor $[a_{l_1 l_2}]_{l_3}$ skal forstås som den todimensionale matrice beskrevet ovenfor med elementet $a_{k_1 k_2}$ erstattet med $a_{k_1 k_2 l_3}$.

Løst sagt kan man med denne repræsentation foretage en mental visualisering af kernen i et højrevendt cartesisk koordinatsystem med x -aksen pegende vandret til højre og y -aksen pegende lodret opad.

5.5 Noter og litteratur

Digital topologi er beskrevet af blandt andet Bogomolny [1988], Kovalevsky [1989], Herman [1990] og Malandain et al. [1991]. Fokus har ofte været på at beskrive sammenhæng i billeder for på den måde at løse *sammenhængsparadokset*.

Cellekomplekser optræder i kombinatorisk og algebraisk topologi. Den sidste disciplin er på mange måder en interessant matematisk disciplin fordi tilsyneladende helt forskellige begreber forenes i en samlet teori med nogle smukke resultater. Ingredienserne er blandt andet geometri, topologi, algebra og kategorier.

Hvor målet i algebraisk topologi er at fjerne sig fra de geometriske objekter via abstraktion er ønsket her at konstruere håndgribelige geometriske objekter. Derfor anvendes først og fremmest begrebsverdenen fra algebraisk topologi og i mindre grad de matematiske resultater. Når man studerer algebraisk topologi bliver man derfor hurtigt ført væk fra de emner dette speciale omhandler. For den interesserede kan jeg dog anbefale [Hilton og Wylie 1960, Massey 1967, Dold 1972, Agoston 1976, Rourke og Sanderson 1982, Fritsch og Piccinini 1990].

I de følgende kapitler vil der nu og da blive anvendt begreber fra gruppeteori. Disse begreber er beskrevet i næsten enhver grundbog om dette emne. For eksempel kan jeg anbefale [Yale 1968, Lyndon 1985].

Kapitel 6

En todimensional tegl-algoritme

- 6.1 Indledning
- 6.2 Facetter
- 6.3 Flertydighed
- 6.4 Egenskaber ved tegl
- 6.5 Tegl og gruppeteori
- 6.6 Algoritmen

I dette kapitel beskrives en todimensional tegl-algoritme. De i litteraturen beskrevne algoritmer, som for eksempel „marching cubes“ algoritmen, er „tredimensionale“, men et af resultaterne i dette speciale er at tegl-algoritmer kan generaliseres til et vilkårligt antal dimensioner. Det er derfor på sin plads at starte i to dimensioner, men man skal ikke blive overrasket hvis man synes at den todimensional algoritme ikke er særlig nyttig — grunden til at den bringes her er at det derved bliver muligt at introducere en række begreber og metoder på en forhåbentlig forståelig og intuitiv måde. Disse kan senere generaliseres til højere dimensioner hvor anvendelsen af intuition kan være vanskelig.

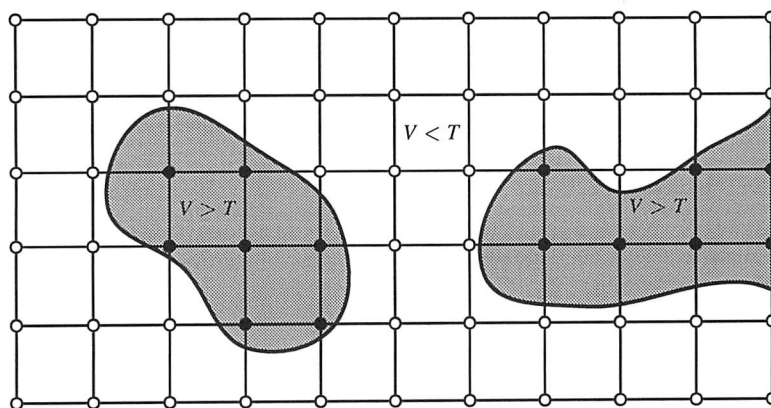
6.1 Indledning

6.1.1 Billedet. Lad der være givet et 2-billede $P = ((X, \mathcal{X}), V, C)$ som defineret i 5.3.12, og antag at den benyttede topologi er 0-topologien. For at forenkle sprogbrugen vil betegnelsen *hjørner* blive benyttet om 0-cellerne i billedet. Tilsvarende vil betegnelsen *kanter* blive anvendt om 1-cellerne i billedet.

6.1.2 Ækvipotentialkurve. Man kan forestille sig værdifunktionen V som en måling på hjørnerne i $\mathcal{X}$ af et kontinuert felt $\tilde{V}: X \rightarrow \mathbb{R}$ defineret på hele X . For en given tærskelværdi T defineres en ækvipotentialkurve

$$\tilde{E} = \{x \in X \mid \tilde{V}(x) = T\}$$

som afhængig af T kan have nul eller flere komponenter. Strengt taget skal $\tilde{V}$ opfylde nogle særlige krav for at $\tilde{E}$ bliver en kurve i matematisk forstand, men dette er uvæsentligt for beskrivelsen af algoritmen. Formålet i dette kapitel er at opstille en algoritme til at bestemme en kurve E opbygget af rette liniestykker som tilnærmer $\tilde{E}$. Denne kurve kaldes for en *kontur*.



Figur 6.1: Et eksempel på en ækvipotentialekurve givet ved tærskelværdien T . De mørke områder repræsenterer de dele af feltet der er over tærskelværdien. I højre side er kurven lukket ved at en del af randen af billedet er tilføjet.

6.1.3 Definition. (Høje og lave hjørner) Alle hjørner (0-celler) $x \in \mathcal{X}$ kan deles i to kategorier:

1. Hjørnet x siges at være højt hvis $V(x) \geq T$.
2. Hjørnet x siges at være lavt hvis $V(x) < T$.

6.1.4 Bemærkning. Definitionen er asymmetrisk idet hjørner x med $V(x) = T$ er høje, og man kunne i stedet lade disse hjørner være lave. Den estimerede kontur E vil blive den samme uanset valg af definition. Se i øvrigt bemærkning 6.2.7.

6.1.5 Opdeling i høje og lave områder. Såfremt kurven (kurverne) $\tilde{E}$ er lukkede vil den (de) dele X i to (flere) komponenter, og hver komponent vil enten indeholde kun høje eller kun lave hjørner som illustreret på figur 6.1.

Hvis en kurve ikke er lukket skyldes det at den starter og ender på randen af X , men derved kan man lukke kurven ved at tilføje randen mellem kurvens to endepunkter som en del af kurven, og argumentet ovenfor holder stadig. Konklusionen er at en konturkurve E som adskiller de høje hjørner fra de lave hjørner er en god tilnærmelse til ækvipotentialekurven $\tilde{E}$.

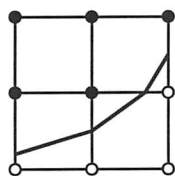
6.1.6 Tegl er det begreb der har givet algoritmen sit navn. På figur 6.2.A ses det at man kan betragte hvert kvadrat for sig med henblik på at bestemme konturen der løber igennem. Hele billedet bliver overdækket af fire kvadrater eller *tegl*. I tre dimensioner er et tegl en kubus, deraf navnet „marching cubes“.

Mere præcist så er et 2-tegl en 2-celle og dens rand på fire 1- og fire 0-celler. Dette kan generaliseres til:

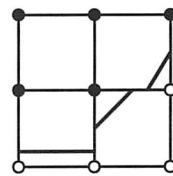
6.1.7 Definition. (Tegl.) For $n \geq 1$ og n -cellen $x \in \mathcal{X}$ defineres n -teglet som det n -dimensionale cellekompleks

$$\tau = \{y \in \mathcal{X} \mid y \preceq x\}.$$

To n -tegl som har en fælles $(n-1)$ -celle kaldes for nabo-tegl.



A) Konsistent kontur der adskiller høje og lave hjørner.



B) Inkonsistent kontur.

Figur 6.2: Et billede hvor høje hjørner er angivet med mørke skiver.

Bemærk at n -tegl (nabo-tegl såvel som andre tegl) kan overlappe hinanden og at n -tegl er opbygget af m -tegl med $m < n$. For enkelhedens skyld vil betegnelsen „tegl“ dække over 2-tegl i resten af dette kapitel når intet andet er angivet, og som før nævnt vil betegnelserne „kanter“ og „hjørner“ dække over 1- og 0-tegl.

Den her givne definition af tegl er ikke specifik for den todimensionale algoritme, men kan anvendes i beskrivelsen af en n -dimensional algoritme. Dette er osse tilfældet for den næste definition:

6.1.8 Definition. (Kritisk tegl.) Et tegl som indeholder både høje og lave hjørner (0-celler) siges at være kritisk.

Et kritisk 2-tegl vil indeholde kritiske 1-tegl og disse kaldes for *kritiske kanter*.

6.2 Facetter

6.2.1 Lineær approksimation til ækvipotentialekurve. Det er klart at konturen E må gå igennem netop de kritiske tegl, og den må skære alle de kritiske kanter. I princippet kan $\tilde{E}$ „sno“ sig frem og tilbage over den samme kritiske kant, men da man kun har målinger af $\tilde{V}$ i „enderne“ af den kritiske kant er det ikke muligt at afgøre om dette er tilfældet. Konturen E konstrueres derfor således at den skærer hver kritisk kant netop én gang, og disse *skæringspunkter* forbindes med rette linier. Disse byggesten, skæringspunkter og liniestykker mellem skæringspunkter, kaldes for *facetter*.

6.2.2 Definition. (0-Facet.) Skæringen mellem konturen E og de kritiske kanter kaldes for E 's 0-facetter eller skæringspunkter. Hvert kritisk 1-tegl τ indeholder netop én 0-facet

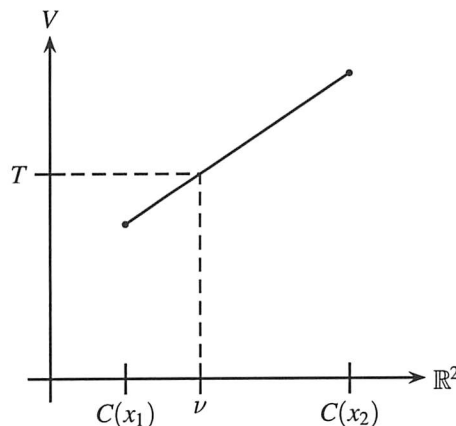
$$f^0(\tau) = \{\nu\}$$

hvor ν er et punkt i τ . En 0-facet er et cellekompleks med dimension nul. Hvis τ ikke er kritisk sættes $f^0(\tau) = \emptyset$.

6.2.3 Estimation af skæringspunkt. Hvis man antager at feltet $\tilde{V}$ varierer lineært fra det lave til det høje hjørne i en kritisk kant kan man estimere skæringspunktets placering ν ved

$$\nu = \frac{T - V(x_1)}{V(x_2) - V(x_1)} (C(x_2) - C(x_1)) + C(x_1) \quad (6.1)$$

hvor x_1 og x_2 er de to hjørner (se figur 6.3).



Figur 6.3: Lineær interpolation af skæringspunkt. Bemærk at abscissen i realiteten er planet $\mathbb{R}^2$.

6.2.4 Bemærkning. Man kan osse foreslå andre estimater, eventuelt baseret på højere ordens approksimationer, men man må kræve at de er af en form

$$\nu = \alpha(x_1, x_2, T) \cdot (C(x_2) - C(x_1)) + C(x_1)$$

hvor α opfylder

1. $0 \leq \alpha(x_1, x_2, T) \leq 1$ når $\min(V(x_1), V(x_2)) \leq T \leq \max(V(x_1), V(x_2))$,
2. $\alpha(x_1, x_2, V(x_1)) = 0$ og
3. $\alpha(x_1, x_2, V(x_2)) = 1$.

6.2.5 Definition. (1-Facet.) To 0-facetter ν_1 og ν_2 i et kritisk 2-tegl τ forbindes med et liniestykke så en 1-facet dannes. Facetten er et éndimensionalt cellekompleks bestående af én 1-celle og to 0-celler, facetten er orienteret og repræsenteres ved parret (ν_1, ν_2) . Mængden af 1-facetter i τ betegnes med

$$f^1(\tau) = \{ (\nu_1^i, \nu_2^i) \mid i = 1, \dots, m \}$$

hvor m er antallet af 1-facetter og ν_j^i er 0-facetterne i hver af 1-facetterne. Når τ ikke er kritisk sættes $f^1(\tau) = \emptyset$.

I dette kapitel vil betegnelsen facetter fremover dække over 1-facetter når intet andet er angivet, og 0-facetter kaldes da for skæringspunkter.

6.2.6 Orientering af facetter. Det forudsættes at en facet (ν_1, ν_2) er orienteret således at når man „går“ fra ν_1 til ν_2 langs facetten så vil det høje område være på venstre hånd. Det betyder at hvis der dannes en lukket kontur i et billede så vil det høje område som konturen omslutter ligge på venstre hånd når man bevæger sig i positiv retning langs facetterne i konturen.

6.2.7 Bemærkning. En facet kan være udartet således at $\nu_1 = \nu_2$. Dette forekommer når et hjørne x har værdien $V(x) = T$ og i dette tilfælde gælder der $\nu_1 = \nu_2 = x$. Det er egentlig definition af høje

og lave hjørner der er skyld i at facetter kan udarte sig — var høje hjørner defineret som hjørner der opfyldte $V(x) > T$ i stedet for $V(x) \geq T$ ville fænomenet ikke optræde, men hjørner som hverken er høje eller lave kunne så forekomme.

I en konkret implementering af en tegl-algoritme er det forholdsvis enkelt at lokalisere udartede facetter og fjerne dem fra uddata som antydnet i bemærkning 7.8.4.

6.2.8 Algoritmen kan nu skitseres i grove træk: Billedet afsøges for kritiske tegl. Hvert kritisk tegl behandles for sig med henblik på at fastlægge facetten (eller facetterne) som løber gennem teglet. Facetterne sammensættes til den endelige kontur E , men dette kan kun lade sig gøre såfremt to facetter der udgår fra samme kritiske kant udgår fra samme skæringspunkt. Sagt på en anden måde så kræves der...

6.2.9 Konsistens. To kritiske nabo-tegl som deler en kritisk kant skal fastlægge det samme skæringspunkt i kanten. Denne egenskab kaldes for *konsistent fastlæggelse af skæringspunkter*.

Figur 6.2.B viser et eksempel på nogle facetter der *ikke* er konsistente. En vigtig konsekvens af konsistens er at man er i stand til at behandle hvert tegl for sig uafhængigt af de øvrige, og bagefter sammensætte resultaterne fra hvert kritisk tegl til et *konsistent* hele.

Definitionen af konsistens kan generaliseres til n dimensioner. Foreløbig er her kun redegjort for definition for $n = 2$, men her kommer alligevel den generelle definition.

6.2.10 Definition. (Konsistens.) To kritiske n -nabo-tegl som deler et kritisk $(n - 1)$ -tegl τ skal fastlægge samme $(n - 2)$ -facet i τ . Dette kaldes for konsistens mellem nabo-tegl.

I to dimensioner kan dette omskrives til at to nabo-tegl τ_1 og τ_2 som deler en kritisk kant ϵ skal opfylde

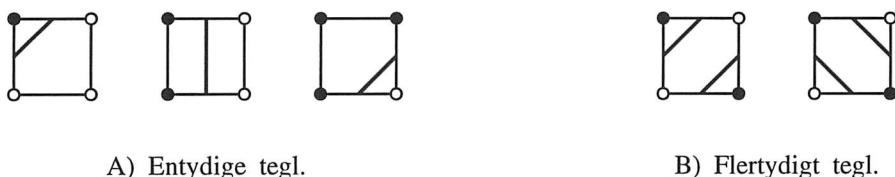
$$f_1^0(\epsilon) = f_2^0(\epsilon) \quad (6.2)$$

hvor f_i^0 er f^0 anvendt i τ_i 's kontekst. Dette er opfyldt når udtrykket (6.1) anvendes da $f^0(\epsilon)$ er den samme uanset om ϵ regnes som hørende til τ_1 eller τ_2 .

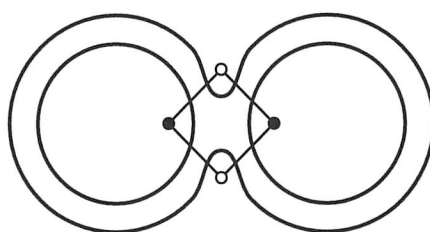
6.3 Flertydighed

Definitionen af facetter angiver ikke altid hvilke skæringspunkter der skal forbindes — der opstår et problem hvis der i et kritisk tegl er mere end to skæringspunkter — hvilke skæringspunkter skal da forbindes med rette linier? For at finde svaret skal man se på mulige...

6.3.1 Konfigurationer af kritiske tegl. Et kritisk tegl kan have ét, to eller tre høje hjørner som vist på figur 6.4. Normalt vil dette betyde at der er to kritiske kanter og dermed to skæringspunkter i teglet, men i det tilfælde hvor der er to høje hjørner placeret diagonalt overfor hinanden vil der være fire skæringspunkter som vist på figur 6.4.B. Når der er to skæringspunkter forbindes de med en ret linie som er facetten, men dette kan ikke umiddelbart gøres når der er fire skæringspunkter. I dette tilfælde er der to muligheder som vist på figuren. Dette fører til *flertydighed* i den forstand at man mellem flere alternativer må vælge netop ét.



Figur 6.4: Konfigurationer af kritiske tegl.



Figur 6.5: Det i teksten beskrevne højdekort der tjener til at eksemplificere flertydighed.

6.3.2 To alternativer. Man kan forbinde de fire skæringspunkt på to måder og det vil i begge tilfælde føre til at teglet deles i tre områder. På den ene måde vil de to høje hjørner være i hvert sit område, og på den anden måde vil de to lave hjørner være i hvert sit område. Man kan benytte følgende betegnelser om de to alternativer:

1. Når de høje hjørner skilles ad af et sammenhængende område hvori de to lave hjørner befinder sig benyttes det *lavt sammenhængende* alternativ.
2. I det modsatte tilfælde benyttes det *højt sammenhængende* alternativ.

Man må vælge et af de to alternativer, men hvilken betydning har dette valg?

6.3.3 Eksempel. For at illustrere betydningen af valget kan man som felt tage et højdekort over et landskab med to lige høje bakker forbundet af en ås med et pas (saddelpunkt). På kortet kan man indlægge højdekurver for forskellige højder eller *tærskelværdier*, og det er opførslen af disse der skal undersøges.

Hvis man vælger en højde større end højden af de to bakketoppe vil der ingen højdekurver være på kortet. Hvis højden er en højde mellem højden af passet og højden af bakketoppene så er der to kurver der markerer de to bakker. Er højden under pashøjden vil de to adskilte kurver smelte sammen til én (ottetalsformet) kurve. De to forskellige typer højdekurver er vist på figur 6.5.

Nu digitaliseres højdekortet på en sådan måde at der er et tegl med centrum i passet og med en diagonal der løber langs åsen som vist på figuren. Når højden for højdekurven er lige omkring pashøjden vil dette tegl være kritisk idet de to hjørner der ligger i åsen vil være over højden, mens de to hjørner der ligger uden for åsen

vil være under højden. Der er fire skæringspunkter i teglet og hvis det lavt sammenhængende alternativ vælges vil den konstruerede kontur svare til det tilfælde hvor der er to højdekurver. Hvis derimod det højt sammenhængende alternativ vælges fås én (ottetalsformet) kontur.

6.3.4 Globale egenskaber. Valget af alternativ bestemmer nogle globale egenskaber ved konturen, nemlig hvor mange sammenhængende komponenter den har. Alligevel er algoritmen lokal i sin natur (hvilket osse gør den effektiv) og derfor er det mest hensigtsmæssigt at foretage et valg ud fra lokale informationer.

6.3.5 Metoder til afgørelse af flertydighed. Her er nogle forslag til hvordan flertydighed kan afgøres:

1. Man kan benytte enten det lavt eller det højt sammenhængende alternativ i alle flertydige tilfælde. Denne metode kan man kalde for den *konsekvente metode*. Afhængigt af hvilket alternativ der anvendes kaldes metoden for den *konsekvent lavt sammenhængende metode* eller for den *konsekvent højt sammenhængende metode*.
2. Man kan interpolere en feltværdi i centret af teglet. Dette svarer til at forsøge at beregne pashøjden i eksempel 6.3.3. Hvis centerværdien er under tærskelværdien benyttes det lavt sammenhængende alternativ og ellers benyttes det højt sammenhængende. Den enkleste form for interpolation fås ved at beregne et gennemsnit af de fire feltværdier i teglet, men man kan osse benytte højere ordens metoder. Denne metode kan man kalde for *estimationsmetoden*.
3. Endelig kan valget af alternativ være „tilfældigt“, det vil sige usystematisk. Denne mangel på system kan i det tredimensionale tilfælde give problemer som demonstreret i afsnit 7.2.7.

6.3.6 Den konsekvente metode er enkel, men kan samtidig synes utilfredsstillende. Endvidere virker valget af lavt eller højt sammenhængende alternativ noget tilfældigt (og er det osse) da der er symmetri mellem de to alternativer. Det der gør denne løsning tiltrækkende er at der i højere dimensioner er tilsvarende flertydigheder (antallet er $n - 1$ når dimensionen er n , se afsnit 7.3.4), og her er der ikke længere symmetri. Her er det naturligt konsekvent at vælge det alternativ som man forventer er det mest sandsynlige.

6.3.7 Estimationsmetoden. I stedet kan man benytte estimationsmetoden, men i højere dimensioner kan det være meget omstændeligt at foretage et valg ud fra en beregnet centerværdi. Endvidere er antallet af flertydige tegl normalt ganske lavt.

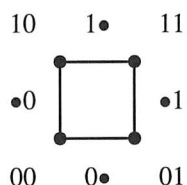
6.3.8 Mangel på viden. Endelig skal man osse bemærke at det er overmod at sige at man har løst flertydighedsproblemet. Problemet er underbestemt og man kan i almindelighed kun benytte „gæt“ til at afgøre flertydighed. Disse kan dog være mere eller mindre kvalificerede.

6.4 Egenskaber ved tegl

Nu studeres tegl og deres symmetrier. Dette kan senere anvendes til at opstille en effektiv algoritme til fastlæggelse af facetter i kritiske tegl.



Figur 6.6: Eksempler på ækvivalente tegl.



Figur 6.7: Repræsentation af 0-celler og 1-celler i tegl.

6.4.1 „Ens“ tegl. Forskellige tegl kan have visse træk til fælles. For eksempel kan to tegl have de samme høje hjørner selv om værdifunktionen på teglets hjørner er forskellig. Det betyder at facetterne i de to tegl kan „minde“ om hinanden selv om skæringspunkterne har forskellig placering (se figur 6.6.A).

6.4.2 Stærk ækvivalens. Dette formaliseres ved at definere en ækvivalensrelation på tegl. Det bliver nødvendigt at definere endnu en ækvivalensrelation på tegl, og derfor vil denne form for ækvivalens blive betegnet med *stærk ækvivalens* da denne ækvivalensrelation har de *mindste* ækvivalensklasser.

Der vil straks blive bragt en mere præcis definition af stærk ækvivalens, men først introduceres den anden ækvivalensrelation der ikke overraskende kommer til at hedde...

6.4.3 Svag ækvivalens. Der findes tegl som ikke er stærkt ækvivalente, men som efter rotation af det ene tegl bliver „ens“ (det vil sige stærkt ækvivalente) som vist på figur 6.6.B. Denne form for ækvivalens vil kaldes for *svag ækvivalens* da den er „svagere“ end stærk ækvivalens — antallet af tegl der er svagt ækvivalent med et givet tegl er større end (eller lig med) antallet af tegl der er stærkt ækvivalent med teglet.

6.4.4 Anvendelse af ækvivalens. De netop beskrevne ækvivalensrelationer er naturligvis nyttige i analysen og konstruktionen af en tegl-algoritme. Dette kræver naturligvis definitionerne formaliseres så de kan håndteres matematisk, lissom der må opstilles en repræsentation af tegl og facetter så de kan manipuleres med en computer.

6.4.5 Repræsentation af tegl. Et tegl τ er karakteriseret ved de fire 0-celler indeholdt i teglet hvorpå værdifunktionen V er defineret. Disse hjørner nummereres fra 0 til 3. Hvis teglet tænkes indlagt i et cartesisk koordinatsystem med nederste venstre hjørne i origo og sidelængde 1, som vist på figur 6.7, så nummereres hjørnerne i rækkefølgen $(0,0)$, $(1,0)$, $(0,1)$ og $(1,1)$. Hjørnets nummer kan repræsenteres i det binære talsystem, og i så fald bliver hjørne i skrevet som YX hvor (X,Y) er de cartesiske koordinater af hjørnet. Bemærk at rækkefølgen af de binære cifre ikke er umiddelbart intuitiv, men ikke desto mindre den korrekte.

6.4.6 Definition. En 0-celle x i teglet τ har nummeret $n(x) = YX$ tildelt som beskrevet ovenfor.

6.4.7 Definition. (Høje hjørner.) For teglet τ defineres de høje hjørner som mængden

$$h(\tau) = \{x \in \tau \mid |x| = 0, V(x) \geq T\}$$

6.4.8 Definition. (Signatur.) Teglets signatur defineres som

$$s(\tau) = \sum_{x \in h(\tau)} n(x) \quad (6.3)$$

Signaturen s kan fås ved at opskrive et binært tal der har cifret 1 på plads i hvis hjørne i i teglet er højt (ellers cifret 0). Da der er fire hjørner i et tegl og da hvert hjørne kan være enten højt eller lavt fås $2^4 = 16$ mulige signaturer og der gælder $0 \leq s \leq 15$. Endvidere haves følgende...

6.4.9 Sætning. Lad τ_1 og τ_2 være to tegl. Følgende udsagn gælder da

$$h(\tau_1) = h(\tau_2) \quad \text{netop hvis} \quad s(\tau_1) = s(\tau_2).$$

6.5 Tegl og gruppeteori

Den uformelle beskrivelse af ækvivalens mellem tegl vil nu blive formaliseret. Den matematiske disciplin der benyttes er *gruppeteori* og dens sammenhæng med *symmetrier*.

6.5.1 Definition. (Stærk ækvivalens.) To tegl τ_1 og τ_2 siges at være stærkt ækvivalente hvis de har samme mængde af høje hjørner. Det vil sige

$$\tau_1 \sim_s \tau_2 \quad \text{netop hvis} \quad h(\tau_1) = h(\tau_2).$$

Ifølge sætning 6.4.9 har to stærkt ækvivalente tegl samme signatur og der er derfor seksten forskellige stærke ækvivalensklasser.

6.5.2 Definition. Mængden af stærke ækvivalensklasser betegnes S . Der gælder at $|S| = 16$, og hvert af elementerne $\sigma \in S$ er identificeret ved sin signatur $s(\sigma) = s(\tau)$ hvor τ er en repræsentant for σ .

Bemærk at det giver mening at definere signaturen $s(\sigma)$ af en ækvivalensklasse $\sigma \in S$ da alle teglene $\tau \in \sigma$ har samme signatur.

Når forvirring ikke er mulig vil elementet $\sigma \in S$ med signaturen $s(\sigma) = a$ blive betegnet med tallet a . For eksempel vil ækvivalensklassen som teglet i figur 6.6.A repræsenterer blive betegnet med $1 + 4 + 8 = 13$. Med denne notation kan man skrive

$$S = \{0, 1, \dots, 15\}.$$

Elementerne i S kaldes for *tegl* selv om det egentlig er ækvivalensklasser af tegl. Kun når det er nødvendigt at præcisere at elementet er en samling af flere tegl vil benyttes den „korrekte“ betegnelse *ækvivalensklasse*.

6.5.3 Symmetrier af tegl. Mængden S har en struktur som fastlægges dels af sammenhængen mellem 0-cellerne, dels af 0-cellernes lav/høj status. Man kan permutere 0-cellerne således at denne struktur bibeholdes og det svarer netop til en „rotation“ af teglet. Det er centralt at facetten i et kritisk tegl ikke forandres under denne operation (bortset fra at den „roterer“ med teglet), men dette er osse opfyldt (dette er for eksempel illustreret på figur 6.6.B). *Symmetrigruppen* som beskriver denne symmetrioperation er den cykliske gruppe af orden fire. Lad ρ være den operation der roterer teglet en kvart omgang mod uret. Dette element kan være *frembringer* for gruppen der så kan skrives på formen

$$G = \{1, \rho, \rho^2, \rho^3\}.$$

6.5.4 Gruppevirkning. Elementerne i gruppen *virker* på teglene gennem den netop beskrevne „rotation“ eller afbildning $G \times S \rightarrow S$, og virkningen af elementet $g \in G$ på teglet $\tau \in S$ skrives som $g(\tau)$.

6.5.5 Svag ækvivalens blev oprindeligt beskrevet som en ækvivalensrelation på mængden af tegl, men det er mere bekvemt at definere den som en relation på S . Dette er helt acceptabelt da to stærkt ækvivalente tegl altid er svagt ækvivalente. Nu kan den beskrevne symmetrigruppe anvendes ved en formel definition af svag ækvivalens.

6.5.6 Definition. (Svag ækvivalens.) Givet $\tau_1, \tau_2 \in S$ defineres svag ækvivalens $\sim_w$ som

$$\tau_1 \sim_w \tau_2 \quad \text{netop hvis} \quad \exists g \in G : g(\tau_1) = \tau_2.$$

Mængden af svage ækvivalensklasser betegnes med W .

Det næste emne bliver hvordan man bestemmer mængden W .

6.5.7 Baner og banerum. Givet et tegl $\tau \in S$ så er *banen* af τ under G 's virkning

$$G(\tau) = \{g(\tau) \mid g \in G\}.$$

Det er klart at banen $G(\tau)$ er den svage ækvivalensklasse for τ . Mængden af svage ækvivalensklasser er dermed den samme som mængden af baner, osse kaldet *banerummet*, der skrives som

$$W = S/G$$

og udtales „ S modulo G “.

6.5.8 Antallet af svage ækvivalensklasser er det samme som antallet af baner $|S/G|$ kan nu bestemmes ved anvendelse af *Polya-Burnsides formel* der i denne sammenhæng antager formen

$$|S/G| = \frac{1}{|G|} \sum_{g \in G} |S^g| \quad (6.4)$$

hvor $S^g = \{\tau \in S \mid g(\tau) = \tau\}$ er fikspunkterne i S for g .

I tabel 6.8 er der for hvert $g \in G$ beregnet mængden S^g og tallet $|S^g|$. Dette indsættes i (6.4) og man får

$$|S/G| = \frac{1}{|G|} \sum_{g \in G} |S^g| = \frac{16 + 2 + 4 + 2}{4} = 6$$

som er antallet af svage ækvivalensklasser.

g	S^g	$ S^g $
1	$\{0, 1, \dots, 15\}$	16
ρ	$\{0, 15\}$	2
ρ^2	$\{0, 6, 9, 15\}$	4
ρ^3	$\{0, 15\}$	2

Tabel 6.8: Størrelser som indgår i beregningen af antallet af svage og ækvivalensklasser ved Polya-Burnsides formel.

6.5.9 Bestemmelse af baner foregår ved at man bestemmer en repræsentant fra hver bane. En sådan mængde kaldes osse for en *transversal*. Følgende observationer kan hjælpe til at bestemme en transversal:

1. Alle tegl i en given bane må have det samme antal høje hjørner. Da der kan være fra nul til fire høje hjørner kan man analysere de fem tilfælde hver for sig.
2. Når tilfældet med m høje hjørner betragtes så er der $K_{4,m}$ mulige fordelinger af høje hjørner i teglet hvor $K_{n,q} = n! / (q! (n - q)!)$.

En anden nyttig størrelse er...

6.5.10 Banelængden, det vil sige størrelsen af hver af de svage ækvivalensklasser. Denne fås fra en gruppeteoretisk sætning: Givet teglet τ gælder der

$$|G(\tau)| = \frac{|G|}{|\tau G|} \quad (6.5)$$

hvor $\tau G = \{g \in G \mid g(\tau) = \tau\}$ er...

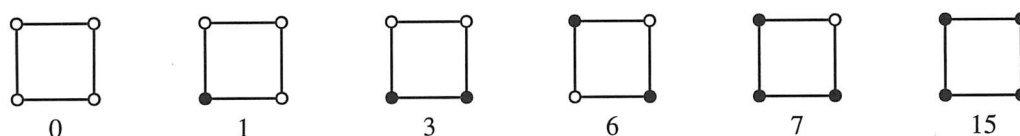
6.5.11 Isotropigruppen for τ . Isotropigruppen kaldes osse somme tider for *stabilisatoren* for τ og denne mængde indeholder de gruppeelementer som har τ som fikspunkt. Isotropigruppen er en undergruppe i G , og G har foruden de to trivielle undergrupper $\{1\}$ og G kun $\{1, \rho^2\}$ som undergruppe.

Hvis τ ingen symmetri har så er isotropigruppen $\tau G = \{1\}$, og banelængden bliver $|G|/|\tau G| = 4/1 = 4$. Hvis τ er fuldstændig symmetrisk (det vil sige hvis alle $g \in G$ har τ som fikspunkt) så er isotropigruppen $\tau G = G$, og banelængden bliver $|G|/|\tau G| = 4/4 = 1$. Endelig kan isotropigruppen være $\tau G = \{1, \rho^2\}$. Gruppeelementet ρ^2 svarer til en rotation en halv omgang, men svarer osse til en spejling i teglets centrum. Derfor vil tegl der har spejlsymmetri (men ikke fuldstændig symmetri) som isotropigruppe have $\tau G = \{1, \rho^2\}$, og banelængden bliver $|G|/|\tau G| = 4/2 = 2$.

6.5.12 Konstruktion af transversal. I tabel 6.9 er det demonstreret hvordan en transversal kan bestemmes. Det vides på forhånd at der er seks elementer i transversalen. I tabellens første kolonne er angivet antallet af høje hjørner $|h|$, og i den anden kolonne er antallet af tegl med det antal høje hjørner beregnet. Beregningen foregår derefter ved at der for hver række i tabellen vælges et tegl τ med det ønskede antal høje hjørner $|h(\tau)|$. Symmetrien af τ eller ækvivalent isotropigruppen τG bestemmes, og

$ h $	$K_{4, h }$	τ	τG	$ G(\tau) $
0	1	0	G	1
1	4	1	$\{1\}$	4
2	6	3	$\{1\}$	4
		6	$\{1, \rho^2\}$	2
3	4	7	$\{1\}$	4
4	1	15	G	1

Tabel 6.9: Bestemmelse af de svage ækvivalensklasser.



Figur 6.10: Grafisk beskrivelse af de seks repræsentanter for de svage ækvivalensklasser. Tallet der står under hvert af teglene er teglets signatur.

udtrykket (6.5) benyttes til at bestemme længden af banen $|G(\tau)|$. De tegl som er angivet i tabellen er osse vist grafisk i figur 6.10 hvor deres symmetri hurtigt kan ses.

Hvis banelængden, som er angivet i femte kolonne, svarer til tallet som står i anden kolonne, så er der kun én bane med netop det antal høje hjørner som er angivet i første kolonne. I det tilfælde hvor der er flere baner med samme antal høje hjørner må der gættes på endnu et tegl τ for hvilket banelængden beregnes indtil summen af banelængderne i femte kolonne svarer til antallet af tegl med det givne antal høje hjørner.

Dette er tilfældet når antallet af høje hjørner er to. Først vælges teglet $\tau = 3$, men banelængden for dette tegl er kun $|G(3)| = 4$, og da der skal være i alt $K_{4,2} = 6$ tegl må der findes endnu et tegl med to høje hjørner som ikke ligger i banen $G(3)$. Sådan et tegl er $\tau = 6$, og da dette tegl er spejlsymmetrisk bliver banelængden $|G(6)| = 2$. Alle banerne er dermed fundet for tegl med to høje hjørner idet $|G(3)| + |G(6)| = K_{4,2}$.

Den ønskede transversal er nu teglene som optræder i den tredje kolonne i tabel 6.9, det vil sige

$$\cdot = \{0, 1, 3, 6, 7, 15\}.$$

For fuldstændighedens skyld kan man nu sige at hvis den svage ækvivalensklasse af $\tau \in S$ skrives som $[\tau]_w$ så have

$$W = \{[\tau]_w \mid \tau \in \cdot\} = \{[0]_w, [1]_w, [3]_w, [6]_w, [7]_w, [15]_w\}.$$

6.6 Algoritmen

Nu kan den ønskede algoritme opstilles. Den vil blive beskrevet „oppefra og ned“ ved gentagen forfinelse.

6.6.1 Algoritme. (Todimensional tegl-algoritme.) Inddata til algoritmen er 2-billedet $P = ((X, \mathcal{X}), V, C)$ og tærskelværdien T . Uddata er mængden af facetter F . Mængden af 2-tegl i et billede P betegnes med $\text{tegl}_2(P)$.

```

1   $F := \emptyset$ 
2  For hver  $\tau \in \text{tegl}_2(P)$  gør
3      For hver  $(\nu_1, \nu_2) \in f^1(\tau)$  gør
4           $F := F \cup \{(\nu_1, \nu_2)\}$ 

```

Denne beskrivelse af algoritmen er næsten trivial — det virkelige problem ligger i at beregne $f^1(\tau)$ som anvendes i linie 3. Dette gøres ved introduktionen af en...

6.6.2 Symbolsk facet. Dette er en „facet“ hvor skæringspunkternes endelige placering i de kritiske kanter ikke er fastlagt. I stedet holdes der blot rede på hvilke kanter i teglet facetten løber imellem.

Man kan symbolsk repræsentere en kant i teglet ved at benytte et tocifret tal i et særligt tegralssystem. Ud over cifrene 0 og 1 optræder osse cifret $\bullet$ som dog kun må optræde på netop én plads i tallet. Tallet $YX = 0\bullet$ skal forstås således at y -værdien for kanten er 0 mens x -værdien kan „løbe frit“ det vil sige ligge i intervallet $]0; 1[$. De fire kanter $\bullet 0$, $0\bullet$, $1\bullet$ og $\bullet 1$ er vist på figur 6.7. Et skæringspunkt er placeret i en kant og kan derfor repræsenteres på samme måde.

Mængden af symbolske facetter for et givet tegl bliver da (ordnede) par af skæringspunkter repræsenteret som beskrevet. For eksempel bliver mængden af symbolske facetter for teglet vist til venstre på figur 6.4.B

$$\{(0\bullet, \bullet 1), (1\bullet, \bullet 0)\}.$$

Når en kant er repræsenteret som beskrevet fås de to hjørner (0-celler i teglet) som er 0-skelettet af kanten ved funktionerne

$$q_0(YX) = YX|_{\bullet=0}, \quad q_1(YX) = YX|_{\bullet=1} \quad (6.6)$$

hvor cifret $\bullet$ erstattes af henholdsvis 0 og 1.

6.6.3 Algoritme. (Beregning af facet.) Inddata til funktionen er et tegl τ og uddata er mængden af facetter F i teglet. Oplysninger om billedet (det vil sige koordinater og værdier) samt tærskelværdien indgår implicit. Der anvendes en funktion $p^1(\tau)$, defineret nedenfor, som giver mængden af 1-facetter for τ på symbolsk form. Funktionen I til beregning af skæringspunkternes placering er ligeledes beskrevet nedenfor.

```

1  Funktion  $f^1(\tau)$ 
2       $F := \emptyset$ 
3      For hver  $(i_1, i_2) \in p^1(s(\tau))$  gør
4           $\nu_1 = I(\tau, i_1)$ 
5           $\nu_2 = I(\tau, i_2)$ 
6           $F := F \cup \{(\nu_1, \nu_2)\}$ 
7      Returner  $F$ 

```

Den eksakte placering af det symbolske skæringspunkt i i teglet τ fastlægges af funktionen $I(\tau, i)$ der er afledt af udtrykket (6.1) og har formen

$$I(\tau, i) = \frac{T - V(\tau, q_0(i))}{V(\tau, q_1(i)) - V(\tau, q_0(i))} (C(\tau, q_1(i)) - C(\tau, q_0(i))) + C(\tau, q_0(i)). \quad (6.7)$$

Funktionen $V(\tau, \nu)$ hvor τ er et tegl og ν er et hjørne i teglet er værdifunktionen V beregnet i det hjørne. På tilsvarende måde skal $C(\tau, \nu)$ forstås.

Der mangler en beskrivelse af funktionen p^1 .

6.6.4 Beregning af symbolsk facet. Stærkt ækvivalente tegl har (som regel) samme symbolske facet. I det flertydige tilfælde kan facetten dog være forskellige for forskellige tegl i samme ækvivalensklasse. Det kræves nu at alle tegl i samme stærke ækvivalensklasse har samme symbolske facet. Når τ_1 og τ_2 er tegl så er dette ækvivalent med

$$p^1(\tau_1) = p^1(\tau_2) \quad \text{netop hvis} \quad s(\tau_1) = s(\tau_2). \quad (6.8)$$

Kravet er at flertydighed afgøres ved at benytte det samme sammenhængende alternativ i alle tegl med samme signatur, og det er derfor ikke muligt at benytte estimationsmetoden til afgørelse af flertydighed. Senere i afsnit 6.6.12 vil det blive demonstreret hvordan denne metode alligevel kan introduceres i algoritmen.

Man skal bemærke at kravet (6.8) ikke automatisk sikrer at den konsekvente metode anvendes idet de to flertydige stærke ækvivalensklasser, 6 og 9, godt kan benytte hver sit sammenhængende alternativ.

6.6.5 Anvendelse af opslagstabel. Med kravet (6.8) kan signaturen nu anvendes til bestemmelse af den symbolske facet, for eksempel ved konstruktion af en opslagstabel L indekseret ved signatur som følger

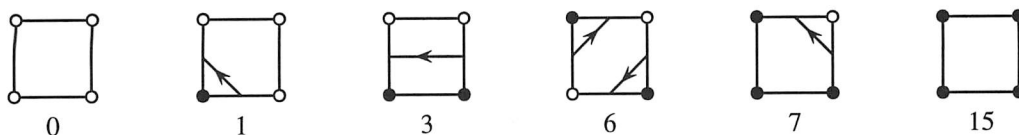
$$p^1(\tau) = L_{s(\tau)}. \quad (6.9)$$

Dette giver en særdeles effektiv algoritme da opslagstabellen L kan beregnes på forhånd.

6.6.6 Anvendelse af svag ækvivalens. Man kan konstruere opslagstabellen ved at gennemgå alle seksten stærke ækvivalensklasser og konstruere en symbolsk facet for hver af disse. I stedet kan man udnytte teglets symmetrier og nøjes med at konstruere en facet for hver af de svage ækvivalensklasser. Dette har den bivirkning at alle svagt ækvivalente tegl får „ens“ facetter. Her skal „ens“ forstås således at hvis $\tau_2 = g(\tau_1)$ så er $p^1(\tau_2) = p^1(g(\tau_1))$ hvor $g \in G$. Det betyder osse at de to flertydige tegl (som er i samme svage ækvivalensklasse) kommer til at få facetter med det samme sammenhængende alternativ, og derved bliver flertydighed afgjort ved anvendelse af den konsekvente metode.

6.6.7 Konstruktion af symbolsk facet. Nu vælges det lavt sammenhængende alternativ. For hver repræsentant for de svage ækvivalensklasser (se figur 6.10) konstrueres en facet. Disse er vist grafisk på figur 6.11 og skrevet op symbolsk i tabel 6.12. Dette benyttes så til at konstruere en opslagstabel med følgende...

6.6.8 Algoritme. (Konstruktion af opslagstabel.) Inddata er en transversal $\cdot$ hvor p^1 er defineret for hvert element samt symmetrigruppen G . Uddata er opslagstabellen L med et opslag for hver signatur.



Figur 6.11: Facetter for repræsentanter for de svage ækvivalensklasser. Tallet der står under hvert af teglene er teglets signatur. I dette tilfælde er flertydighed afgjort ved anvendelse af det lavt sammenhængende alternativ (teglet med signatur 6).

τ	$p^1(\tau)$
0	$\emptyset$
1	$\{(0\bullet, \bullet 0)\}$
3	$\{(\bullet 1, \bullet 0)\}$
6	$\{(\bullet 1, 0\bullet), (\bullet 0, 1\bullet)\}$
7	$\{(\bullet 1, 1\bullet)\}$
15	$\emptyset$

Tabel 6.12: Symbolske facetter for repræsentanter for de svage ækvivalensklasser. I dette tilfælde er flertydighed afgjort ved anvendelse af det lavt sammenhængende alternativ.

- 1 For hver $\tau \in \cdot$ gør
- 2 For hver $g \in G$ gør
- 3 $L_{s(g(\tau))} := g(p^1(\tau))$

Bemærk at der ofte findes forskellige $g_1, g_2 \in G$ så $g_1(\tau) = g_2(\tau)$ og dermed vil den samme position i L blive tildelt en værdi flere gange. Værdien er dog altid den samme.

6.6.9 Gruppevirkning på tegl. I forbindelse med beregningen af signaturen $s(g(\tau))$ i algoritmen ovenfor er det nødvendigt at kende $h(g(\tau))$ da signaturen er en sum over høje hjørner, og det er derfor nødvendigt at beskrive hvad der sker med et hjørne i teglet under G 's virkning. Dette hjørne er repræsenteret ved det binære tal $YX = n(x)$ som beskrevet i afsnit 6.4.5, og ρ 's virkning på dette hjørne bliver

$$\rho(YX) = X\bar{Y} \quad (6.10)$$

hvor $\bar{0} = 1$ og $\bar{1} = 0$. Man kan lade ρ virke på de enkelte hjørner i $h(\tau)$ og får

$$h(\rho(\tau)) = \{\rho(x) \mid x \in h(\tau)\}.$$

For et vilkårligt $g \in G$ findes der et $m \in \{0, \dots, 3\}$ således at $g = \rho^m$. Gruppevirkningen af g fås derfor ved m gentagelser af ρ 's virkning.

6.6.10 Gruppevirkning på symbolske facetter. Hvis operationen $\bar{\cdot}$ udvides til elementet $\bullet$ således at $\bar{\bullet} = \bullet$, så er gruppevirkningen på et skæringspunkt repræsenteret ved det ternære tal YX tillige givet ved udtrykket (6.10). Ved at lade ρ virke på hvert af skæringspunkterne i mængden af symbolske facetter fås virkningen på en symbolsk facet

$$\rho(p^1(\tau)) = \left\{ \left(\rho(\nu_1), \rho(\nu_2) \right) \mid (\nu_1, \nu_2) \in p^1(\tau) \right\}$$

hvilket er en størrelse der skal beregnes i algoritmen ovenfor.

τ	$L_{s(\tau)}$
0	$\emptyset$
1	$\{(0\bullet, \bullet 0)\}$
2	$\{(\bullet 1, 0\bullet)\}$
3	$\{(\bullet 1, \bullet 0)\}$
4	$\{(\bullet 0, 1\bullet)\}$
5	$\{(0\bullet, 1\bullet)\}$
6	$\{(\bullet 1, 0\bullet), (\bullet 0, 1\bullet)\}$
7	$\{(\bullet 1, 1\bullet)\}$
8	$\{(1\bullet, \bullet 1)\}$
9	$\{(1\bullet, \bullet 1), (0\bullet, \bullet 0)\}$
10	$\{(1\bullet, 0\bullet)\}$
11	$\{(1\bullet, \bullet 0)\}$
12	$\{(\bullet 0, \bullet 1)\}$
13	$\{(0\bullet, \bullet 1)\}$
14	$\{(\bullet 0, 0\bullet)\}$
15	$\emptyset$

Tabel 6.13: Den endelige opslagstabel.

6.6.11 Den endelige opslagstabel kan nu beregnes. Den er vist i tabel 6.13. I stedet for det lavt sammenhængende alternativ kunne man have anvendt det højt sammenhængende alternativ og tabellen ville da have et andet indhold på plads 6 og 9 som svarer til de to flertydige facetter.

6.6.12 Estimationsmetoden kan i stedet anvendes. Dette kræver en modifikation i beregningen af $p^1(\tau)$ der før blev beregnet ved et tabelopslag (6.9).

6.6.13 Algoritme. Der anvendes et prædikat $\text{flertydig}(\tau)$ som er sandt hvis teglet τ er flertydigt. Derudover haves to tabeller L_h og L_l som giver den symbolske facet for teglet τ i det lavt henholdsvis det højt sammehængende alternativ. Hvis τ ikke er flertydigt er $L_{s(\tau)} = L_{s(\tau)}^h = L_{s(\tau)}^l$.

Hvilket alternativ der anvendes afgøres ved at estimere en værdi i centret af teglet. Hvis denne værdi er høj benyttes det højt sammenhængende alternativ og ellers benyttes det lavt sammenhængende alternativ. Estimatet beregnes her ved midling af værdien af de fire hjørner i teglet, men højere ordens metoder der inddrager værdien af hjørner uden for teglet kan osse anvendes.

```

1  Funktion  $p^1(\tau)$ 
2      Hvis flertydig( $\tau$ ) så
3           $v := \frac{1}{4} \sum_{x \in {}^o0(\tau)} V(x)$ 
4      Hvis  $v \geq T$  så
5          Returner  $L_{s(\tau)}^l$ 
6      Ellers
7          Returner  $L_{s(\tau)}^h$ 
8      Ellers
9          Returner  $L_{s(\tau)}$ 

```

I praksis kan prædikatet flertydig(τ) og tabellerne L_h og L_l implementeres ved at tabellen L på de flertydige positioner 6 og 9 indeholder et særligt mærke der viser at positionen er flertydig og samtidig angiver værdien af L_h og L_l .

Kapitel 7

En tredimensional tegl-algoritme

- 7.1 Indledning
- 7.2 Facetter
- 7.3 Flertydighed
- 7.4 Tegl og gruppeteori
- 7.5 Algoritmen
- 7.6 Anvendelse af den todimensionale algoritme
- 7.7 Repræsentation af facetter
- 7.8 Implementering i praksis
- 7.9 En konkret implementering

I dette kapitel vil den todimensionale tegl-algoritme blive generaliseret til tre dimensioner. Dette giver en algoritme som minder om „marching cubes“ algoritmen beskrevet af Lorensen og Cline [1987] og Cline et al. [1988]. I afsnit 7.6 beskrives en variant af algoritmen der minder om algoritmen beskrevet af Wyvill et al. [1986]. Derefter vil forskellige aspekter af en konkret implementering blive omtalt.

7.1 Indledning

Mange af begreberne som er defineret i forbindelse med den todimensionale algoritme kan direkte overføres til det tredimensionale tilfælde med kun få ændringer. Billedet $P = ((X, \mathcal{X}), V, C)$ har tre dimensioner i stedet for to, og ækvipotentialkurven givet ved tærskelværdien T erstattes med *ækvipotentialfladen*

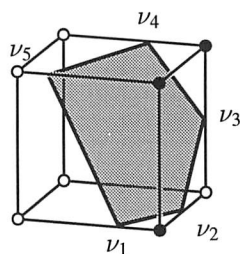
$$\tilde{E} = \{x \in X \mid \tilde{V}(x) = T\}.$$

Denne ækvipotentialflade skal tilnærmes af en *konturflade* E eller blot *kontur* som adskiller høje fra lave hjørner i billedet.

De anvendte tegl er nu 3-tegl som defineret i definition 6.1.7. Ofte kaldes 0-cellerne for „hjørner“ og 1-cellerne for „kanter“ som det var tilfældet i beskrivelsen af den todimensionale algoritme. Desuden vil 2-cellerne blive kaldt for *sider* når det er praktisk. Betegnelsen „tegl“ vil, når intet andet er nævnt, dække over 3-tegl.

Høje og lave hjørner i et tegl defineres som før, og et kritisk tegl er naturligvis et tegl der både indeholder høje og lave hjørner. Et kritisk tegl vil indeholde kritiske sider og kritiske kanter.

Så langt, så godt. Generalisationen af facetter fra to til tre dimensioner kræver lidt mere omhu.



Figur 7.1: Eksempel på et kritisk tegl og den tilhørende facet.

7.2 Facetter

7.2.1 Den todimensionale facet. I den todimensionale algoritme bestod byggestenene i konturen af liniestykker, osse kaldet 1-facetter. I det tredimensionale tilfælde vil der på „overfladen“ af et kritisk tegl, nærmere bestemt på de kritiske sider, kunne trækkes rette linier mellem skæringspunkterne som ligger i de kritiske kanter (se figur 7.1). Disse rette linier vil udgøre en lukket kurve (eller nu og da flere lukkede kurver) og 2-facetten er den flade som *udspændes* af denne kurve. For at konkretisere meningen af „udspænder“ så kan man forestille sig at den lukkede kurve udspænder en sæbehinde — facetten er så sæbehinden, og dens form vil være styret af fysikkens love.

I praksis er det uhensigtsmæssigt at implementere „sæbehinder“ på computer, og man kan i stedet betragte facetten som en (i reglen) „ikke-plan polygon“ i rummet. Sådan en „polygon“ kan trianguleres således at den opbygges af (plane) trekanter. Hvordan dette gøres er emnet for afsnit 8.1 — indtil videre er sæbehinde metaforen tilstrækkelig.

7.2.2 Bemærkning. Allerede her fås en indikation af hvordan metoden kan generaliseres til højere dimensioner idet 2-facetten opbygges af 1-facetter som igen er opbygget af 0-facetter (skæringspunkter).

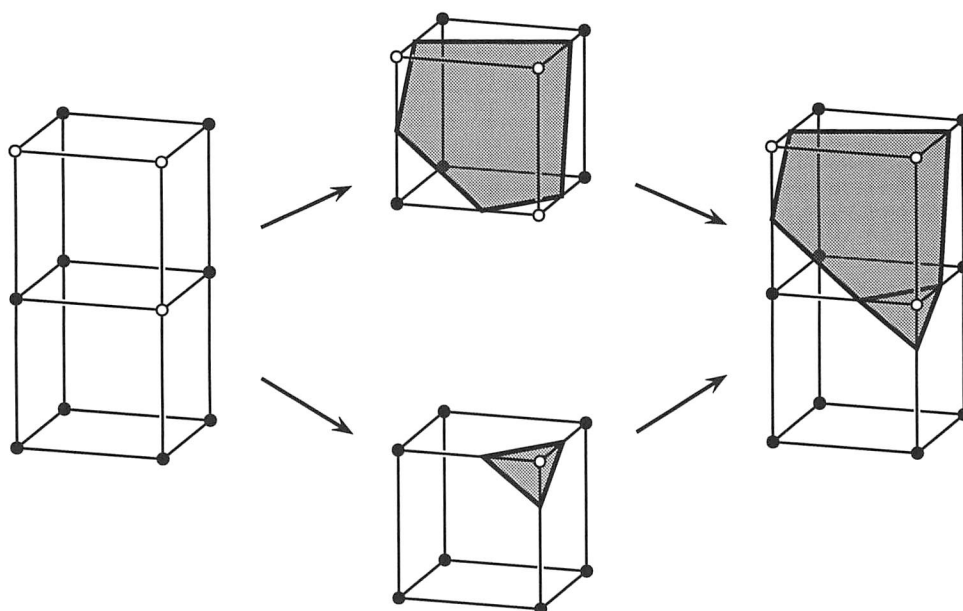
7.2.3 Definition. (2-Facet.) Lad der være givet et kritisk tegl τ med n skæringspunkter $\nu_1, \nu_2, \dots, \nu_n$ der kan forbindes med liniestykker til en lukket kurve der forløber i de kritiske sider af τ . Den (i reglen ikke-plane) polygon der udspændes af kurven er 2-facetten. Denne er orienteret og repræsenteret ved m -tuplen $(\nu_1, \nu_2, \dots, \nu_n)$. Mere generelt bliver mængden af 2-facetter i τ betegnet med

$$f^2(\tau) = \{ (\nu_1^i, \nu_2^i, \dots, \nu_{n_i}^i) \mid i = 1, \dots, m \}$$

hvor m er antallet af 2-facetter i τ . Hver 2-facet, indekseret ved i , er opbygget af n_i 0-facetter (skæringspunkter) og dermed af n_i 1-facetter.

Senere vil det blive beskrevet hvorledes 1-facetter sammensættes til 2-facetter samt hvad der menes med „udspænder“.

7.2.4 Orientering. Facettens repræsentation $(\nu_1, \nu_2, \dots, \nu_n)$ beskriver hvorledes de enkelte skæringspunkter er forbundet: ν_1 forbindes med en ret linie til ν_2 som forbindes videre til ν_3 og så videre som vist på figur 7.1. Kurven lukkes ved at ν_n forbindes til ν_1 . På den måde tildeles facetten en orientering, og ved brug af højrehåndsreglen induceres der en „normalvektor“ på facetten: Man griber „omkring“ facetten med højre hånd således at fingrene peger samme vej som kurven er orienteret. Tommelfingerretningen definerer nu „normalvektoren“ eller mere præcist en inderside og en yderside af facetten. Algoritmen konstrueres således at „normalvektoren“ fastlægger den side der „vender mod“ de høje hjørner.



Figur 7.2: Et eksempel på konsistens mellem facetter i to kritiske nabo-tegl.

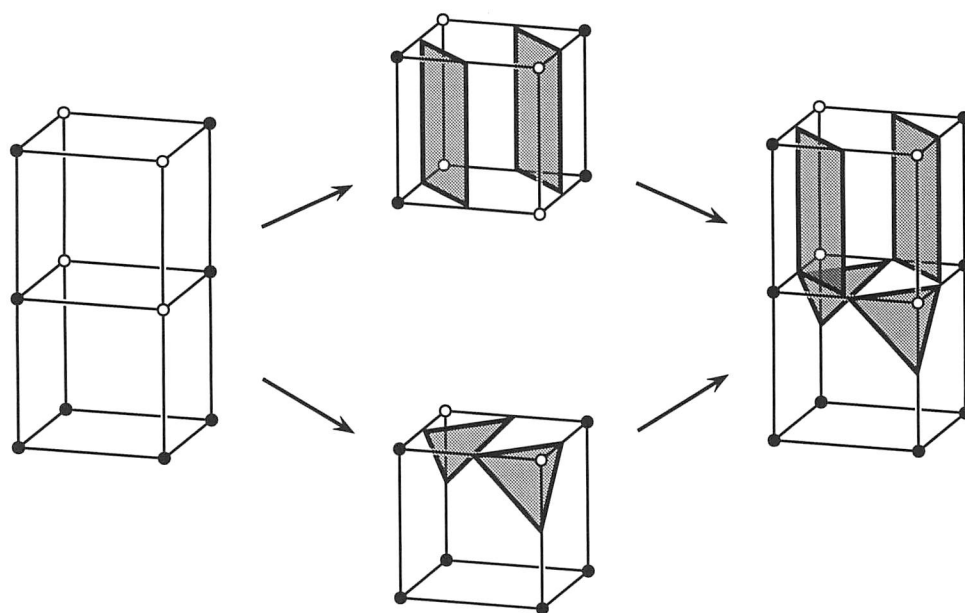
7.2.5 Konsistens. Kravet om konsistens er fundamentalt i „del- og hersk“ strategien som anvendes af tegl-algoritmerne. På figur 7.2 er vist to kritiske nabo-tegl. Hvert af de to tegl betragtes for sig, og en facet konstrueres. Når teglene „sættes sammen“ skal facetterne fra hvert af teglene passe sammen. Dette ses osse at være tilfældet på figuren, og man kan se at konsistens kan sikres ved at den kritiske side som deles af de to tegl indeholder den samme 1-facet i begge tegl. Dette krav er formuleret formelt i definition 6.2.10.

7.2.6 Inkonsistens. På figur 7.3 ses et eksempel på inkonsistens. Selv om hvert af teglene tilsyneladende indeholder en gyldig facet der adskiller de høje fra de lave hjørner så dannes der et „hul“ i konturen når de to facetter sættes sammen. Via „hullet“ kan man komme fra den „høje“ til den „lave“ side af konturen. En korrekt algoritme må derfor være i stand til at undgå den slags huller.

7.2.7 Flertydighed. Studerer man eksemplet lidt nøjere vil man opdage at den kritiske side hvor problemet opstår er flertydig (i todimensional forstand). Skæringspunkterne er fastlagt konsistent ved udtrykket (6.1), men når siden er flertydig har man mulighed for at anvende det lavt eller det højt sammenhængende alternativ. Hullet opstår netop når der benyttes forskellige alternativer i hvert af de kritiske tegl.

Benyttes den konsekvente metode til at afgøre flertydighed så vil det samme alternativ blive benyttet hver gang et flertydigt tegl optræder og konsistens vil være sikret. Benyttes estimationsmetoden vil det være den interpolerede centerværdi af den kritiske side der fastlægger hvilket alternativ der skal anvendes, og det vil osse sikre konsistens.

7.2.8 „Marching cubes“ algoritmen er bemærkelsesværdig fordi konsistens ikke er garanteret. Den inkonsistente kontur vist på figur 7.3 er netop konstrueret med denne



Figur 7.3: Et eksempel på inkonsistens mellem facetter i to nabo-tegl.

algoritme. Med notationen anvendt af Lorensen og Cline [1987] så er det øverste tegl af type 10, mens det nederste tegl er af type 3.

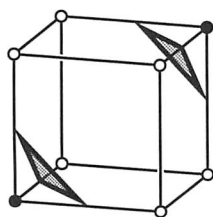
7.3 Flertydighed

Flertydighed har som beskrevet betydning når man ønsker at sikre konsistens, men der optræder osse nye former for flertydighed i højere dimensioner.

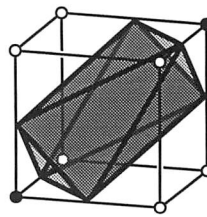
7.3.1 Flertydighed i to dimensioner. Flertydighed i to dimensioner opstår når der er en „diagonal struktur“ i et 2-tegl. Spørgsmålet er om det er de lave hjørner eller de høje hjørner der skal forbindes, og dette svarer til det lavt og det højt sammenhængende alternativ.

7.3.2 Flertydighed i tre dimensioner. I et 3-tegl er der flere slags diagonaler: I hver af de seks 2-celler er der to diagonaler, og her kan den netop beskrive flertydighed optræde, men 3-teglets 3-celle indeholder osse diagonaler, fire i alt, som går mellem modstående 0-celler i teglet. Hvis alle hjørner i et tegl er lave, undtagen to hjørner som er endepunkter for en sådan diagonal, så får man et flertydigt tilfælde af en ny type. På figur 7.4.A er vist et tegl med en facet som adskiller de to høje hjørner, mens der på figur 7.4.B er vist et tilsvarende tegl med en facet der *ikke* adskiller de to høje hjørner. Bemærk at facetterne er de samme selv om høj og lav ombyttes.

7.3.3 Bemærkning. I 3-teglets 3-celle er der osse seks diagonaler om hvilke der er symmetri, men disse diagonaler går mellem 1-celler i teglet, og netop af den grund er de ikke interessante i forbindelse med flertydighed.



A) Adskillende facetter.



B) En forbindende facet.

Figur 7.4: Et 3-flertydigt tegl.

7.3.4 Generel definition af flertydighed. For at skelne mellem de forskellige typer flertydighed kaldes den først beskrevne type flertydighed for 2-flertydighed og den sidste beskrevne form for 3-flertydighed. Mere generelt kan følgende definitioner anvendes:

7.3.5 Definition. (Diagonal.) Et n -tegl har én n -celle og denne har en række symmetriakser udspændt af 0-cellerne i teglet. Disse diagonaler kaldes for n -diagonaler.

De to 0-celler der udspænder en bestemt diagonal definerer diagonalen og hvis 0-cellerne begge er høje (lave) siges diagonalen at være høj (lav).

7.3.6 Bemærkning. Antallet af 0-celler i et n -tegl er 2^n så derfor må antallet af n -diagonaler i et n -tegl være $2^n/2 = 2^{(n-1)}$.

7.3.7 Definition. (Flertydighed.) Et n -tegl som indeholder udelukkende lave (høje) 0-celler på nær to 0-celler som udspænder en høj (lav) n -diagonal siges at være n -flertydigt. Den pågældende diagonal kaldes for den flertydige diagonal.

Bemærk at et 3-tegl indeholder seks 2-tegl så der kan både optræde 3- og 2-flertydighed i et 3-tegl (dog ikke samtidig). Mere generelt er der mulighed for $n - 1$ forskellige slags flertydighed i et n -tegl.

7.3.8 Adskillende eller forbindende alternativ. Facetter i et n -flertydigt n -tegl kan enten *adskille* de to 0-celler som udspænder den flertydige diagonal som det for eksempel er tilfældet på figur 7.4.A, eller *forbinde* de to 0-celler som vist på figur 7.4.B. Men hvilket af de to alternativer skal man vælge?

7.3.9 Valg af alternativ. Det virker umiddelbart mest naturligt at vælge det adskillende alternativ da afstanden mellem de to høje hjørner er større end afstanden fra et højt hjørne til de nærmeste lave hjørner. Dette faktum (eller måske mangel på overvejelse) er årsag til at alle eksisterende tegl-algoritmer mig bekendt benytter det adskillende alternativ.

Konsekvent anvendelse af det modsatte alternativ, det vil sige det forbindende alternativ, kan virke noget aparte at benytte, men man kan kombinere de to alternativer ved at anvende...

7.3.10 Estimationsmetoden. Dette er en enkel generalisation af den tilsvarende estimationsmetode anvendt på 2-flertydighed. Analogt med det 2-flertydige tilfælde estimerer man en værdi af feltet i 3-teglets centrum, for eksempel ved at beregne gennemsnittet af feltværdien i de otte hjørner i teglet. Hvis den flertydige diagonal er høj (lav) og den estimerede centerværdi er høj (lav) så benyttes det forbindende alternativ. Ellers benyttes det adskillende alternativ.

Hvis den flertydige diagonal er høj (lav) så vil der i estimeringen af centerværdien på ovenfor beskrevne måde indgå seks lave (høje) hjørner og kun to høje (lave) hjørner. Hvis disse hjørner indgår med lige vægt i estimatet (og hvad skulle de ellers?) så vil det betyde at kun en fjerdedel af alle estimater i middel er høje (lave), og dermed vil det adskillende alternativ i middel benyttes i tre fjerdedele af tilfældene. Dette understøtter udsagnet om at det adskillende alternativ er det hyppigst forekommende.

7.3.11 Valg af metode. Problemet er, som altid når man prøver at udtale sig om noget „inden i“ et tegl, at man ikke ved noget som helst — man kender kun feltets værdier i teglets hjørner. Problemet er derfor underbestemt og løsningen vil altid være et gæt (men på den anden side så findes der gode og dårlige gæt).

På den baggrund vil det adskillende alternativ altid blive anvendt i forbindelse med 3-flertydighed fremover. At tillade det forbindende alternativ vil komplicere algoritmerne unødigt meget i forhold til en eventuel gevinst i nøjagtigheden af den konstruerede kontur.

Endelig skal det nævnes at definitionen af 2-facetter skal modificeres hvis man vil tillade facetter af den type som er vist på figur 7.4.A. En sådan facet har nemlig genus ét (hvor genus nul er det sædvanlige), og dette gør det for eksempel mere subtilt at orientere facetten.

7.4 Tegl og gruppeteori

Her introduceres i analogi med det todimensionale tilfælde stærk og svag ækvivalens samt teglets symmetrigruppe.

7.4.1 Nummerering af 0-celler. Som i det todimensionale tilfælde tænkes 3-teglet placeret i et cartesisk koordinatsystem med nederste venstre hjørne i origo og med sidelængden 1. En 0-celle x tildeles et trecifret binært nummer $n(x) = ZYX$ hvor (X, Y, Z) er de cartesiske koordinater af 0-cellen.

Definition 6.4.7 af høje hjørner og definition 6.4.8 af signatur kan nu genanvendes. Det er osse klart at sætning 6.4.9, om sammenhængen mellem mængden af høje hjørner og teglets signatur, gælder i det tredimensionale tilfælde.

7.4.2 Stærk ækvivalens defineres analogt med det todimensionale tilfælde. Det vil sige at to tegl er stærk ækvivalente hvis de har samme signatur (definition 6.5.1). Mængden af stærke ækvivalensklasser betegnes endnu engang med S og man kan skrive

$$S = \{0, \dots, 255\}$$

idet der er $2^8 = 256$ forskellige signaturer.

7.4.3 Svag ækvivalens defineres ud fra en symmetrigruppe for S . Hvis G er symmetrigruppen der virker på S gennem virkningen $G \times S \rightarrow S$ så er mængden af svage ækvivalensklasser $W = S/G$.

7.4.4 Valg af symmetrigruppe er delikat. Man kan vælge kubusens 24 orienteringsbevarende symmetrier, som ud over identiteten, består af seks kvarte rotationer omkring en sideakse, tre halve rotationer omkring en sideakse, seks halve rotationer omkring en kantakse og otte trediedele rotationer omkring en hjørneakse. En *sideakse* er en

Type	Elementer
Identitet	1
En kvart rotation om sideakse	$\rho_x, \rho_y, \rho_z, \rho_x^3, \rho_y^3, \rho_z^3$
En halv rotation om sideakse	$\rho_x^2, \rho_y^2, \rho_z^2$
En halv rotation om kantakse	$\rho_x^2\rho_z, \rho_x^2\rho_y, \rho_y^2\rho_x, \rho_y^2\rho_z, \rho_z^2\rho_y, \rho_z^2\rho_x$
En trediedel rotation om hjørneakse	$\rho_x\rho_y, \rho_y\rho_z, \rho_z\rho_x, \rho_x\rho_z, \rho_x^3\rho_z, \rho_y^3\rho_x, \rho_z^3\rho_y, \rho_x^2\rho_z\rho_x$

Tabel 7.5: Opsummering af orienteringsbevarende symmetrier af kubusen skrevet som produkter af de tre kvarte rotationer i positiv omløbsretning om sideakserne.

symmetriakse som går gennem kubusens sider. *Kantakser* og *hjørneakser* defineres tilsvarende.

Væsentligt er det at en facets adskillende egenskab i et kritisk tegl ikke forandres under disse symmetrioperationer.

7.4.5 Opskrivning af symmetrigruppe. Det viser sig at denne symmetrigruppe kan frembringes af tre elementer ρ_x , ρ_y og ρ_z som hver svarer til en kvart rotation i positiv omløbsretning omkring en sideakse parallel med henholdsvis x -, y - og z -aksen. Alle elementer i gruppen kan skrives som et produkt af højst fire frembringere. Bestemmelse af gruppen på denne form er en enkel opgave i gruppeteori og kan nemt gøres med et lille computerprogram. Her bringes én af de mulige beskrivelser (tabel 7.5) hvor identiteten 1 for enkelhedens skyld optræder. Bemærk at $1 = \rho_x^0 = \rho_x^4$ så udsagnet om de tre frembringere er stadig korrekt.

Opskrivningen er naturligvis ikke særlig „elegant“ i matematisk forstand, men den er nyttig når gruppen skal håndteres af en computer da man kun behøver at implementere de tre frembringere.

7.4.6 Bemærkning. Er man tvivl om hvordan gruppeelementer sammensættes så kan det for eksempel nævnes at $\rho_y^2\rho_x$ skal forstås som den transformation af kubus der fås, ved først at anvende ρ_y to gange på kubus efterfulgt af ρ_x én gang. Den som fristes til at tro at rækkefølgen af transformationerne er omvendt er nok blevet forledt til den fejlagtige antagelse, på grund af den upraktiske notation man sædvanligvis benytter til at opskrive funktioner med. Inden for algebraen benytter man ofte notation $(x)f$ i stedet $f(x)$ og med dette in mente skulle forvirringen forsvinde som dug for solen — eller hvad? Og hvis man tror at koordinatsystemet „roteres“ under transformationerne så er man osse galt på den — det er kun kubusen som transformeres.

7.4.7 Spejling i centrum. Lidt eftertanke viser dog at man osse kan tillade at tilføje spejlingen i kubusens centrum til symmetrigruppen — facettens adskillende egenskab bevares under en spejling. Denne symmetrioperation permuterer med de øvrige, og derved haves en symmetrigruppe med 48 elementer som beskriver alle symmetrier af kubus. Spejlingen i kubusens centrum betegnes med μ og den tilføjes til sættet af frembringere. Nu kan alle elementer i gruppen skrives som et produkt af højst fem frembringere.

En ting skal man dog være opmærksom på og det er løst sagt at „højre og venstre ombyttes“ når μ virker på et tegl. Denne ombytning betyder ikke noget for teglets signatur, men en facet i teglet har en orientering givet ved højrehåndsreglen, som beskrevet i afsnit 7.2.4, og denne orientering bliver vendt under en spejling.

7.4.8 Bestemmelse af svage ækvivalensklasser. Bestemmelse af en transversal $\cdot$ og dermed af banerne W kan nu gøres som beskrevet i afsnit 6.5. Da antallet af stærke ækvivalensklasser er helt oppe på 256 kan det være svært af foretage beregningerne i hånden. I stedet kan følgende algoritme anvendes:

7.4.9 Algoritme. Inddata er gruppen G og uddata er en transversal $\cdot$. Undervejs benyttes mængden S der til at begynde med svarer til mængden af stærke ækvivalensklasser, men gradvist udtyndes ved fjernelse af baner $G(\tau)$.

```

1   $S := \{0, \dots, 255\}$ 
2   $\cdot = \emptyset$ 
3  Så længe  $S \neq \emptyset$  gør
4       $\tau := \min S$ 
5       $\cdot := \cdot \cup \{\tau\}$ 
6      For hver  $g \in G$  gør
7           $S := S - \{g(\tau)\}$ 

```

I linie 4 vælges den næste repræsentant for en endnu ikke behandlet bane. Da minimum over signaturer anvendes her, får algoritmen den særlige egenskab at hver repræsentant for en svag ækvivalensklasse $\tau \in \cdot$ har mindst mulig signatur.

7.4.10 Gruppevirkning på tegl. Det er nødvendigt at præcisere hvorledes $g(\tau)$ beregnes for $g \in G$ og $\tau \in S$. Den teknik der er beskrevet i afsnit 6.6.9 benyttes igen, men udtrykket (6.10) erstattes med

$$\begin{aligned}\rho_x(ZYX) &= Y\bar{Z}X, \\ \rho_y(ZYX) &= \bar{X}YZ, \\ \rho_z(ZYX) &= ZX\bar{Y}, \\ \mu(ZYX) &= \overline{ZYX}.\end{aligned}$$

Endvidere skal orienteringen af en facet vendes under μ 's virkning. Hvis en facet med m skæringspunkter er givet ved listen $(ZYX_1, \dots, ZYX_m)$ så fås μ 's virkning ved udtrykket

$$\mu(ZYX_1, ZYX_2, \dots, ZYX_m) = (\mu(ZYX_m), \dots, \mu(ZYX_2), \mu(ZYX_1)).$$

7.4.11 De svage ækvivalensklasser er givet ved transversalen som bestemmes af algoritme 7.4.9, og resultatet bliver

$$\begin{aligned}\cdot &= \{0, 1, 3, 6, 7, 15, 22, 23, 24, 25, 27, 30, 31, \\ &\quad 60, 61, 63, 105, 107, 111, 126, 127, 255\}.\end{aligned}\tag{7.1}$$

Det ses at der er 22 svage ækvivalensklasser.

7.4.12 Bemærkning. Havde man i stedet indskrænket sig til at benytte orienteringsbevarende symmetrier er det klart at antallet af svage ækvivalensklasser ville være større da gruppen er mindre. Overraskende er det måske at antallet blot er én større, det vil sige 23. Det betyder at der kun er én svag ækvivalensklasse som kommer i en venstre- og i en højrevendt udgave. Det drejer sig om klassen repræsenteret af 27 — dens „spejlbillede“ er 29, og man kan vælge den som repræsentant for den 23. ækvivalensklasse der optræder når G ikke indeholder μ .

7.5 Algoritmen

Det er nu muligt at opskrive en tredimensional tegl-algoritme. I dette afsnit demonstreres det hvordan en opslagstabel kan anvendes — senere vil andre muligheder blive diskuteret. Først bringes dog den overordnede algoritme som er fælles for alle varianterne.

7.5.1 Algoritme. (Tredimensional tegl-algoritme.) Inddata til algoritmen er 3-billedet $P = ((X, \mathcal{X}), V, C)$ og tærskelværdien T . Uddata er mængden af facetter F . Mængden af 3-tegl i billedet P betegnes med $\text{tegl}_3(P)$.

```
1   $F := \emptyset$ 
2  For hver  $\tau \in \text{tegl}_3(P)$  gør
3      For hver  $(\nu_1, \dots, \nu_n) \in f^2(\tau)$  gør
4           $F := F \cup \{(\nu_1, \dots, \nu_n)\}$ 
```

7.5.2 Symbolske facetter. Som i det todimensionale tilfælde anvendes *symbolske facetter* til at beregne $f^2(\tau)$. Ideerne fra afsnit 6.6.2 overføres direkte med den modifikation at skæringspunktet i den symbolske facet nu er et *trecifret* ternært tal ZYX stadig med den begrænsning at cifret $\bullet$ kun må optræde på én plads. Der gælder at gruppevirkningen på 0-celler, som beskrevet i afsnit 7.4.10, osse kan anvendes på symbolske facetter.

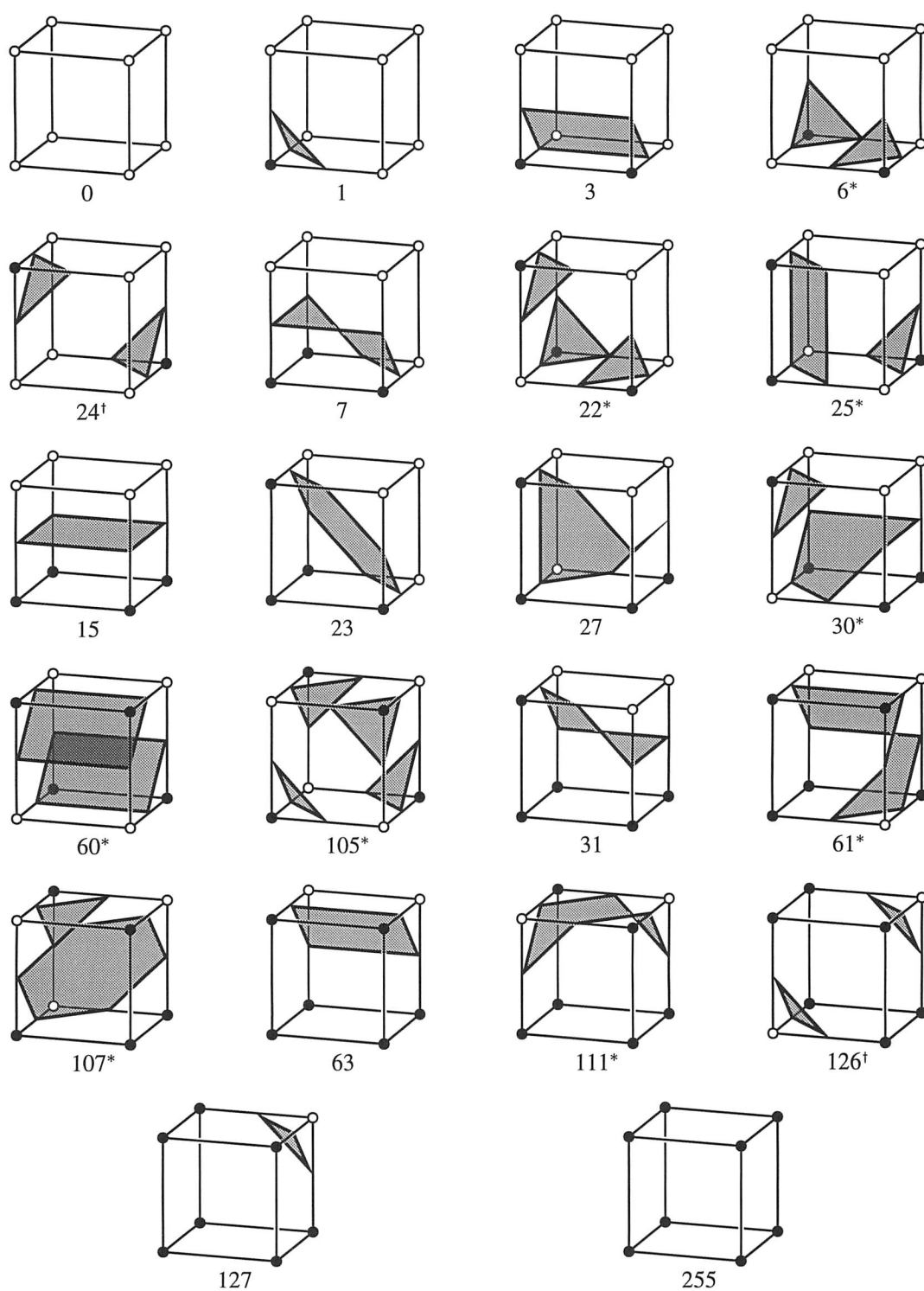
7.5.3 Algoritme. (Beregning af facet.) Inddata til funktionen er et tegl τ og uddata er en mængde af facetter F . Det antages at der er en funktion $p^2(\tau)$ som giver mængden af 2-facetter for τ på symbolsk form. Funktionen I til beregning af skæringspunkterne er givet ved udtrykket (6.7) hvor funktionerne q_0 og q_1 , givet ved (6.6), modificeres til tre dimensioner ved at erstatte „ YX “ med „ ZYX “.

```
1  Funktion  $f^2(\tau)$ 
2       $F := \emptyset$ 
3      For hver  $(i_1, \dots, i_n) \in p^2(\tau)$  gør
4          For hver  $k \in \{1, \dots, n\}$  gør
5               $\nu_k = I(\tau, i_k)$ 
6               $F := F \cup \{(\nu_1, \dots, \nu_n)\}$ 
7      Returner  $F$ 
```

7.5.4 Konsistens. Det er vigtigt at $p^2(\tau)$ sikrer konsistens. Som tidligere beskrevet kan dette sikres hvis 2-flertydighed afgøres enten ved den konsekvente metode eller ved estimationsmetoden. Den sidste løsningsmodel er ganske kompliceret sammenlignet med den første idet der findes 3-tegl hvor alle seks 2-tegl er flertydige. Dette giver $2^6 = 64$ forskellige facetter for tegl i samme stærke ækvivalensklasse!

7.5.5 Anvendelse af opslagstabel. Funktionen $p^2(\tau)$ kan implementeres ved anvendelse af en opslagstabel L som beskrevet i afsnit 6.6.5, det vil sige ved at sætte $p^2(\tau) = L_{s(\tau)}$.

7.5.6 Konstruktion af opslagstabel foretages ved anvendelse af algoritme 6.6.8. Det kræver at den symbolske facet $p^2(\tau)$ fastlægges for hver svag ækvivalensklasse $\tau \in W$ eller ækvivalent, for hvert tegl τ i transversalen $\cdot$. For at sikre konsistens skal den konsekvente metode anvendes i tilfælde af 2-flertydighed. Endvidere skal hver facet



Figur 7.6: Repræsentanter for de svage ækvivalensklasser med tilhørende lavt sammenhængende facet. Teglene er ordnede efter antallet af høje hjørner og signatur. Tegl som er mærket med en stjerne er 2-flertydige, og tegl som er mærket med et kors er 3-flertydige.

τ	$P^2(\tau)$
0	$\emptyset$
1	$\{(\bullet 00, 0\bullet 0, 00\bullet)\}$
3	$\{(\bullet 01, \bullet 00, 0\bullet 0, 0\bullet 1)\}$
6*	$\{(0\bullet 1, \bullet 01, 00\bullet), (0\bullet 0, \bullet 10, 01\bullet)\}$
24†	$\{(01\bullet, \bullet 11, 0\bullet 1), (1\bullet 0, \bullet 00, 10\bullet)\}$
7	$\{(\bullet 01, \bullet 00, \bullet 10, 01\bullet, 0\bullet 1)\}$
22*	$\{(0\bullet 1, \bullet 01, 00\bullet), (0\bullet 0, \bullet 10, 01\bullet), (1\bullet 0, \bullet 00, 10\bullet)\}$
25*	$\{(10\bullet, 1\bullet 0, 0\bullet 0, 00\bullet), (01\bullet, \bullet 11, 0\bullet 1)\}$
15	$\{(\bullet 10, \bullet 11, \bullet 01, \bullet 00)\}$
23	$\{(\bullet 01, 10\bullet, 1\bullet 0, \bullet 10, 01\bullet, 0\bullet 1)\}$
27	$\{(\bullet 11, \bullet 01, 10\bullet, 1\bullet 0, 0\bullet 0, 01\bullet)\}$
30*	$\{(0\bullet 0, \bullet 10, \bullet 11, \bullet 01, 00\bullet), (1\bullet 0, \bullet 00, 10\bullet)\}$
60*	$\{(0\bullet 0, \bullet 10, \bullet 11, 0\bullet 1), (1\bullet 0, \bullet 00, \bullet 01, 1\bullet 1)\}$
105*	$\{(\bullet 00, 0\bullet 0, 00\bullet), (01\bullet, \bullet 11, 0\bullet 1), (\bullet 01, 1\bullet 1, 10\bullet), (\bullet 10, 1\bullet 0, 11\bullet)\}$
31	$\{(1\bullet 0, \bullet 10, \bullet 11, \bullet 01, 10\bullet)\}$
61*	$\{(\bullet 01, 1\bullet 1, 1\bullet 0, \bullet 10, \bullet 11, 0\bullet 1, 00\bullet)\}$
107*	$\{(\bullet 11, 1\bullet 1, 10\bullet, \bullet 00, 0\bullet 0, 01\bullet), (\bullet 10, 1\bullet 0, 11\bullet)\}$
63	$\{(1\bullet 0, \bullet 10, \bullet 11, 1\bullet 1)\}$
111*	$\{(\bullet 00, 1\bullet 0, 11\bullet, \bullet 11, 1\bullet 1, 10\bullet)\}$
126†	$\{(0\bullet 0, \bullet 00, 00\bullet), (11\bullet, \bullet 11, 1\bullet 1)\}$
127	$\{(11\bullet, \bullet 11, 1\bullet 1)\}$
255	$\emptyset$

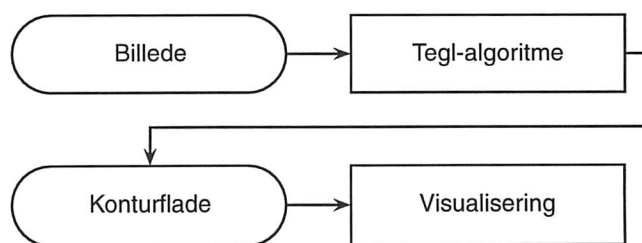
Tabel 7.7: Symbolske facetter for repræsentanter for de svage ækvivalensklasser. Det lavt sammenhængende alternativ er anvendt til at afgøre flertydighed. Tegl som er mærket med en stjerne er 2-flertydige, og tegl som er mærket med et kors er 3-flertydige.

orienteres, som beskrevet i afsnit 7.2.4, så det sikres at de høje hjørner altid er på „undersiden“.

På figur 7.6 er alle tegl $\tau \in \cdot$ med tilhørende facetter vist. For at sikre konsistens er det lavt sammenhængende alternativ benyttet, men man kunne lige så godt have anvendt det højt sammenhængende. Disse facetter er osse opsummeret i symbolsk form i tabel 7.7.

Nu bliver nytten af svag ækvivalens for alvor klar: Man behøver blot at bestemme facetter for 22 tegl. Derefter anvendes algoritme 6.6.8 til at konstruere de øvrige $256 - 22 = 234$ facetter.

7.5.7 Anvendelse af algoritme. For at præcisere hvordan algoritmen anvendes er der på figur 7.8 et diagram der viser hvordan databehandlingen foregår. De afrundede kasser repræsenterer data og de kantede kasser repræsenterer processer. Det sidste trin, visualiseringen, er ikke noget der vil blive beskrevet her — mange programmer og standardbiblioteker er tilgængelige lissom, der findes en omfattende litteratur om emnet, for eksempel [Foley et al. 1990, Mortenson 1989].



Figur 7.8: Diagram over anvendelsen af den tredimensionale algoritme.

7.6 Anvendelse af den todimensionale algoritme

Her vil nu blive beskrevet en radikalt anderledes måde hvorpå den symbolske facet $p^2(\tau)$ kan beregnes. Den bygger på det faktum at en 2-facet er opbygget af 1-facetter.

7.6.1 Rekursion over lavere dimensioner. Metoden går kort sagt ud på at der i hver af 3-teglets 2-tegl konstrueres 1-facetter. Disse vil udgøre et antal lukkede, orienterede kurver og herfra fås 2-facetterne. Figur 7.9 viser et eksempel på hvorledes dette foregår i praksis. Da en figur som bekendt fortæller mere end tusinde ord overlades ordet til figuren — studér den nøje...

Men jeg slipper ikke for nogle uddybende forklaringer.

7.6.2 Orientering af 2-facetter. Den endelige 2-facet, som er vist på figur 7.9.D, er orienteret som beskrevet i afsnit 7.2.4. „Indersiden“, det vil sige den side der vender mod de høje hjørner, fås ved anvendelse af tommelfingerreglen på den lukkede, orienterede kurve af 1-facetter som udspænder 2-facetten.

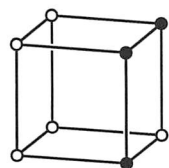
For at få denne orientering må man have to andre „orienteringer“ i spil:

7.6.3 Orientering af tegl. Et 3-tegl har på naturlig måde en „inderside“ og en „yderside“. Dette er vist med pile på figur 7.9.B hvor pilens retning angiver „ud“ af facetten. Når den todimensionale algoritme anvendes på hvert af 2-teglerne så skal algoritmen „betragte“ teglet „ude fra“, det vil sige mod pilens retning.

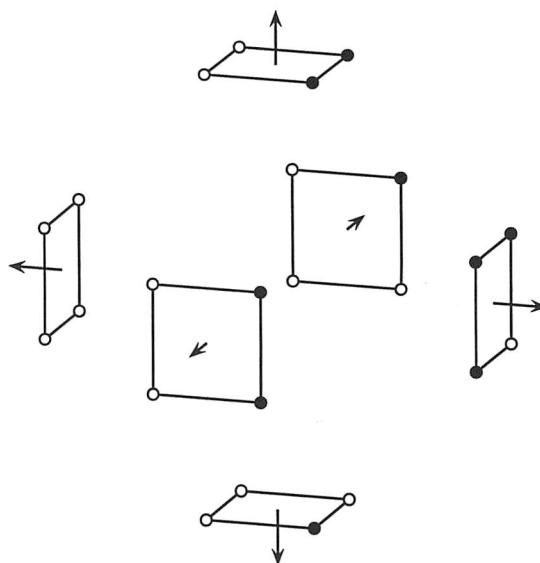
7.6.4 Orientering af 1-facetter. De resulterende 1-facetter, som er vist på figur 7.9.C, er hver orienteret som beskrevet i afsnit 6.2.6. Overholdes den ovenfor beskrevne regel for hvordan den todimensionale algoritme skal „betragte“ hvert 2-tegl så opnår man at 1-facetterne kan sættes sammen til en lukket kurve, som oven i købet har den egenskab at „indersiden“ er bestemt af tommelfingerreglen.

7.6.5 Algoritme. Inddata er et 3-tegl τ og uddata er mængden af facetter F . Undervejs skal man iterere over 1-tegl og 2-tegl i τ og disse mængder betegnes med henholdsvis $\text{tegl}_1(\tau)$ og $\text{tegl}_2(\tau)$. Første del af algoritmen konstruerer en tabel A som beskriver konnektiviteten mellem skæringspunkterne i teglet ved anvendelse af den éndimensionale symbolske facet $p^1(\tau)$.

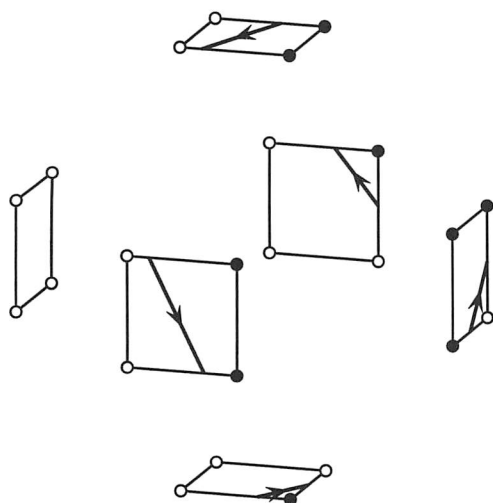
I tabellen A er der et opslag for hvert muligt skæringspunkt (det vil sige for hvert 1-tegl i 3-teglet). Hvis de tolv mulig skæringspunkter $00\bullet$, $01\bullet$, ... nummereres fra 1 til 12 på passende vis så kan A implementeres som en tabel indekseret fra 1 til 12.



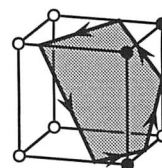
A) Det kritiske 3-tegl.



B) Teglet deles op i seks 2-tegl.



C) Den todimensionale algoritme anvendes på hvert 2-tegl.



D) De forskellige 1-facetter kombineres til en lukket, orienteret kurve der udspænder en 2-facet.

Figur 7.9: Eksemplificering af hvorledes det tredimensionale problem kan løses ved anvendelse af den todimensionale algoritme.

i	Kode	A_i
1	00●	2
2	0●1	7
3	01●	□
4	0●0	□
5	●00	□
6	●01	□
7	●11	11
8	●10	□
9	10●	1
10	1●1	□
11	11●	9
12	1●0	□

Tabel 7.10: Indholdet af tabel A efter gennemløb af (linierne 2–6 i algoritme 7.6.5 når teglet på figur 7.9 behandles.

Hvis skæringspunktet med indekset i hører til facetten, det vil sige hvis 1-teglet med indeks i er kritisk, så skal A_i være indeks af næste skæringspunkt i 2-facetten. Hvis derimod skæringspunkt i ikke hører med til facetten så skal A_i have særlig værdi $\square$ som med ovenstående nummerering passende kan implementeres som værdien 0.

Med denne definition af tabellen A vil den for et givet tegl „indeholde“ nul eller flere *cykler* hvor en cykel skal forstås på følgende vis: Indfør notationen $A_i^{m+1} = A_{A_i^m}$ og sæt $A_i^0 = i$. Så vil en cykel med k elementer som indeholder skæringspunkt i have formen

$$(i, A_i, A_i^2, \dots, A_i^k)$$

hvor der gælder at $i = A_i^{k+1}$. I tabel 7.10 er vist et eksempel på indholdet af A når teglet i figur 7.9 behandles, og her er der én cykel $(1, 2, 7, 11, 9)$.

Først opstilles tabellen A hved at hvert 2-tegl undersøges.

```

1  Funktion  $p^2(\tau)$ 
2      For hver  $i \in \text{tegl}_1(\tau)$  gør
3           $A_i := \square$ 
4      For hver  $\sigma \in \text{tegl}_2(\tau)$  gør
5           $(i_1, i_2) := p^1(\sigma)$ 
6           $A_{i_1} = i_2$ 
```

På dette tidspunkt vil A indeholde nul, én eller flere cykler svarende til et tilsvarende antal symbolske 2-facetter i 3-teglet. Bemærk at det er vigtigt hvorledes 2-teglet $\sigma \in \text{tegl}_2(\tau)$ præsenteres for den todimensionale algoritme. Som tidligere beskrevet skal 2-teglet „betragtes ude fra“ for at 1-facetterne skal blive korrekt orienteret. Og hvis ikke 1-facetterne er korrekt orienteret, vil A ikke indeholde cykler da man så ikke kan gå fra det ene skæringspunkt til det næste på en cyklisk måde.

Algoritmen afsluttes ved at der konstrueres symbolske 2-facetter fra informationen i A . En cykel repræsenteret ved tuplen C fjernes fra A og tilføjes som symbolsk facet til mængden af facetter F , indtil A har værdien $\square$ på alle pladser.

```

7       $F := \emptyset$ 
8      Så længe  $\exists \mu : A_\mu \neq \square$  gør
9           $C := ()$ 
10          $A_\nu = \min\{A_\mu \mid A_\mu \neq \square\}$ 
11         Så længe  $A_\nu \neq \square$  gør
12              $C := (C, \nu)$ 
13              $\nu := A_\nu$ 
14              $A_\nu := \square$ 
15          $F := F \cup \{C\}$ 
16     Returner  $F$ 

```

7.6.6 Afgørelse af flertydighed med den netop beskrevne algoritme afhænger af hvorledes flertydighed afgøres af den todimensionale algoritme, som anvendes rekursivt via funktionen $p^1(\sigma)$. Hvis den konsekvente metode anvendes i den todimensionale algoritme vil den konsekvente metode osse blive anvendt af den tredimensionale algoritme til afgørelse af 2-flertydighed. Hvis derimod estimationsmetoden anvendes i det todimensionale tilfælde vil estimationsmetoden osse blive anvendt i det tredimensionale tilfælde til afgørelse af 2-flertydighed.

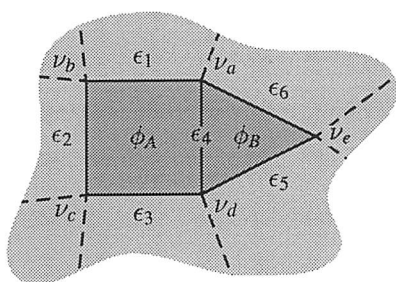
7.6.7 Implementering af estimationsmetoden. Dette giver en bekvem måde at implementere estimationsmetoden til afgørelse af 2-flertydighed i det tredimensionale tilfælde — man skal blot benytte algoritmen beskrevet ovenfor sammen med en todimensional algoritme der benytter estimationsmetoden. Problemet er at en algoritme baseret på anvendelse af en opslagstabel, vil være en del hurtigere end en algoritme baseret på rekursion over lavere dimensioner.

Løsningen er at kombinere de to algoritmer således at en opslagstabel benyttes i alle ikke-flertydige tilfælde, mens rekursion over lavere dimensioner benyttes i de flertydige tilfælde. Da antallet af flertydige tilfælde i reglen er lavt vil algoritmen være næsten lige så hurtig som én kun baseret på anvendelse af en opslagstabel. Den vil dog være noget mere kompliceret.

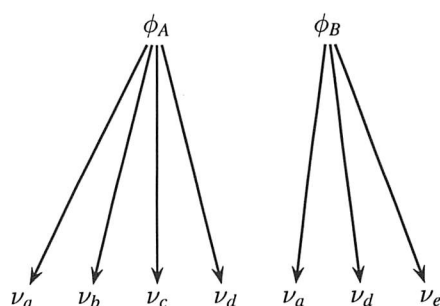
7.6.8 Afgørelse af 3-flertydighed. Denne idé kan osse benyttes til at afgøre 3-flertydighed, idet man så skal opstille en algoritme der er i stand til at skelne mellem det adskillende og det forbindende alternativ ved brug af estimationsmetoden. Denne algoritme skal så blot benyttes i de 3-flertydige tilfælde (det vil sige når den svage ækvivalensklasse er 24 eller 126).

7.7 Repræsentation af facetter

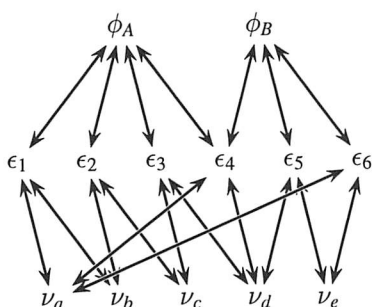
Når man vil implementere en algoritme må man naturligvis gøre sig nogle overvejelser omkring hvordan uddata, det vil sige facetterne, skal repræsenteres. For at illustrere de forskellige muligheder for repræsentation af konturflader opbygget af facetter er der på figur 7.11.A vist et lille fladestykke bestående af to 2-facetter ϕ_A og ϕ_B , seks 1-facetter $\epsilon_1, \epsilon_2, \epsilon_3, \epsilon_4, \epsilon_5$ og ϵ_6 og fem 0-facetter $\nu_a, \nu_b, \nu_c, \nu_d$ og ν_e .



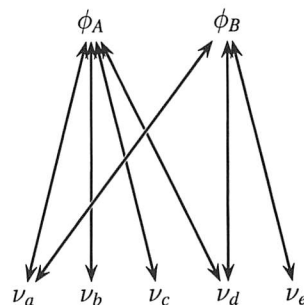
A) Et lille fladestykke.



B) Simpel beskrivelse af fladestykket.



C) Komplet beskrivelse af fladestykket givet ved konnektiviteten mellem 2-facetter, 1-facetter og 0-facetter.



D) Forenklet beskrivelse af fladestykket givet ved konnektiviteten mellem 2-facetter og 0-facetter.

Figur 7.11: Illustration af hvorledes en flade beskrevet af facetter kan repræsenteres ved konnektiviteten mellem 2-facetter, 1-facetter og 0-facetter.

7.7.1 En enkel repræsentation af dette fladestykke er illustreret på figur 7.11.B hvor hver 2-facet er repræsenteret ved en liste af 0-facetter. For eksempel kan ϕ_A skrives som $\phi_A = (v_a, v_b, v_c, v_d)$. Da hver 0-facet (i reglen) deles af flere 2-facetter vil hver 0-facet optræde flere gange i hver sin 2-facet. Dette er for eksemplet tilfældet for 0-facetten v_a . Med en sådan repræsentation vil det være svært at bestemme hvilke 2-facetter der tilhører en bestemt 0-facet, og dette er på figuren antydnet ved at forbindelseslinierne mellem 2-facetterne og 0-facetter er orienteret mod 0-facetterne.

7.7.2 En fuldstændig repræsentation af fladestykket er illustreret på figur 7.11.C hvor sammenhængen mellem 2-, 1- og 0-facetter er vist. Hver n -facet optræder netop én gang og det betyder at man er i stand til, for eksempel, at bestemme hvilke 2-facetter der hører til en given 0-facet. Dette er på figuren antydnet ved at forbindelseslinierne mellem facetterne er orienteret begge veje. Hvis man kun er interesseret i fladens „form“ behøver man blot at lagre de nedadgående forbindelseslinier. Er man interesseret i andet end fladens form, for eksempel i naboskab og sammenhæng mellem facetter, må de opadgående forbindelseslinier osse lagres. Et konkret eksempel hvor denne information

er nødvendig er ved anvendelse af nogle af de metoder til forenkling af fladen som er beskrevet i afsnit 8.3. Denne repræsentation af fladen er for øvrigt nært beslægtet med *winged-edge* repræsentationen [Foley et al. 1990].

7.7.3 En forenklet repræsentation. Et af problemerne ved den fuldstændige repræsentation er at informationen om 1-facetterne ofte ikke er nødvendig, men blot bidrager til et større forbrug af lagerplads. I stedet kan 1-facetterne helt udelades som illustreret på figur 7.11.D. Med denne repræsentation er det stadig muligt at besvare spørgsmål om sammenhæng mellem 0-facetter og 2-facetter såfremt facetten er orienteret. Det er denne type repræsentation som benyttes i den her beskrevne implementering af en tredimensional tegl-algoritme.

7.8 Implementering i praksis

7.8.1 Beregning af skæringspunkter. Uanset om man har tænkt sig at benytte en fladerepræsentation hvor hvert skæringspunkt (0-facet) gentages for hver facet den optræder i, eller en repræsentation hvor skæringspunktet kun optræder én gang, så kan det betale sig kun at beregne skæringspunktet én gang under konstruktionen af fladen. Det skyldes at beregningen af et skæringspunkt er ganske krævende (se (6.7)). I afsnit 8.2 beskrives det hvorledes der knyttes normaler til hvert skæringspunkt, og beregningen af denne normal komplicerer yderligere beregningen af skæringspunkt. Hvis man endvidere har tænkt sig at benytte en fladerepræsentation hvor hvert skæringspunkt kun optræder én gang, er det påkrævet at skæringspunkterne kun beregnes én gang.

7.8.2 Anvendelse af hash tabel. I praksis kan dette realiseres ved at der til hvert muligt skæringspunkt knyttes et entydigt *indeks*. Dette indeks kan så benyttes som nøgle i en *hash* tabel. Når et nyt skæringspunkt skal beregnes undersøges det først om skæringspunktet allerede findes i hash tabellen. Er dette tilfældet genanvendes det allerede beregnede skæringspunkt. På den anden side, hvis skæringspunktet ikke findes i hash tabellen beregnes et nyt skæringspunkt som placeres i tabellen. Grunden til at det er attraktivt at benytte hash tabeller, er at antallet af skæringspunkter i en konturflade i reglen er langt mindre end det totale antal mulige skæringspunkter i et 3-billede. Man behøver derfor ikke at afsætte plads til samtlige mulige skæringspunkter, men blot til et passende antal i hash tabellen. Med lidt snilde kan hash tabellen endda med jævne mellemrum skrives ud til en uddata fil og tømmes hvorved tabellen kan gøres endnu mindre.

7.8.3 Et entydigt indeks for skæringspunkter. Der er mulighed for ét skæringspunkt i hvert 1-tegl i billedet. Et entydigt indeks til beskrivelse af skæringspunkter kan derfor fås ved at konstruere et entydigt indeks til beskrivelse af 1-tegl. Lad dimensionen af billedet være $m_1 \times m_2 \times m_3$ og lad 1-teglet $\tau_{j_1 j_2 j_3}^i$ være givet ved

$$\tau_{j_1 j_2 j_3}^i = \sum_{\nu=1}^3 j_{\nu} e_{\nu} + t e_i, \quad 0 \leq t \leq 1$$

hvor e_1 , e_2 og e_3 er basisvektorerne i det sædvanlige, tredimensionale cartesiske koordinatsystem, og hvor $i = 1, 2, 3$ og $j_{\nu} = \{0, 1, \dots, m_{\nu} - 1\}$ (jævnfør (5.1)). Det er klart

at indekset H givet ved

$$H = ((im_3 + j_3)m_2 + j_2)m_1 + j_1 \quad (7.2)$$

er entydigt for hvert 1-tegl og dermed er det entydigt for hvert skæringspunkt.

7.8.4 Bemærkning. Da der kan optræde udartede facetter, som diskuteret i bemærkning 6.2.7, så kan flere forskellige skæringspunkter i praksis være placeret i ét og samme punkt. Anvendes udtrykket (7.2) til beregning af skæringspunktets indeks vil disse sammenfaldende eller *udartede* skæringspunkter have forskelligt indeks da såvel i , som j_1 , j_2 og j_3 varierer for forskellige 1-tegl der indeholder det udartede skæringspunkt.

Da sammenfaldende skæringspunkter er uønskede kan disse erstattes med et enkelt skæringspunkt ved at man under beregningen af skæringspunkter undersøger om skæringspunktet er udartet. Dette kan for eksempel gøres ved at undersøge om konstanten α (beskrevet i bemærkning 6.2.4) er lig med nul eller én.

Men hvilket indeks skal man anvende til sådan et skæringspunkt? Da skæringspunktet altid falder på en 0-celle så kan udtrykket 7.2 anvendes igen, blot man for j_1 , j_2 og j_3 anvender værdierne for 0-cellen (jævnfør (5.1)) og indsætter $i = 4$. Denne sidste „magiske“ værdi anvendes for at sikre at udartede skæringspunkter aldrig får det samme indeks som et ikke udartet skæringspunkt.

Med denne teknik vil en udartet facet indeholde et udartet skæringspunkt flere gange. Denne sekvens af udartede skæringspunkter slås sammen til en forekomst af skæringspunktet i facetten hvorved facetten får færre sider. Facetter, som på den måde ender med at have færre end tre sider, bidrager slet ikke til den endelige flade og fjernes helt fra uddata.

7.9 En konkret implementering

Jeg har foretaget en implementering af den i afsnit 7.6.7 beskrevne „kombinerede“ algoritme der forener anvendelsen af opslagstabel i de ikke-flertydige tilfælde med rekursiv anvendelse af den todimensionale algoritme i de flertydige tilfælde. Udartede facetter fjernes hvis de ikke bidrager til fladen. Endvidere anvendes en „smart“ buffer teknik således at det tredimensionale billede behandles „lagvis“. Hvis det tredimensionale billede tænkes opbygget af todimensionale billeder så behandles alle tegl mellem to sådanne billeder, hvorefter al information som ikke skal genanvendes i næste „lag“ skrives ud. Det gør at lagerpladsen som til stadighed anvendes er langt mindre end den samlede lagerplads som kræves af hele 3-billedet og hele konturfladen.

Programmet er skrevet i C++ og kan osse køres på noget så simpelt som en MS-DOS PC'er. Da dette er en 16 bit computer med 20 bit adressering (og med diverse tilfældige begrænsninger) er der dog grænser for hvor store rumfang der kan håndteres. Alligevel har jeg med succes håndteret et rumfang med cirka 100.000 billedpunkter af en halv underkæbe som resulterede i cirka 6.000 facetter og cirka 6.000 skæringspunkter. Programmet anvender 8 bit data i rumfanget, men med ganske få modifikation kan andre datatyper anvendes som, for eksempel, flydende-komma tal.

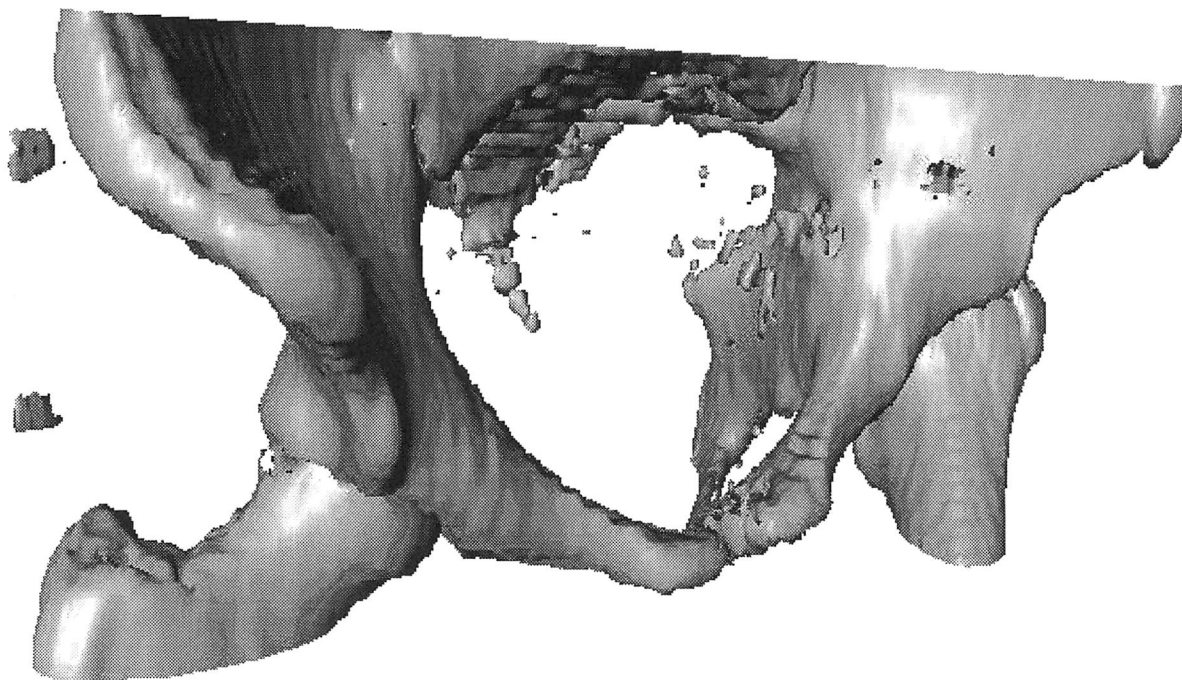
Tabel 7.12 viser nogle estimater for ydelsen af programmet på forskellige computere. Ovenfra og ned er det en Silicon Graphics Indigo², en Silicon Graphics Iris 4D/20, en „hurtig“ MS-DOS PC'er og en „langsom“ MS-DOS PC'er. De angivne værdier skal naturligvis kun opfattes som omtrentlige da forskellige inddata kan give forskellige resultater — for eksempel op til en faktor to langsommere eller hurtigere. Man kan se at et „typisk“ rumfang med $200 \times 200 \times 100 = 4.000.000$ tegl kan behandles på under et halvt minut af Indigo'en, mens Iris'en vil være mere end 10 minutter om dette. Det

Processor	Tegl per sekund
MIPS R4400 150 MHz	160.000
MIPS R2000 10 MHz	6.000
i486DX/2 66 MHz	6.000
i386 20 MHz	800

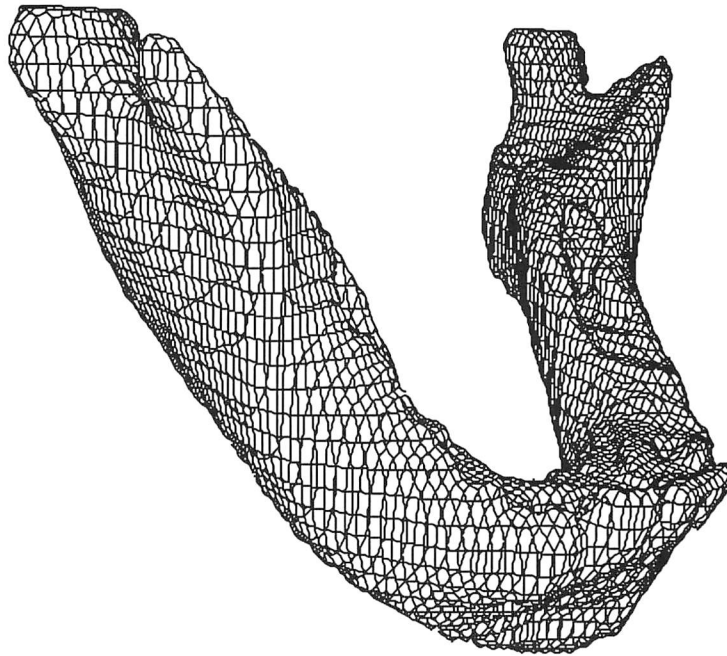
Tabel 7.12: Omtrentlig ydelse af den implementerede algoritme på forskellige computere.

betyder at det på en Indigo² er der mulighed for en „næsten“ interaktiv justering af tærskelværdien.

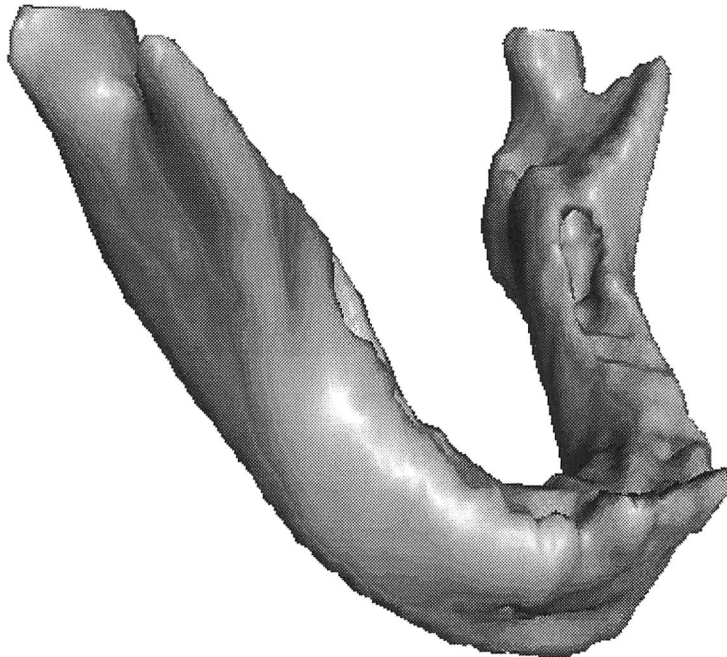
På de følgende sider er der vist nogle eksempler på nogle visualiseringer af forskellige konturflader. De er visualiseret på den før nævnte Indigo² med værktøjerne *Explorer* og *Inventor*. Fladerne er ikke triangulerede da dette ikke er nødvendigt med disse værktøjer. Alle illustrationerne, på nær den første, er lavet med Phong toning, og dette kræver at der i hvert skæringspunkt er fastlagt en normal. Hvordan dette er gjort er beskrevet i afsnit 8.2. Alle konturerne er fremstillet ud fra røntgentomogrammer, og de viser alle, på nær én, knoglestrukturer. På figur 7.16 er det demonstreret hvordan man ved at ændre tærskelværdien kan få en konturflade der beskriver en anden type væv.



Figur 7.13: En model af en hofte bestående af cirka 100.000 facetter.

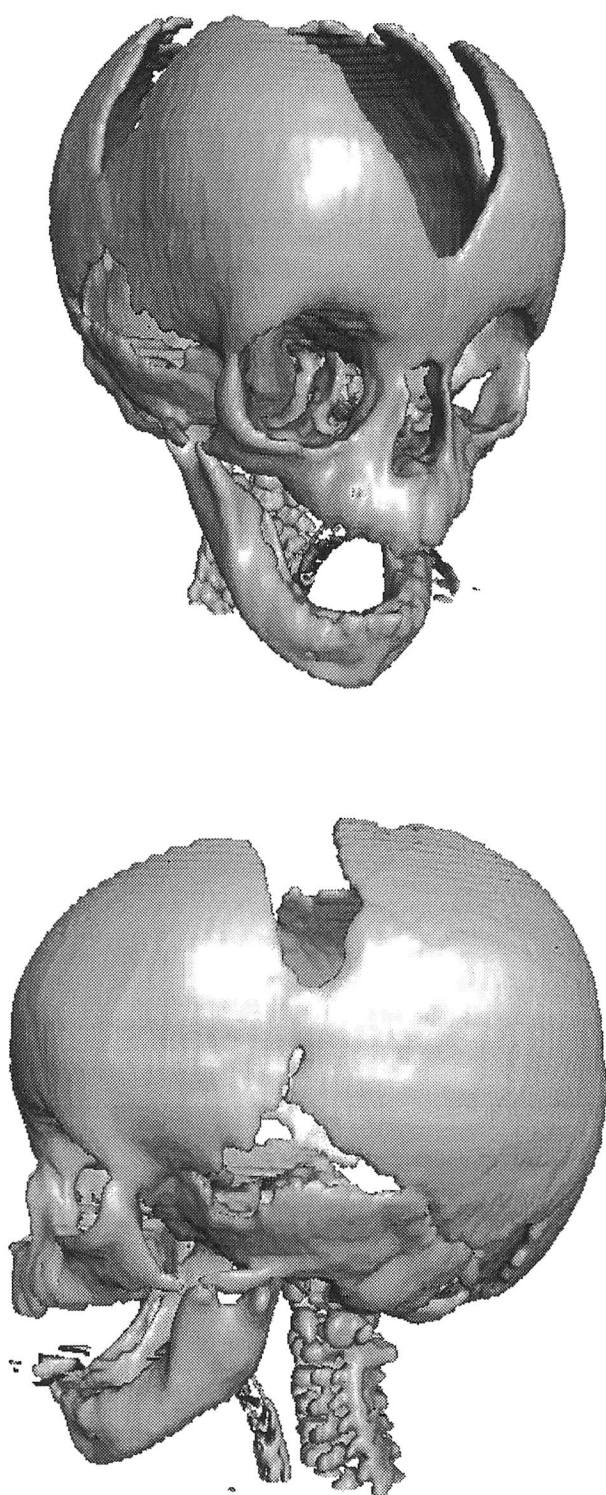


A) Vist som „wireframe“.

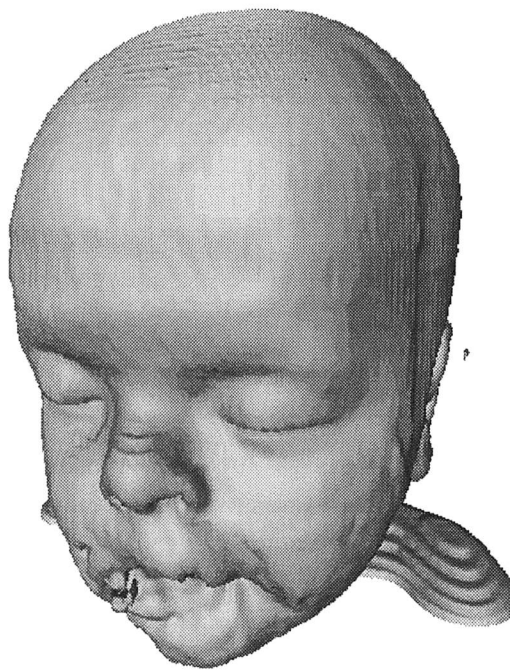


B) Tonet med Phong toning.

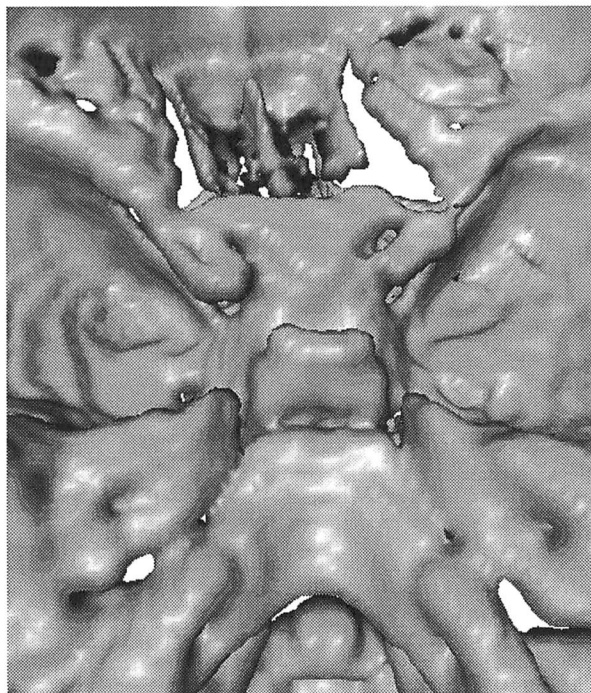
Figur 7.14: En model af en underkæbe bestående af cirka 12.000 facetter.



Figur 7.15: En model af et kranie bestående af cirka 160.000 facetter set fra to vinkler. Sammenlign eventuelt med figur 4.1 på side 42.



Figur 7.16: En model af et hoved bestående af cirka 120.000 facetter. Bløddelene fås ved at anvende en anden tærskelværdi end den der anvendes til modellerne af knoglen.



Figur 7.17: En model af et kranie bestående af cirka 200.000 facetter. Et nærbillede af kraniets bund (*basis cranii interna*) er vist.

Kapitel 8

Facetter og flader

8.1 Triangulering af facetter

8.2 Estimation af normaler

8.3 Forenkling af flade

Ved anvendelse af en tredimensional tegl-algoritme kan man få en flade opbygget af facetter. I dette kapitel vil nogle emner knyttet til disse flader blive berørt. Først beskrives hvordan en facet kan trianguleres således at den opbygges af plane trekanter. Derefter beskrives hvordan der kan knyttes normaler til skæringspunkter på fladen således at en bedre visualisering kan opnås. Endelig vil der blive bragt nogle ideer om hvordan en flade kan forenkles således at antallet af facetter reduceres.

8.1 Triangulering af facetter

Hidtil er en 2-facet blevet beskrevet som en flade udspændt af en lukket kurve, og i mangel af en mere præcis beskrivelse har fladen været beskrevet som en sæbehinde udspændt af kurven. Hvis man i praksis ønsker at præcisere facetens forløb, for eksempel inden den bliver anvendt som inddata til et visualiseringsprogram, så kan man *triangulere* den. På den måde bliver den (i reglen) ikke-plane flade eller polygon opbygget af plane trekanter.

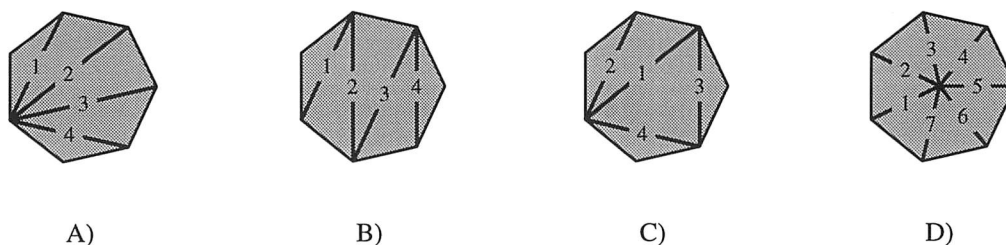
8.1.1 Triangulering af polygon. Når en facet trianguleres så foregår det ved at man forbinder visse af skæringspunkterne med rette linier således at der dannes trekanter. Facetten er egentlig en rumlig figur, men triangulering af denne er ækvivalent med trianguleringen af en polygon i planet, og det er dette problem der nu vil blive analyseret.

Man skal dog huske på at de facetter der er vanskeligst at triangulere tilfredsstillende netop er de meget „slyngede“ og „ikke-plane“ facetter, som for eksempel facetten for den svage ækvivalensklasse med signaturen 61 vist på figur 7.6. Når man vurderer anvendeligheden af en metode skal man derfor både overveje hvordan „næsten plane“ og „ikke-plane“ facetter håndteres.

8.1.2 Forskellige metoder. Udvælgelsen af skæringspunkter der skal forbindes kan foregå på flere måder. Figur 8.1 viser nogle eksempler på metoder anvendt på en syvkant. Selv om facetter med mere end seks skæringspunkter ikke optræder særligt hyppigt så er der her vist en syvkant fordi problemet for „små“ n -kanter næsten er trivielt.

De små tal der optræder på figurerne antyder i hvilken rækkefølge kanterne tilføjes til facetten, og dermed får man en idé om hvordan den underliggende algoritme virker.

8.1.3 Orientering. En facet har en orientering givet ved en omløbsretning. Derved defineres en „normalretning“ eller positiv side hvor de høje hjørner findes. Efter en



Figur 8.1: Eksempler på forskellige metoder til triangulering af en syvkantet facet.

triangulering er det vigtigt at hver af trekantene har samme orientering som den oprindelige facet. Metoderne beskrevet her opfylder alle dette krav.

8.1.4 Metode 1 som er vist på figur 8.1.A er måske den enklest tænkelige. Man udvælger et skæringspunkt, og herfra trækkes der kanter til alle øvrige skæringspunkter i facetten, på nær de to nabo-skæringspunkter hvortil der allerede er en kant. Mere formelt haves følgende...

8.1.5 Algoritme. Inddata er facetten $(\nu_1, \dots, \nu_n)$. Uddata er mængden af trekantede facetter F .

```

1  Funktion trianguler1( $\nu_1, \dots, \nu_n$ )
2       $F := \emptyset$ 
3      For  $i := 2, \dots, n - 1$  gør
4           $F := F \cup \{(\nu_1, \nu_i, \nu_{i+1})\}$ 
5      Returner  $F$ 

```

Algoritmen er nok enkel, men man kan med en vis ret hævde at triangulering er „asymmetrisk“, og de resulterende trekanter har en tendens til at blive meget „spidse“.

8.1.6 Metode 2 som er eksemplificeret på figur 8.1.B tjener til at afhjælpe nogle af problemerne med den første metode. Forenklet kan man sige at facetten trianguleres ved at „siksakke hen over den“ som vist på figuren. Denne metode kan realiseres ved følgende...

8.1.7 Algoritme. Inddata er facetten $(\nu_1, \dots, \nu_n)$. Uddata er mængden af trekantede facetter F . Undervejs benyttes to indeks, i og j , der itereres fra henholdsvis „starten“ og „slutningen“ af facetten mod hinanden indtil de „mødes“.

```

1  Funktion trianguler2( $\nu_1, \dots, \nu_n$ )
2       $F := \emptyset$ 
3       $i := 1$ 
4       $j := n - 1$ 
5      Så længe  $i + 1 < j$  gør
6           $F := F \cup \{(\nu_i, \nu_{i+1}, \nu_{j+1}), (\nu_j, \nu_{j+1}, \nu_{i+1})\}$ 
7           $i := i + 1$ 
8           $j := j - 1$ 
9      Hvis  $i < j$  så
10          $F := F \cup \{(\nu_i, \nu_{i+1}, \nu_{j+1})\}$ 
11     Returner  $F$ 

```

Ved hvert gennemløb af løkken i linierne 5–8 tilføjes to trekanter. Det tilfælde hvor et ulige antal trekanter skal tilføjes kontrolleres i linie 9.

8.1.8 Asymmetri i valg af startpunkt. De to metoder til triangulering hidtil beskrevet har det til fælles at kræver at der udvælges et „startpunkt“ på facetten, og to forskellige valg af „startpunkt“ vil i reglen føre til forskellige trianguleringer. Det kan derfor være ønskeligt at anvende en egenskab ved facetten der entydigt udvælger hvilken af flere mulige trianguleringer der anvendes.

8.1.9 Metode 3 sikrer dette ved rekursivt at dele facetten „midt over“ i to dele indtil den er trianguleret. Dette er eksemplificeret på figur 8.1.C hvor de små numre viser hvorledes hele facetten først deles i to mindre facetter. Den ene af disse deles så igen til to trekanter, mens den anden skal deles yderligere to gange før den er helt opdelt i trekanter.

8.1.10 Deling af facet. Men hvordan deles en facet „midt over“? En sekskant deles i to firkanter, og en syvkant, i en firkant og en femkant. Mere præcist har man at en n -kant som deles ligeligt, bliver delt i en k - og en l -kant med k og l givet ved

$$k = l = \frac{n+2}{2}, \quad n \text{ lige},$$

$$k = \frac{n+1}{2}, \quad l = \frac{n+3}{2}, \quad n \text{ ulige}.$$

Det er naturligvis osse interessant at finde ud af hvor mange mulige delinger der er for en given n -kant. Da hver deling involverer to hjørner kan man tælle antallet af delinger for hvert hjørne og dividere med to. Antallet m af mulige delinger bliver derfor

$$m = \frac{n}{2}, \quad n \text{ lige},$$

$$m = \frac{2n}{2} = n, \quad n \text{ ulige}$$

da en n -kant kan deles på to måder når n er ulige.

Disse observationer fører til følgende rekursive algoritme:

8.1.11 Algoritme. Inddata er facetten $(\nu_1, \dots, \nu_n)$. Uddata er mængden af trekantede facetter F . Hvilken deling af facetten der skal vælges afgøres af funktionen α der tildeler hver mulig deling en vægt. Delingen med minimal vægt vælges.

```

1  Funktion trianguler3( $\nu_1, \dots, \nu_n$ )
2      Hvis  $n = 3$  så
3          Returner  $\{(\nu_1, \nu_2, \nu_3)\}$ 
4      Ellers
5          Hvis  $n|2$  så
6               $m := n/2$ 
7              For  $i \in \{1, \dots, m\}$  gør
8                   $A_i := \alpha(i, i+m)$ 
9          Ellers
10              $m := (n-1)/2$ 
11             For  $i \in \{1, \dots, n\}$  gør
12                  $A_i := \alpha(i, (i+m)|_1^n)$ 
```

Bemærk hvordan rekursionen stoppes i linierne 2–3 hvis facetten der ønskes trianguleret er en trekant. Ellers sættes m til „forskydningen“ mellem første og andet punkt i en deling, og tabellen A_i som indeholder en vægt for hver mulig deling af facetten konstrueres. Der er to tilfælde, når n er lige og når n er ulige. Dernæst udvælges delingen med minimal vægt, og funktionen kaldes rekursivt med de to „halvdele“ F_1 og F_2 som facetter.

```

13       $A_j := \min_i A_i$ 
14       $F_1 := (\nu_j, \nu_{(j+1)|_1^n}, \dots, \nu_{(j+m)|_1^n})$ 
15       $F_2 := (\nu_{(j+m)|_1^n}, \nu_{(j+m+1)|_1^n}, \dots, \nu_{(j+m+n)|_1^n})$ 
16      Returner  $\text{trianguler}_3(F_1) \cup \text{trianguler}_3(F_2)$ 

```

8.1.12 Valg af deling. Der mangler en beskrivelse af funktionen α der blandt de m mulige delinger vælger den foretrukne. Det er endnu et åbent spørgsmål hvilke egenskaber ved facetten α skal afhænge af, men her er nogle ideer:

8.1.13 Geometri af facet. Man kan ønske at konstruere så „regulære“ trekanter så muligt. Det vil sige trekanter som har nogenlunde lige lange sider. Dette kan opnås i ved at vælge den korteste delelinie. Derved undgår man dele en meget „aflang“ facet i to lige så „aflange“ facetter.

Dette er en meget enkel metode. Et alternativ er følgende noget mere komplicerede metode:

8.1.14 „Knæk“ af facet. Når en „slynget“ facet deles vil det resultere i to mindre „slyngede“ og mere „plane“ facetter. Funktionen α kan afhænge af hvor voldsomt facetten „knækker“ langs delelinien. Dette kan kvantificeres på følgende måde: Hvert af de to skæringspunkter i delelinien betragtes for sig. Et plan gennem skæringspunktet med delelinien som normal defineres. De to kanter i facetten som udgår fra skæringspunktet projiceres ind på dette plan og vinklen mellem projektionerne beregnes. Hvis vinklen er nær 180° så er „knækket“ lille, mens en vinkel nær 0° svarer til et meget voldsomt „knæk“. Dette kan benyttes til at tildele skæringspunktet en vægt som for eksempel kan normeres til intervallet $[0; 1]$. Vægten af delelinien er middelværdien af de to skæringspunkters vægt, og man kan vælge at foretrække delelinier med det kraftigste „knæk“ eller delelinier med det svageste „knæk“ — det sidste valg er nok at foretrække.

Skal denne metode implementeres i praksis kræver det at en ortonormal basis for planet konstrueres ved anvendelse af Gram-Schmidt ortogonaliseringsmetoden. Det er derfor forholdsvis krævende at beregne alle vægte for en facet da ortonormaliseringen skal beregnes to gange for hver mulig deling.

8.1.15 Bemærkning. Man kan spørge sig selv om de tre metoder til triangulering der indtil nu er blevet beskrevet altid vil resultere i det samme antal trekanter — det virker umiddelbart sandsynligt at dette er tilfældet, men en nøjere analyse af problemet kan føre til en dybere forståelse for trianguleringsproblematikken. Der gælder følgende...

8.1.16 Sætning. En n -kant med $n \geq 3$ kan ved deling $n - 3$ gange opdeles i trekanter.

8.1.17 Bevis ved induktion. Antag at sætningen gælder for $3 \leq n < m$. Man kan dele en m -kant i en p - og en q -kant med $3 \leq p < m$ og $3 \leq q < m$. Der gælder at p - og q -kanten har to hjørner til fælles i m -kanten og derfor haves

$$p + q - 2 = m.$$

Ifølge antagelse skal man dele p -kanten $p-3$ gange og q -kanten $q-3$ gange for at dele dem i trekanter. Antallet af gange m -kanten skal deles for at opdele den i trekanter bliver derfor

$$p-3+q-3+1=(p+q-2)-3=m-3.$$

Da sætningen trivielt ses at gælde for $n=3$ bevises sætningen ved induktion over $n \geq 3$.

8.1.18 Introduktion af centrum. Metoden vist på figur 8.1.D er speciel, idet man i stedet for at trække linier mellem de eksisterende hjørner i n -kanten tilføjer et nyt punkt i centrum af n -kanten. Fra alle hjørner trækkes så linier således at der dannes n trekanter.

8.1.19 Barycenter. Ved at variere hvordan „centrum“ af en facet defineres fås forskellige resultater. Et naturligt forslag vil være barycentret af skæringspunkterne. Med facetten $(\nu_1, \dots, \nu_n)$ bliver barycentret

$$\nu_c = \frac{1}{n} \sum_{i=1}^n \nu_i.$$

8.1.20 Anvendelse af mål. Problemet med dette valg er at to skæringspunkter som ligger meget tæt, så at sige „indgår med dobbelt vægt“ i midlingen. For at komme uden om dette problem må man i stedet beregne et vægtet gennemsnit — men hvilken vægt skal man tildele de enkelte skæringspunkter?

Svaret er, at man slet ikke skal tildele skæringspunkterne nogen vægt. I stedet skal man midle over midtpunkterne i kanterne (ν_i, ν_{i+1}) hvor hvert midtpunkt indgår med vægt der svarer til kantens relative længde i forhold til den samlede længde af kanterne.

Mere præcist kan man opstille følgende definitioner:

8.1.21 Definition. Barycentret af 1-facetten (ν_1, ν_2) er

$$B(\nu_1, \nu_2) = \frac{\nu_1 + \nu_2}{2}.$$

8.1.22 Definition. Målet af 1-facetten (ν_1, ν_2) er

$$M(\nu_1, \nu_2) = |\nu_2 - \nu_1|.$$

8.1.23 Definition. Barycentret af 2-facetten $(\nu_1, \nu_2, \nu_3, \dots, \nu_n)$ sammensat af 1-facetterne $(\nu_1, \nu_2), (\nu_2, \nu_3), \dots, (\nu_n, \nu_1)$ er

$$B(\nu_1, \dots, \nu_n) = \frac{\sum_{i=1}^n M(\nu_i, \nu_{(i+1)|_1^n}) B(\nu_i, \nu_{(i+1)|_1^n})}{\sum_{i=1}^n M(\nu_i, \nu_{(i+1)|_1^n})}.$$

Senere vil disse definitioner blive generaliseret til højere dimensioner (definition 10.1.4).

8.1.24 Deling af tegl. Som tidligere nævnt er det de „slyngede“ facetter der er de mest problematiske at triangulere. Det skyldes egentlig at facetten, der jo essentielt er beskrevet ved en lukket kurve, ikke fortæller nok om fladens forløb. En mulighed er som sagt at introducere barycentret for på den måde at konkretisere facettens forløb, men man kan i stedet dele teglet som indeholder facetten midt over. Derved opstår to nye tegl som deler en side med fire hjørner. Disse fire hjørner findes ikke i det underliggende gitter, men ved interpolation kan værdier i disse hjørner estimeres. Det er sandsynligt at de to nye tegl indeholder facetter der er mindre „slyngede“ end den oprindelige facet. Delingsprocessen kan gentages idet „aflange“ tegl deles på tværs af deres længste akse, og når facetterne er „jævne“ nok kan de trianguleres ved en af de tidligere beskrevne metoder.

8.1.25 „Generic patch“. Denne metode skal anvendes mens der stadig er knyttet en sammenhæng mellem facetten og teglet som indeholder facetten, og den er først og fremmest anvendelig til tegl-algoritmer der konstruerer forenklede, „store“ facetter i tegl som er opbygget af flere mindre tegl som for eksempel „splitting-box“ algoritmen [Müller og Stark 1993]. Fortsættes delingsprocessen uendeligt får man den tidligere omtalte sæbehinde, men i praktiske tilfælde må processen naturligvis stoppes efter et endeligt antal skridt. En sådan flade kaldes på engelsk for en *generic patch* — en god dansk betegnelse for dette begreb er svær at komme på.

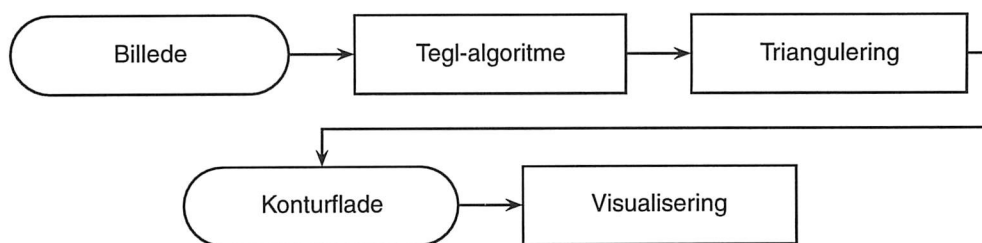
8.1.26 Valg af metode. Her er nu beskrevet fire forskellige metoder til triangulering, men hvilken metode skal man vælge i praksis? De to første metoder er enkle, men her der den anden nok at foretrække frem for den første. Disse to metoder er i reglen ikke så velegnede til at triangulere „slyngede“ facetter. Her vil den fjerde metode være at foretrække da barycentret for facetten er et godt valg når man forsøger at beskrive facettens forløb. Den tredje metode er langt den mest komplicerede og den er nok først relevant at anvende når det drejer sig om „store“ facetter — disse kan dannes i forbindelse med en reduktion af fladen ved fjernelse af skæringspunkter (se afsnit 8.3.14). Konklusionen er at så længe det kun drejer sig om at triangulere „små“ facetter konstrueret med den hidtil beskrevne tegl-algoritme, så er den anden metode nok at foretrække.

8.1.27 Lukning af kontur. Man kan betragte triangulering af facetterne som at „lægge sidste hånd på værket“ inden det visualiseres. I samme ånd kan lukning af konturen betragtes — strengt taget har dette intet med triangulering at gøre, men det vil alligevel blive omtalt her.

Hvis der er høje hjørner på randen af billedet så er konturen ikke lukket (forudsat at det der er „inden i“ billedet er et objekt defineret af høje hjørner). At konturen ikke er lukket er normalt ikke noget problem, men i forbindelse med for eksempel stereolitografi så kræves det at konturen udgør en lukket flade, det vil sige indeslutter et rumfang.

Billedets rand har seks „sider“, og man betragter alle dem der indeholder høje hjørner én af gangen. Hvis siden kun indeholder høje hjørner tilføjes hele siden som en „stor“ firkantet facet til konturfladen. Hvis siden indeholder både høje og lave hjørner vil den indeholde én eller flere konturkurver som er konturfladens skæring med randen. Konturkurven i en sådan side er enten åben eller lukket. Hvis den er åben, det vil sige hvis den starter og ender på en af de andre sider af billedet, så lukkes den ved at tilføje overgangen mellem den aktuelle side og de øvrige sider sådan at det høje område afgrænses af konturkurven. Derved får man én eller flere lukkede konturkurver som afgrænser de høje områder, og disse tilføjes som „store“ 2-facetter til konturfladen. Når dette er gjort for alle siderne med høje hjørner så vil konturfladen være lukket. Da de „store“ facetter er plane er det uvæsentligt hvilken algoritme der anvendes til en eventuel triangulering.

8.1.28 Anvendelse af triangulering. Figur 8.2 viser et diagram der illustrerer hvordan trianguleringen af facetter indarbejdes i den eksisterende algoritme. Det antages at visualiseringsdelen er et eksternt program eller er opbygget over et standardbibliotek. Ofte kræver det at fladen er trianguleret, og der kan tilmed være en hastighedsgevinst under visualiseringen hvis trekanterne præsenteres på en speciel måde (ofte i form af



Figur 8.2: Diagram der viser hvorledes triangulering af facetter passes ind i den eksisterende algoritme.

såkaldte *meshes*). Andre programmer og biblioteker accepterer ikke-plane polygoner som geometriske primitiver, og dermed osse facetter, og i dette tilfælde behøver man ikke at triangulere fladen inden den sendes til visualiseringsdelen.

8.1.29 Triangulering under visualiseringen. I stedet bliver fladen implicit trianguleret i forbindelse med visualiseringen. Dette har flere fordele: For det første slipper man for at implementere en trianguleringsdel, og for det andet er der mulighed for at visualiseringsdelen kan udnytte særlig grafisk hardware til trianguleringen således at den samlede procestid fra billedet indlæses til den færdige flade visualiseres bliver nedsat.

På den anden side kan det være at en trianguleret flade kan visualiseres hurtigere end en tilsvarende flade som ikke er trianguleret. Derved får man en hurtigere responstid i visualiseringsdelen på bekostning af en længere procestid i de forudgående trin. Men da visualiseringsdelen typisk er den interaktive del hvor brugeren utålmodigt sidder ved sin skærm og venter på uddata, så er dette at foretrække.

8.1.30 Triangulering på forhånd. Hvis en opslagstabel benyttes af tegl-algoritmen kan facetterne være trianguleret på forhånd. Med reference til figur 7.6 så vil den „fir-kantede“ facet for den svage ækvivalensklasse med signatur 3 blive trianguleret til to facetter, lissom facetten for den svage ækvivalensklasse med signatur 6 allerede består af to trekantede facetter. Denne teknik benyttes i „marching cubes“ algoritmen.

Alt i alt må det konkluderes at spørgsmålet om hvordan trianguleringen af fladen realiseres afhænger af den konkrete anvendelse og hvilken computer der er tilgængelig.

8.1.31 Triangulering tilføjer intet nyt. Det er for øvrigt værd at bemærke at de facetter som konstrueres af tegl-algoritmen i en vis forstand indeholder maksimal information om fladens forløb, i og med at facetterne kun beskrives i form af sæt af skæringspunkter. Lige så snart facetterne skal trianguleres må man anvende enten en form for tilfældighed i form af en vilkårlig deling af facetterne (de første to metoder), eller en form for kriterie om „pænhed“ af facetten (den tredje metode), eller en form for interpolation (den sidste metode). Triangulering af fladen tilføjer ikke nogen væsentlig ny information til fladen, men er udelukkende påkrævet for at afgøre den ubestemthed der er i facetternes forløb.

Ønsker man at manipulere den konstruerede flade inden den visualiseres, for eksempel med henblik på at undersøge fladens krumningsparametre, så kan dette ofte med fordel foregå før fladen trianguleres.

8.2 Estimation af normaler

8.2.1 Toning af flade. Den tredimensionale tegl-algoritme hidtil beskrevet fastlægger geometrien af en konturflade. Denne flade kan visualiseres, for eksempel ved at anvende *Gouraud* eller *Phong toning* [Foley et al. 1990]. Dette kræver at der i hvert af fladens skæringspunkter er fastlagt en normal som angiver den lokale orientering af fladen i skæringspunkt. Med kendskab til belysningskildernes og betragterens position samt fladens optiske egenskaber benyttes normalen til at beregne hvor meget og hvilken farve lys der reflekteres mod betragteren i skæringspunktet. Derved fastlægges en tone i hvert skæringspunkt.

Hver facet får en tone som varierer hen over facetten således at tonen i „hjørnerne“ af facetten svarer til tonen fastlagt af skæringspunktet i det hjørne. I praksis kræves det at facetten er trekantet, og derfor forudsættes det i det følgende at fladen er trianguleret som beskrevet i afsnit 8.1. Alle facetter er dermed trekantede (og plane).

8.2.2 Gouraud og Phong toning. Ved Gouraud toning foretages en lineær interpolation mellem tonerne i de tre hjørner af trekanten. Ved Phong toning er det i stedet normalerne i de tre hjørner der interpoleres. Mere præcist så skankonverteres hver trekant hvorved en række punkter på trekanten udvælges til at bidrage til de endelige billedpunkter. I hvert af disse punkter interpoleres normalen fra hjørnerne af trekanten og tonen beregnes ud fra denne interpolerede normal. Phong toning er derfor mere beregningskrævende da beregningen af tone skal gentages for hvert punkt på trekanten der bidrager til det endelige billede. Til gengæld bliver resultatet osse bedre set ud fra en „fotorealistisk“ betragtning.

8.2.3 Fastlæggelse af flade normaler er derfor vigtig i forbindelse med visualisering. Hvordan dette kan gøres er emnet for resten af afsnittet.

8.2.4 Flad toning. Normalen i hvert skæringspunkt kan fastlægges ud fra de omgivende faceters geometri. En trekantet facet er plan og fastlægger dermed en normal. Orienteringen af normalen fås som tidligere beskrevet fra orienteringen af facetten ved anvendelse af højrehåndsreglen. Benyttes denne normal vil der ikke være nogen tonevariation hen over hver facet og fladen vil fremstå med tydeligt markerede facetter. Da hver facet ser „flad“ ud kan man kalde denne form for toning for *flad toning* efter *flat shading* på engelsk.

8.2.5 Geometrisk normal. Hvert skæringspunkt er omgivet af en række facetter som hver fastlægger en normal. Disse normaler kan midles således at der til hvert skæringspunkt fastlægges en normal. Denne normal kan så benyttes som udgangspunkt for Gouraud eller Phong toning. Denne normal kan kaldes for den *geometriske normal*, og den er alene fastlagt ud fra fladens geometri og er uafhængig af det underliggende billede. Denne normal anvendes ofte til visualisering af flader hvor man kun er i besiddelse af fladens form og ikke på forhånd har givet nogle normaler.

8.2.6 Gradient normal. Som omtalt i afsnit 4.3.14 kan man bestemme normalen til en ækvipotentiaflade ved at bestemme gradienten af det felt som definerer fladen i det punkt hvor normalen ønskes bestemt. Normaliseres gradienten fås den ønskede normal. Nedenfor vil det blive beskrevet hvordan man bestemmer gradienten i hvert billedpunkt i 3-billedet. Et skæringspunkt ligger i et 1-tegl udspændt af to billedpunkter (osse

kaldet hjørner), og en gradient kan derfor bestemmes ved lineær interpolation mellem de to billedpunkter. Denne type normal vil kan kaldes for *gradient normalen*.

8.2.7 Interpolation af gradient. Lad x_1 og x_2 være de to hjørner. Med reference til udtrykket (6.1) og bemærkning 6.2.4 sættes

$$\alpha = \frac{T - V(x_1)}{V(x_2) - V(x_1)}.$$

Hvis gradienten i hjørnet x betegnes med $D(x)$ så kan gradienten i skæringspunktet ν i det kritiske 1-tegl med x_1 og x_2 som hjørner skrives som

$$D(\nu) = (1 - \alpha)D(x_1) + \alpha D(x_2).$$

Normalen fås som før nævnt ved normalisering af gradienten. I det tilfælde hvor gradienten er nul kan man naturligvis ikke normalisere gradienten, men normalt vil gradienten ikke være nul i nærheden af et kritisk 1-tegl.

8.2.8 Beregning af gradient. Det væsentlige problem er derfor hvordan man bestemmer gradienten i hvert billedpunkt. Her kan man benytte resultater fra traditionel todimensionale billedanalyse hvor gradienten ofte har været anvendt til, for eksempel, simpel kantdetektion. Størrelsen af den lokale gradient giver information om *styrken* af en eventuel kant og gradientens retning giver en idé om kantens orientering. Der findes derfor adskillige filtre som giver estimater for den lokale gradient i et todimensional billede.

Disse resultater kan naturligvis generaliseres til tre eller flere dimensioner, men man skal passe på inden man ukritisk anvender metoder fra traditionel billedanalyse. Det skyldes at man har brug for den *matematiske* gradient og ikke bare en passende parameter der giver information om kanter i billedet. Og den matematiske gradient er ikke blot afhængig af værdifunktionen på billedet, men osse af koordinatfunktionen, og hvis denne funktion ikke er identiteten kan de velkendte kerner til beregning af gradienten ikke direkte bruges. I stedet må man introducere visse modifikationer.

Indledningsvis vil det blive beskrevet hvordan man estimerer grader i en dimension. På det grundlag kan mere avancerede metoder opstilles.

8.2.9 Estimation af differentialkvotient. Følgende er velkendt fra numerisk analyse: Lad $f: \mathbb{R} \rightarrow \mathbb{R}$ være en C^∞ afbildning. Den har da følgende to Taylor udviklinger i en omegn omkring x

$$f(x \pm h) = f(x) \pm f'(x)h + \frac{f''(x)}{2}h^2 + o(h^2). \quad (8.1)$$

Subtraheres de to udviklinger fås

$$f(x + h) - f(x - h) = 2f'(x)h + o(h^2)$$

som kan omskrives til

$$f'(x) = \frac{f(x + h) - f(x - h)}{2h} + o(h).$$

Man kan dermed opskrive en tilnærmelse til $f'(x)$ som kaldes for *central differens* og som har formen

$$D_c(f, h)(x) = \frac{f(x+h) - f(x-h)}{2h}. \quad (8.2)$$

Udtrykket (8.1) kan i stedet benyttes til at opskrive

$$f(x+h) - f(x) = f'(x)h + o(h)$$

hvor andengradsledet er indeholdt i restledet $o(h)$. Ved omordning af ledene fås

$$f'(x) = \frac{f(x+h) - f(x)}{h} + o(1)$$

som benyttes til at definere differentiation ved *forlæns differens*

$$D_f(f, h)(x) = \frac{f(x+h) - f(x)}{h}.$$

Tilsvarende fås *baglæns differens*

$$D_b(f, h)(x) = \frac{f(x) - f(x-h)}{h}$$

Ved at betragte restledene i de involverede udtryk ses det forlæns og baglæns differens er en orden h „dårligere“ end central differens.

8.2.10 Estimation af gradient. Disse tre metoder kan direkte anvendes til estimation af gradienten i et billede $P = ((X, \mathcal{X}), V, C)$ såfremt koordinatfunktionen C er regulær i alle koordinater.

8.2.11 Regulær koordinatfunktion med konstanter c_k for $k = 1, \dots, n$. Hvis dimensionen i billedet er n og m -topologien anvendes ($m = 0$ eller $m = n$) så er gradienten givet ved

$$D = (D_1, \dots, D_n): {}^{\circ m}(X) \rightarrow \mathbb{R}^n$$

hvor hver enkelt komponent af gradienten fås fra den centrale differens (8.2) med $h = c_k$ for $k = 1, \dots, n$. Dette giver

$$D_k(x_{j_1 \dots j_n}) = \frac{V(x_{j_1 \dots (j_k+1) \dots j_n}) - V(x_{j_1 \dots (j_k-1) \dots j_n})}{2c_k}, \quad k = 1, \dots, n. \quad (8.3)$$

8.2.12 Ikke-regulær koordinatfunktion. Hvis koordinatfunktionen ikke er regulær i alle koordinater så må man være lidt mere snedig. Lad igen en C^∞ -funktion $f: \mathbb{R} \rightarrow \mathbb{R}$ være givet. Man har følgende Taylor udvikling

$$f(x \pm h_\pm) = f(x) \pm f'(x)h_\pm + \frac{f''(x)}{2}h_\pm^2 + o(h_\pm^2)$$

hvor der ikke nødvendigvis gælder at $h_+ = h_-$. Ved at betragte differensen

$$\frac{1}{h_+}f(x+h_+) - \frac{1}{h_-}f(x-h_-) = \left(\frac{1}{h_+} - \frac{1}{h_-}\right)f(x) + 2f'(x) + o(1)$$

fås følgende udtryk for differentialkvotienten

$$f'(x) = \frac{1}{2h_+}(f(x+h_+) - f(x)) + \frac{1}{2h_-}(f(x) - f(x-h_-)) + o(1).$$

Dette fører til følgende modificerede udtryk for den centrale differens

$$D_{c'}(f, h_+, h_-)(x) = \frac{1}{2}(D_f(f, h_+)(x) + D_b(f, h_-)(x)).$$

Dette generaliseres nemt til n dimensioner ved at erstatte udtrykket (8.3) med

$$D_k(x_{j_1 \dots j_n}) = \frac{1}{2} \left(\frac{V(x_{j_1 \dots (j_k+1) \dots j_n}) - V(x_{j_1 \dots j_k \dots j_n})}{C_k(x_{j_1 \dots (j_k+1) \dots j_n}) - C_k(x_{j_1 \dots j_k \dots j_n})} + \frac{V(x_{j_1 \dots j_k \dots j_n}) - V(x_{j_1 \dots (j_k-1) \dots j_n})}{C_k(x_{j_1 \dots j_k \dots j_n}) - C_k(x_{j_1 \dots (j_k-1) \dots j_n})} \right). \quad (8.4)$$

Bemærk at fejlen her er af størrelsesordenen $o(1)$ i stedet for $o(h)$ som det var tilfældet i (8.3), og dermed er estimatet en størrelsesorden „dårligere“. Bemærk osse at hvis koordinatfunktionen er regulær så reduceres udtrykket (8.4) til (8.3) som forventet.

8.2.13 Anvendelse af foldning Bedre estimater for gradienten i et n -billede kan fås ved at folde billedet med en passende kerne. Udtrykket (8.3) kan i princippet omskrives til en foldning med en $1 \times 1 \times \dots \times 3 \times \dots \times 1$ kerne hvor treetallet optræder på den k 'te plads, og hvor dimensionen af kernen naturligvis er n .

8.2.14 Sobel kernen i tre dimensioner. Mere interessante gradient kerner fås ved at generalisere Prewitt eller Sobel kernerne kendt fra to dimensioner [Niblack 1985]. For eksempel er en tredimensionel Sobel kerne med sidelængde tre hensigtsmæssig at anvende i den tredimensionale tegl-algoritme. Kernen til estimation af gradientens x -komponent får udseendet [Analyze 1993]

$$\frac{1}{52} \left[\begin{bmatrix} -2 & 0 & 2 \\ -3 & 0 & 3 \\ -2 & 0 & 2 \end{bmatrix}, \begin{bmatrix} -3 & 0 & 3 \\ -6 & 0 & 6 \\ -3 & 0 & 3 \end{bmatrix}, \begin{bmatrix} -2 & 0 & 2 \\ -3 & 0 & 3 \\ -2 & 0 & 2 \end{bmatrix} \right].$$

Tilsvarende fås for y -komponenten af gradienten udtrykket

$$\frac{1}{52} \left[\begin{bmatrix} 2 & 3 & 2 \\ 0 & 0 & 0 \\ -2 & -3 & -2 \end{bmatrix}, \begin{bmatrix} 3 & 6 & 3 \\ 0 & 0 & 0 \\ -3 & -6 & -3 \end{bmatrix}, \begin{bmatrix} 2 & 3 & 2 \\ 0 & 0 & 0 \\ -2 & -3 & -2 \end{bmatrix} \right]$$

og for z -komponenten af gradienten fås

$$\frac{1}{52} \left[\begin{bmatrix} 2 & 3 & 2 \\ 3 & 6 & 3 \\ 2 & 3 & 2 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \begin{bmatrix} -2 & -3 & -2 \\ -3 & -6 & -3 \\ -2 & -3 & -2 \end{bmatrix} \right].$$

En kerne til estimation af gradienten kan kun anvendes direkte hvis koordinatfunktionen er identiteten, men ved at anvende ideerne skitseret ovenfor kan koordinatfunktionen inddrages i foldningen hvorved et gyldigt udtryk fås.

8.2.15 Kombination af foldning med ikke-identisk koordinatfunktion. Generelt vil en kerne til estimation af D_k opfylde

$$a_{l_1 \dots l_k \dots l_n} |_{l_k=0} = 0.$$

Foldningen givet ved definition 5.4.1 kan da omskrives til

$$\begin{aligned} V'(x_{j_1 \dots j_n}) &= \sum_{l_1=-d_1}^{d_1} \cdots \sum_{l_k=1}^{d_k} \cdots \sum_{l_n=-d_n}^{d_n} a_{l_1 \dots l_n} V(x_{(j_1+l_1) \dots (j_n+l_n)}) \\ &+ \sum_{l_1=-d_1}^{d_1} \cdots \sum_{l_k=-d_k}^{-1} \cdots \sum_{l_n=-d_n}^{d_n} a_{l_1 \dots l_n} V(x_{(j_1+l_1) \dots (j_n+l_n)}). \end{aligned} \quad (8.5)$$

Dette udtryk kan opfattes som en generalisation af udtrykket (8.2) for central differens med $h = 1$. Denne observation benyttes til at generalisere udtrykkene (8.3) og (8.4). Er koordinatfunktionen regulær i den k 'te koordinat fås

$$\begin{aligned} D_k(x_{j_1 \dots j_n}) &= \frac{1}{c_k} \left(\sum_{l_1=-d_1}^{d_1} \cdots \sum_{l_k=1}^{d_k} \cdots \sum_{l_n=-d_n}^{d_n} a_{l_1 \dots l_n} V(x_{(j_1+l_1) \dots (j_n+l_n)}) \right. \\ &\quad \left. + \sum_{l_1=-d_1}^{d_1} \cdots \sum_{l_k=-d_k}^{-1} \cdots \sum_{l_n=-d_n}^{d_n} a_{l_1 \dots l_n} V(x_{(j_1+l_1) \dots (j_n+l_n)}) \right). \end{aligned}$$

Hvis koordinatfunktionen ikke er regulær i den k 'te koordinat fås i stedet det mere generelle udtryk

$$\begin{aligned} D_k(x_{j_1 \dots j_n}) &= \frac{\sum_{l_1=-d_1}^{d_1} \cdots \sum_{l_k=1}^{d_k} \cdots \sum_{l_n=-d_n}^{d_n} a_{l_1 \dots l_n} V(x_{(j_1+l_1) \dots (j_n+l_n)})}{C_k(x_{j_1 \dots (j_k+1) \dots j_n}) - C_k(x_{j_1 \dots j_k \dots j_n})} \\ &+ \frac{\sum_{l_1=-d_1}^{d_1} \cdots \sum_{l_k=-d_k}^{-1} \cdots \sum_{l_n=-d_n}^{d_n} a_{l_1 \dots l_n} V(x_{(j_1+l_1) \dots (j_n+l_n)})}{C_k(x_{j_1 \dots j_k \dots j_n}) - C_k(x_{j_1 \dots (j_k-1) \dots j_n})}. \end{aligned}$$

8.2.16 Kontekstafhængige metoder. Billeder, specielt medicinske billeder, vil næsten altid indeholde diskontinuiteter, det vil sige kraftige spring i værdifunktionen. Ikke overraskende er estimeret af gradienten ofte dårligt omkring disse diskontinuiteter. Uheldigvis vil der ofte være diskontinuiteter i nærheden af den kontur man ønsker at estimere normaler for, da de mest interessante konturer netop repræsenterer overgange mellem, for eksempel, to typer væv.

Yagel et al. [1991] beskriver en metode som under foldningen kun inddrager billedpunkter fra samme *kontekst* i summen (8.5). To billedpunkter som er adskilt af en C^0 eller en C^1 diskontinuitet hører til hver sin kontekst. Metoden anvendes i to dimensioner, tilsyneladende med stor succes, og de beskriver osse hvorledes metoden nemt generaliseres til for eksempel tre dimensioner. Det største problem er i praksis at afgøre om to punkter hører til samme kontekst. Benyttes et „dårligt“ prædikat til dette vil estimeret for gradienten osse blive „dårligt“, men hvordan kvantificerer man hvor „dårlig“ en gradient er? Dette er det første problem der skal løses.

8.2.17 Estimation af normaler i praksis foregår som tidligere nævnt ved at gradienten i hver „ende“ af den kritiske kant beregnes. Da to kritiske kanter som deler et fælles hjørne anvender gradienten i dette hjørne til at beregne normalerne hørende til de respektive skæringspunkter i kanterne så er det muligt at optimere beregningen ved at genanvende tidligere beregnede gradienter.

Dette kan man gøres ved at anvende den hash tabel teknik som er beskrevet i afsnit 7.8.2. Som entydigt indeks for gradienten kan anvendes

$$H = (j_3 m_2 + j_2) m_1 + j_1$$

hvor dimensionen af billedet er $m_1 \times m_2 \times m_3$ og hvor hjørnet for gradienten er 0-cellen $x_{j_1 j_2 j_3}$.

8.2.18 Forudberegning af gradient. Man kan osse forestille sig en strategi hvor man på forhånd beregnede gradienten i hvert hjørne i billedet, men det er som regel en dårlig idé da antallet af gradienter der skal anvendes næsten altid er meget mindre end det totale antal hjørner i billedet. Selv om beregningen kan optimeres ved at anvende *separable* kerner vil det nødvendige lagerforbrug være stort da det ikke er nok at gemme for eksempel længden af gradienten normaliseret til 8 bit — alle tre komponenter af gradienten repræsenteret som flydende-komma tal er nødvendige. I andre sammenhænge kan gradienten dog være nyttig, for eksempel ved kantdetektion i billedet og her kan en eventuel genanvendelse komme på tale.

8.2.19 Orientering af normal. En gradient normal vil „pege mod“ de høje hjørner, og dette er konsistent med den „normalretning“ der fastlægges af facetten som beskrevet i afsnit 7.2.4. I de fleste tilfælde vil det dog være sådan at de høje hjørner er „inden i“ fladen, og det vil derfor være naturligt at erstatte den beregnede normal med den modsatte vektor. I den forbindelse vil det osse være mest korrekt at vende orienteringen af facetterne. Derved bliver det under visualiseringen muligt at afgøre om en facet vender „bort“ fra synsretningen ved anvendelse af *backfacing*. Hvis fladen er lukket og betragteren ikke befinder sig inden for fladen vil disse facetter ikke bidrage til det endelige billede, og de behøver derfor ikke at blive skankonverteret.

8.3 Forenkling af flade

8.3.1 Komplicerede flader. En tredimensional tegl-algoritme anvendt på medicinske data vil ofte resultere i en meget kompliceret flade, forstået på den måde at der benyttes et stort antal facetter til beskrivelsen af fladen. Det skyldes at de enkelte facetter er af omtrent samme størrelse som teglene, og disse har samme størrelse som de enkelte billedpunkter i billedet. For eksempel så består fladen illustreret på figur 7.15 (side 98) af cirka 160.000 facetter.

8.3.2 Forenkling af flade. Når en sådan flade visualiseres kan man ved inspektion ofte finde større områder på fladen som er ganske jævne, men alligevel er disse områder opbygget af et kolossalt antal facetter. Hvis man er i stand til at „udskifte“ et stort antal „små“ facetter med et færre antal større i det jævne område uden at fladen ændrer sig nævneværdigt, så kan man spare resurser dels i forbindelse med lagringen af fladen, dels i forbindelse med visualiseringen. Da et af de ultimative mål i denne forbindelse er at visualisere fladen i sand tid så er dette en af de mulige veje mod dette mål.

8.3.3 Hierarki af detaljer. Man kan abstrakt tænke på de mulige flader som et hierarki af forskellige skalaer eller detaljerigdomme. I bunden af hierarkiet er en meget detaljeret flade opbygget af et stort antal „små“ geometriske primitiver — når tegl-algoritmer anvendes er primitiverne facetter. I toppen af hierarkiet har man en meget grov model som kun benytter et lille antal „store“ geometriske primitiver. En *forenklet* flade er en flade som benytter den *fornødne* detaljerigdom i hver del af fladen.

8.3.4 Metoder til forenkling. Som beskrevet i afsnit 4.5.12–4.5.16 kan metoder til forenkling af en flade opdeles i to kategorier:

1. *Oppefra og ned*, hvor en algoritme er i stand til at variere størrelsen af de anvendte geometriske primitiver afhængigt af variationen af den flade som ønskes konstrueret. Det skal være muligt at anvende „store“ primitiver i områder med lidt variation, mens „mindre“ primitiver skal anvendes i mere komplicerede områder af fladen.
2. *Nedefra og op*, hvor en detaljeret flade konstrueres hvorefter den forenkles ved sammensmeltning af geometriske primitiver.

Den første type er i modsætning til den anden type „elegant“, forstået på den måde at man i hierarkiet af detaljerigdomme kun går så langt ned som det er ønsket. Med den anden metode slipper man ikke for at konstruere den „alt for detaljerede“ flade.

8.3.5 Forenkling „oppe fra og ned“. Hovedparten af de algoritmer jeg har kendskab til som arbejder „oppefra og ned“ kræver at det gitter som værdifunktionen er registreret på kan underinddeles vilkårligt [Bloomenthal 1988, Hall og Warren 1990, Schmidt 1993]. Dette er tilfældet ved modellering af syntetiske objekter.

8.3.6 Anvendelse på empiriske data. På grund af kravet om muligheden for en vilkårlig underinddeling af gitteret er den type algoritmer ikke umiddelbart anvendelige til behandling af empiriske data som foreligger på et fast gitter, men man kan naturligvis konstruere en analytisk funktion ud fra et eksisterende tredimensionalt billede, ved for eksempel at benytte en form for interpolation i hjørner mellem gitterpunkterne. Det ville derfor være ønskeligt at undersøge anvendeligheden af algoritmerne beskrevet af Bloomenthal [1988], Hall og Warren [1990] og Schmidt [1993] på empiriske data, for eksempel medicinske data.

„Splitting-box“ algoritmen beskrevet af Müller og Stark [1993] som kort er beskrevet i afsnit 4.5.17 udemærker sig i modsætning til de ovenfor nævnte algoritmer ved at den er målrettet mod anvendelse på empiriske data.

8.3.7 „Nedefra og op“ algoritmer kan direkte kobles på de tegl-algoritmer der allerede er beskrevet. Når tegl-algoritmen har konstrueret hele fladen analyseres den af en forenklingsalgoritme inden den sendes til visualiseringsdelen af programmet. Man kan forestille sig forskellige strategier til forenkling „nedefra og op“: Mindre geometriske primitiver kan smeltes sammen til større, for eksempel ved anvendelse af *prismetræk* [Ponce og Faugeras 1987]. I stedet kan et enkelt geometrisk primitiv fjernes og det resulterende hul kan så „lappes“. I forbindelse med tegl-algoritmer anvendes som geometrisk primitiv facetter og skæringspunkter.



A) Fladen før den centrale facet fjernes.

B) Facetterne efter reduktionen. Et nyt punkt introduceres i det oprindelige center af den fjernede facet.

Figur 8.3: Udsnit af en flade opbygget af facetter. Den centrale trekant markeret med gråt fjernes.

8.3.8 Fjernelse af facet. Når en facet fjernes fra en flade opstår der et hul. Dette hul kan lukkes ved at introducere et nyt skæringspunkt midt i hullet. Hullet trækkes sammen om dette nye punkt indtil det er lukket, og derved ændres de omgivende facetter. Et eksempel på dette er vist på figur 8.3. De facetter der deler en kant med den facet der fjernes „mister“ en kant efter reduktionen, og det betyder at trekanter bliver til tokanter som derved forsvinder fra fladen.

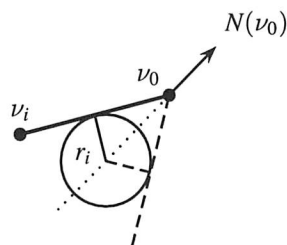
Man skal bemærke at en facet kan ligge på randen af billedet. Mere præcist så vil en facet der har en kant i et kritisk 2-tegl på randen af billedet ikke kunne fjernes uden særlig omtanke. Man kan helt undlade at fjerne facetter af denne type, men man kan osse kræve at det nye punkt, der normalt placeres i centret af den fjernede facet, i stedet skal placeres i randen af billedet. Selv om facetten ligger i et „hjørne“ af billedet kan dette lade sig gøre da der blot kommer flere bånd på placeringen af det nye punkt.

8.3.9 Estimation af midtpunkt. Punktet „midt i“ hullet kan estimeres ved at beregne barycentret af den facet der fjernes. Som beskrevet i afsnit 8.1.19–8.1.20 kan dette gøres på to måder — en simpel og en avanceret.

8.3.10 Udvalgelse af facetter. Det virkelige problem er at afgøre hvilke facetter der skal fjernes. „Små“ facetter er oplagte kandidater, men i områder af fladen hvor der sker store forandringer kan facetterne godt være „små“, men det er alligevel ikke særligt hensigtsmæssigt at fjerne disse facetter. Bedre er det at estimere en lokal „ujævnhedsparameter“ eller krumning.

8.3.11 Forslag til parameter. For en „matematisk“ flade (det vil sige *regulær* flade) kan man definere en størrelse som kaldes den *maksimale principale krumning* k i hvert punkt på fladen. At bestemme denne størrelse i et givet punkt svarer intuitivt til at bestemme radius r af en kugle som har fælles tangetplan med fladen i det givne punkt. Kuglen skal være den største som fladen lokalt kan „krumme omkring“. Der gælder da at $k = 1/r$.

8.3.12 Estimation af krumning. Turk [1992] skitserer en metode hvorved k kan estimeres i hvert skæringspunkt. Skæringspunktet ν_0 er knyttet til m nabo-skæringspunkter



Figur 8.4: Illustration af hvorledes den maksimale principale krumning bestemmes. Se teksten for en nærmere forklaring.

$(\nu_1, \dots, \nu_m)$ via m 1-facetter. For hvert af nabo-skæringspunkterne ν_i estimeres en krumningsradius som følger (se figur 8.4): Forbindelseslinien $\nu_i - \nu_0$ mellem ν_0 og ν_i spejles i linien fastlagt af normalvektoren $N(\nu_0)$ i ν_0 . Antag at $N(\nu_0)$ ikke er ortogonal på $\nu_i - \nu_0$. Derved udspænder $\nu_i - \nu_0$ og dets spejlbillede et plan. Punktet $(\nu_i - \nu_0)/2 + \nu_0$, halvvejs mellem ν_0 og ν_i , og det tilsvarende punkt i spejlbilledet fastlægger den cirkel i planet udspændt af de to linier der har de to linier som tangent i de to punkter. Radius r_i af denne cirkel benyttes som estimat for krumningen $k_i = 1/r_i$. I det tilfælde hvor $N(\nu_0)$ er ortogonal på $\nu_i - \nu_0$ sættes $k_i = 0$.

Beregningen af k_i kan realiseres ved først at beregne

$$\alpha = \arccos \frac{|N(\nu_0) \cdot (\nu_i - \nu_0)|}{|\nu_i - \nu_0|}.$$

Hvis $\alpha = \pi/2$ så sættes $k_i = 0$. Ellers anvendes

$$k_i = \frac{2}{|\nu_i - \nu_0| \tan \alpha}.$$

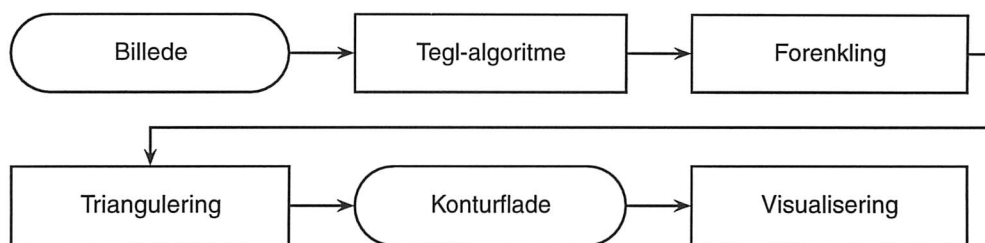
Den maksimale principale krumning i ν_0 sættes til

$$k(\nu_0) = \max_i k_i.$$

Ifølge Turk [1992] er dette estimat til tider „ulineært“, og han foreslår derfor at krumningen i hvert skæringspunkt midles med krumningen af nabo-skæringspunkterne. Da „krumningen“ af en facet er den interessante størrelse i forbindelse med forenkling ved fjernelse af facetter, er det derfor naturligt at sætte denne til middelværdien af krumningen af de skæringspunkter der indgår i facetten. For facetten $\phi = (\nu_1, \dots, \nu_n)$ defineres krumningen af facetten som

$$k(\phi) = \frac{1}{n} \sum_{i=1}^n k(\nu_i).$$

Hermed får man et mål for hvor voldsomt fladen varierer hen over facetten — en stor værdi af $k(\phi)$ betyder en stor variation.



Figur 8.5: Diagram der viser hvorledes forenkling af fladen passes ind i den eksisterende algoritme.

8.3.13 Forslag til reduktionsalgoritme. Givet en flade opbygget af facetter, kan man konstruere en sorteret liste af facetterne ordnet efter størrelsen af krumningen k . Så fjerner man den facet med mindst k . Dette forandrer facettens naboer hvorved de får en ny værdi af k . Nogle facetter forsvinder tilmed, nemlig de trekanten der delte en kant med den fjernede facet. Dette giver anledning til en forandring af den sorterede liste af facetter, men da listen er sorteret kan man ved for eksempel binær søgning forholdsvis hurtigt opdatere denne. Man kan blive ved med at fjerne facetter fra starten af listen indtil en given tærskelværdi for k nås.

En given implementering skal være baseret på datastrukturer der er i stand til at håndtere den sorterede liste af facetter, lissom information om en facets naboer skal være tilgængelig så fjernelsen af en facet kan gøres effektivt. Benyttes en af de hierarkiske repræsentationer af fladen som er beskrevet i afsnit 7.7 så er det forholdsvis enkelt at bestemme en facets naboer.

Jeg har arbejdet med en prototype af den skitserede algoritme og resultaterne ser lovende ud. Desværre har det ikke inden afslutningen af dette speciale være muligt at bearbejde nogle „ordentlige“ data (det vil sige data som er mere komplicerede end små kugleflader og lignende). Det skyldes at prototypen er implementeret i Perl som er et godt programmeringssprog til afestning af prototyper — desværre så bliver prototyperne meget langsomme og derfor kan kun meget enkle flader håndteres i praksis.

8.3.14 Fjernelse af skæringspunkt. I stedet for at fjerne en facet kan man fjerne et skæringspunkt. En sådan algoritme er beskrevet af Schroeder et al. [1992] og den vil nemt kunne tilpasses flader som er opbygget af facetter i stedet for blot trekanten. Det „hul“ der opstår i fladen når skæringspunktet fjernes skal lukkes og dette foregår nemmest ved en triangulering af „hullet“. I den forbindelse kan nogle af de mere avancerede metoder til triangulering beskrevet i afsnit 8.1 måske anvendes med fordel.

8.3.15 Anvendelse af forenkling. Figur 8.5 indeholder et diagram der angiver hvordan forenkling af fladen indarbejdes i den eksisterende algoritme. Det er klart at forenklingen skal foregå før konturen trianguleres.

Kapitel 9

Mere om tegl-algoritmer

- 9.1 Trappeeffekten
- 9.2 Anvendelse af n -topologien
- 9.3 Triangulering af tegl
- 9.4 Marching cubes
- 9.5 Noter og litteratur

Dette kapitel omfatter de emner som ikke rigtig hører til de øvrige kapitler og er derfor en blandet landhandel.

Først omtales den såkaldte *trappeeffekt*. Dette er en form for *aliasing* der kan opstå når en tegl-algoritme anvendes til konstruktion af en kontur.

Derefter undersøges hvordan man kan konstruere konturer i billeder udstyret med n -topologien. Dette fører til en interessant modifikation af den tredimensionale tegl-algoritme.

I afsnit 9.3 omtales hvordan tegl kan trianguleres således at man i stedet kan anvende en tegl-algoritme med simpliciale tegl. Specielt sammenhængen med problemerne omkring flertydighed er interessante.

I afsnit 9.4 beskrives forskellen mellem den her beskrevne tegl-algoritme og „marching cubes“ algoritmen. Hermed bringes det definitive svar på hvorfor „marching cubes“ somme tider konstruerer flader med fejl. Til slut er der et afsnit med henvisninger til litteraturen hvor dette problem og andre emner om tegl-algoritmer er diskuteret.

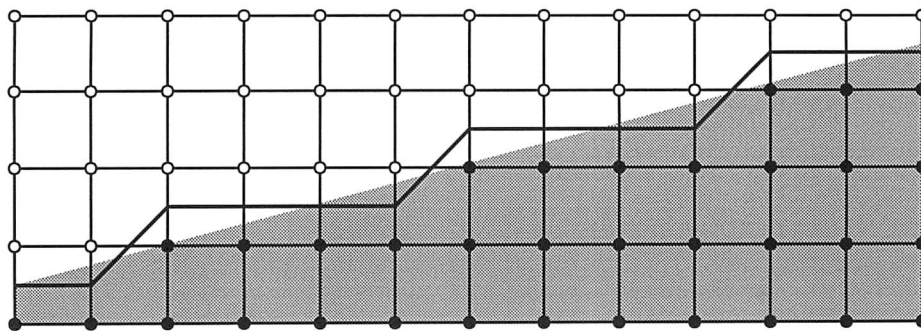
9.1 Trappeeffekten

Denne effekt er et uheldigt resultat af den diskrete registrering af måleværdier og den binære klassifikation af hjørner i høj og lav. Effekten optræder både i to og i tre dimensioner, men den er nemmere at forstå hvis den forklares i to dimensioner.

9.1.1 Eksempel. Betragt et sort-hvidt billede som består af en hvid og en sort halvdel. Gitteret som billedet registreres på er placeret på skrå i forhold til skillelinien mellem sort og hvidt som vist på figur 9.1. Algoritmens opgave er at bestemme denne skillelinie.

Når vinklen mellem en af gitterets akser og skillelinien er passende (for eksempel 15°) så vil den konstruerede kurve ikke være en ret linie, men en tydelig trappeformet linie.

Dette skyldes at de fleste kritiske tegl er i samme stærke ækvivalensklasse, i dette tilfælde 3, og konturen i hvert af disse tegl er den samme da billedet er binært. Hvis billedværdierne for sort og hvid er 0 og 1 og man benytter $0 < T < 1$ som tærskelværdi samt udtrykket (6.1), så vil alle skæringspunkter i alle kritiske kanter være i afstanden



Figur 9.1: Et binært billede registreret på et gitter. Algoritmen er anvendt med tærskelværdi $T = 1/2$ hvorved den tydeligt trappeformede kontur fremkommer.

T fra det lave hjørne (forudsat at koordinatfunktionen er identiteten). På figur 9.1 er $T = 0,5$ anvendt som eksempel.

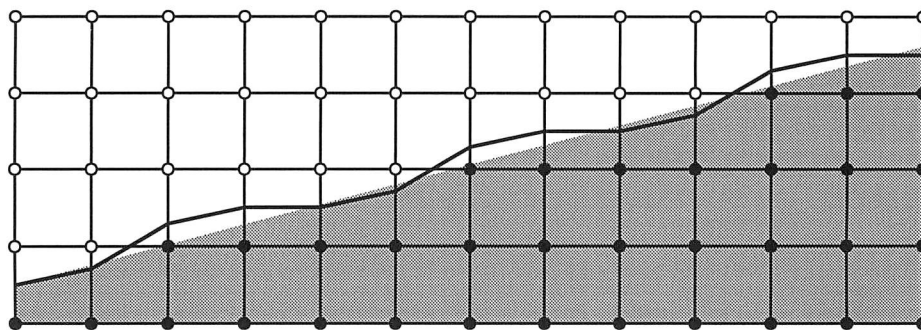
Når facetter fra disse tegl sættes sammen vil det give en ret linie som forløber parallelt med en af gitterets akser (der jo er vinklet i forhold til skillelinien som ønskes bestemt). Ind imellem vil der være et tegl af en anden stærk ækvivalensklasse end den ovenfor beskrevne, og den facet der konstrueres i dette tegl „bringer linien på plads“ fordi den er kommet for langt væk fra skillelinien. Dette vil fremstå som et knæk eller et trappetrin på en ellers ret linie, og fænomenet kaldes derfor for *trappeeffekten*.

9.1.2 Årsagen til problemet er blandt andet at der mistes information når det oprindelige felt registreres på gitteret. Man kan nemlig ikke længere udtale sig om kurvens detaljerede forløb inden for hvert tegl da man kun har oplysninger om feltets værdi i teglets hjørner. I stedet optræder der den i eksemplet viste form for *aliasing* som afhænger af gitterets orientering. Et andet problem er...

9.1.3 Antagelsen om lineær variation. Udtrykket (6.1) benyttes til at fastlægge skæringspunkterne, men hermed antager man at feltet tilnærmelsesvis varierer lineært. Ofte har feltet mere karakter af en trin-funktion hvor feltet springer mellem to værdier (eller værdiområder). Dette er oplagt i ovenstående eksempel hvor billedet er binært, men gælder osse for eksempel medicinske billeder hvor de forskellige organer har forskellige værdier som samtidig er rimelig konstante for hvert organ. Dette gør at skæringspunktternes placering, givet ved (6.1), egentlig er helt vilkårlig, men på den anden side kan man ikke angive et bedre estimat så længe feltet har denne „trin-karakter“.

9.1.4 Lavpas filtrering kan benyttes til at reducere trappeeffekten. Midlingsprocessen gør at feltets „trin-karakter“ formindskes, og konstruktionen af facetten i et kritisk tegl er nu ikke kun baseret på værdien i teglets fire hjørner, men osse i mindre grad på de omkringliggende hjørner som indirekte inddrages via filtrets kerne.

9.1.5 Eksempel. På figur 9.2 er vist konturen der fremkommer når billedet på figur 9.1 lavpas filtreres. Mere præcist, så er der foretaget en foldning af det binære digitale



Figur 9.2: Det samme binære billede som i figur 9.1. Denne gang er billedet først lavpas filtreret inden algoritmen er anvendt med tærskelværdi $T = 1/2$. De før så markante „trappetrin“ er nu blevet udjævnet ganske godt.

billede vist på figur 9.1 med Gauß kernen givet ved

$$[a_{l_1 l_2}] = \frac{1}{16} \begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix}.$$

Resultatet er et billede vist på figur 9.2 som ikke længere er binært, men som stadig indeholder de samme høje og lave hjørner som i det oprindelige billede. De nye billedværdier er ikke vist, men den konstruerede kurve for tærskelværdien $T = 1/2$ er vist. Man ser at den følger den „korrekt“ skillelinie langt bedre end i det første eksempel.

9.1.6 Kontekstafhængig lavpas filtrering. Det kunne umiddelbart se ud som om den konstruerede kontur bliver mere præcis hvis billedet først lavpas filtreres, men man skal huske på at en lavpas filtrering osse er med til at udslette detaljer. Denne reduktion af detaljer vil ofte være uønsket i områder med mange lav/høj overgange. Derfor kan man forestille sig at man udvikler en *kontekstafhængig lavpas filtrering* som afhængigt af konteksten bidrager til at udjævne trappeeffekten uden dog at vigtige detaljer bliver fjernet. Denne filtrering kan eventuelt indbygges direkte i tegl-algoritmen således at kun de billedpunkter som indgår i beregningen af konturens skæringspunkter bliver filtreret.

9.1.7 Bemærkning. Det er ikke givet at alle tegl vil bibeholde deres signatur efter en lavpas filtrering. Dette afhænger dels af den benyttede kerne og dels af tærskelværdien. Hvis billedet er binært og tærskelværdien er $0 < T < 1$ så kan man dog opstille en tilstrækkelig betingelse for at teglene ikke ændrer signatur. Kernen er på formen $[a_{l_1 l_2}]$ med $-d_1 \leq l_1 \leq d_1$ og $-d_2 \leq l_2 \leq d_2$, og det antages at den opfylder

$$a_{l_1 l_2} \geq 0, \quad \sum_{l_1, l_2} a_{l_1 l_2} = 1 \quad (9.1)$$

hvor den dobbelte sum over alle koefficienter i kernen er skrevet kort. I praksis vil en kerne til lavpas filtrering opfylde nogle flere krav, for eksempel at kernen er „symmetrisk“, men dette er ikke væsentligt i den følgende diskussion.

Det filtrerede billedpunkt får værdien givet ved definition (5.4.1) som kan omskrives til

$$V'(x_{j_1 j_2}) = a_{00} V(x_{j_1 j_2}) + \sum_{\substack{l_1 \neq 0 \\ l_2 \neq 0}} a_{l_1 l_2} V(x_{(j_1+l_1)(j_2+l_2)}).$$

Antag at hjørnet $x_{j_1 j_2}$ er lavt. Dette giver følgende udsagn

$$V(x_{j_1 j_2}) = 0 \implies V'(x_{j_1 j_2}) = \sum_{\substack{l_1 \neq 0 \\ l_2 \neq 0}} a_{l_1 l_2} V(x_{(j_1+l_1)(j_2+l_2)}) \leq \sum_{\substack{l_1 \neq 0 \\ l_2 \neq 0}} a_{l_1 l_2} = 1 - a_{00}$$

hvor uligheden fås fra det faktum at billedet er binært og at $a_{l_1 l_2} \geq 0$. En tilstrækkelig betingelse for at et lavt hjørne ikke skifter status må derfor være

$$1 - a_{00} < T.$$

Antag dernæst at hjørnet $x_{j_1 j_2}$ er højt. Da fås følgende udsagn

$$V(x_{j_1 j_2}) = 1 \implies V'(x_{j_1 j_2}) = a_{00} + \sum_{\substack{l_1 \neq 0 \\ l_2 \neq 0}} a_{l_1 l_2} V(x_{(j_1+l_1)(j_2+l_2)}) \geq a_{00}$$

hvor den sidste ulighed fås fra (9.1). En tilstrækkelig betingelse for at et højt hjørne ikke skifter status må derfor være

$$a_{00} \geq T. \quad (9.2)$$

Sammen med (9.2) givet dette at lav/høj status af hjørner i billedet bevares under en lavpas filtrering hvis der gælder

$$1 - a_{00} < T \leq a_{00}$$

og dette kan kun være opfyldt hvis

$$a_{00} > \frac{1}{2}.$$

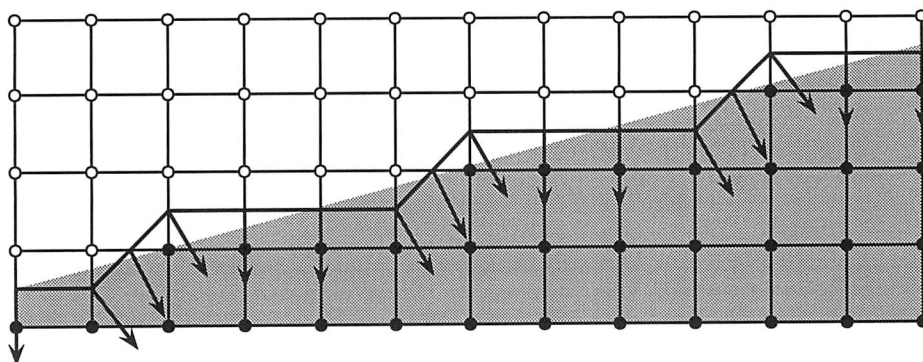
Bemærk at ovenstående udledning forudsætter at billedet er binært. Er det ikke det kan man ikke komme med tilsvarende generelle udsagn.

Man kan se at eksempel 9.1.5 ikke opfylder betingelsen da $a_{00} = 1/4$. Alligevel er der ingen tegn der skifter signatur under filtreringen i eksemplet, men havde der for eksempel været et lavt hjørne omgivet af lutter høje hjørner så ville dette hjørne have skiftet status.

Ovenstående argumenter lader sig direkte generalisere til tre dimensioner hvor samme resultat fås.

9.1.8 Eksempel. Ud fra det billedet i figur 9.1 er det muligt at beregne normalerne i alle kurvens skæringspunkter som beskrevet i afsnit 8.2. Gradienten i hvert hjørne i billedet estimeres med en todimensional Sobel kerne hvor x -komponenten er givet ved kernen

$$[a_{l_1 l_2}] = \frac{1}{8} \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix}$$



Figur 9.3: Det samme billede som i figur 9.1. Normalerne til kurven er beregnet i skæringspunkterne ved anvendelse af et Sobel filter og vist som pile. Omkring „trappetrinene“ er normalerne stort set ortogonale på skillelinien mellem sort og hvid.

og y-komponenten er givet ved

$$[a_{l_1 l_2}] = \frac{1}{8} \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$$

På figur 9.3 er normalerne i hvert skæringspunkt for kurven indtegnet. Selv om vektorernes retning varierer langs kurven må man alligevel sige at de alle er tæt på at være vinkelrette på den underliggende skillelinie mellem sort og hvidt, og det er den linie konturen skal approksimere.

Jeg skal indrømme at eksemplet ikke er helt overbevisende langs de vandrette stykker af konturen, men jeg vil alligevel komme med følgende påstand:

9.1.9 Anvendelse af normaler kan være med til at mindske trappeeffekten forstået på den måde at normalerne indeholder information som kan bidrage til mere præcist at forstå forløbet af en kontur.

Hvordan dette gøres i praksis i to dimensioner er ikke åbenbart, men i tre dimensioner (hvor trappeeffekten osse optræder) er det som oftest hensigten at visualisere den flade der bliver konstrueret. Anvendes en toning som afhænger af normalerne i skæringspunkt vil de ellers bratte overgange i fladens toner, som der normalt vil forekomme ved „trappetrin“, blive udjævnet af normalernes (forhåbentlig) mere regelmæssige forløb.

9.1.10 Form kontra tone. Ved visualiseringen af en flade vil den kvalitative opfattelse af fladens detaljer hos betragterne først og fremmest være bestemt af variationen af tonerne omkring detaljerne. Om detaljerne af fladen indeholder små „trappetrin“ er ikke så vigtigt når blot tonernes variation er tro mod det objekt der modelleres. Det er derfor vigtigt at man er i stand til at komme med gode estimater for fladens normaler (se afsnittene 4.3.14 og 8.2).

Det er dog ikke altid at tonen er i højsædet. Når man for eksempel konstruerer en fysisk model, i for eksempel kunststof, af en kontur fra af en tredimensional tegl-

algoritme, har man ikke mulighed for at benytte informationen fra normalerne. Det er formen, og intet andet, der fastlægger modellen og dermed vores kvalitative opfattelse af den. Det er i dette tilfælde at trappeeffekten bliver meget fremtrædende og det er hensigtsmæssigt at anvende andre algoritmer end de her beskrevne tegl-algoritmer [Swaelens 1992].

9.2 Anvendelse af n -topologien

9.2.1 Anvendelse af 0-topologien. Hidtil har billederne anvendt i forbindelse med tegl-algoritmerne været udstyret med 0-topologien. Valget af topologi er mest af „filosofisk“ art. For eksempel er de beskrevne tegl-algoritmer beskrevet karakteriseret ved at skæringspunkter er placeret i 1-cellerne i billedet. Der er intet til hinder for at modificere en tegl-algoritme så skæringspunkterne bliver placeret helt andre steder selv om 0-topologien anvendes. På den anden side vil det måske være vanskeligt at forstå en sådan modificeret algoritme hvis anvendelsen af 0-topologien fastholdes, lissom algoritmer til rumfangsgengivelse oftest er meget lettere at forstå når de beskrives inden for rammerne af n -topologien.

9.2.2 Anvendelse af n -topologien. Det er derfor naturligt, efter studiet af billeder med 0-topologien, at fortsætte med at studere billeder udstyret med n -topologien, for at se om det er muligt at opstille algoritmer til konstruktion af konturer. Det er netop hvad dette afsnit vil handle om, men i modsætning til de tidligere afsnit hvor tegl-algoritmerne er beskrevet meget detaljeret, så vil denne gennemgang være langt mere overfladisk og efterlade nogle åbne spørgsmål. På trods af dette synes nogle af resultaterne ganske interessante.

Som sædvanlig startes i...

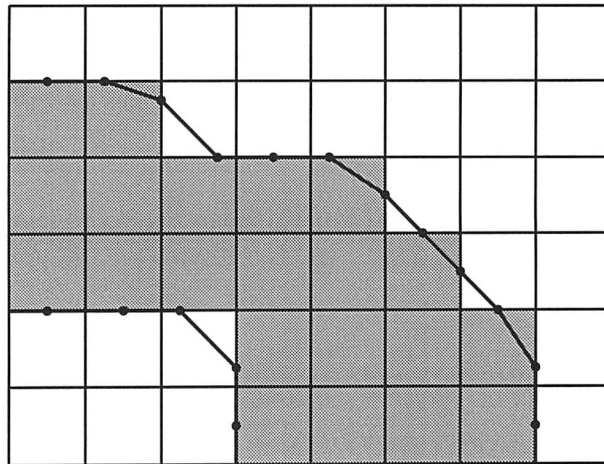
9.2.3 To dimensioner. Cellerne i billedet som har værdier og koordinater er nu 2-cellerne. En 2-celle kan være høj eller lav. Mellem 2-cellerne er der 1-celler eller *kanter*. En kant er *kritisk* hvis den ligger mellem to 2-celler som har forskellig lav/høj status.

9.2.4 Placering af skæringspunkt. En simpel algoritme til konstruktion af en kontur ville placere skæringspunkter midt i hver kritisk kant og forbinde disse til en sammenhængende kontur. Dette er dog ikke en særlig interessant algoritme da den minder meget om den todimensionale tegl-algoritme bortset fra at skæringspunkterne altid er placeret midt mellem to billedpunkter.

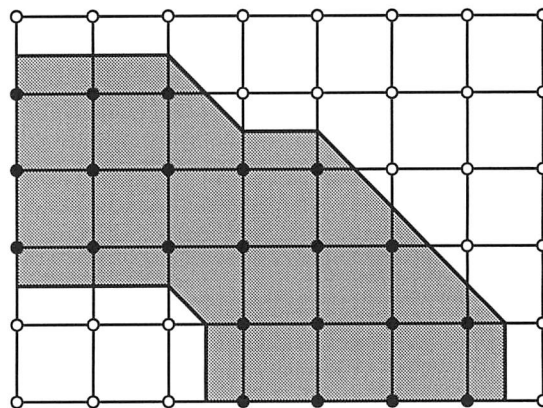
I stedet kan man udnytte den („filosofiske“) frihed man har til at placere skæringspunktet hvor som helst i den kritisk kant. Dette er vist på figur 9.4. Ved at forskyde skæringspunkterne passende i hver kritisk kant opnår man at konturen får et mere jævnt forløb. Dette gælder både omkring „hjørner“ i konturen og ved „trappetrin“ der opstår som følge af trappeeffekten.

Til sammenligning kan på figur 9.5 ses det samme billede, denne gang udstyret med 0-topologien, med en kontur konstrueret med den todimensionale tegl-algoritme.

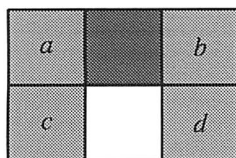
9.2.5 Anvendelse af gravitation til forskydning af skæringspunkter. For at forskyde skæringspunkterne anvendes følgende uformelle metode: Billedpunkter i omegnen af hver kritisk kant undersøges med henblik på at fastlægge en *gravitation* som forskyder



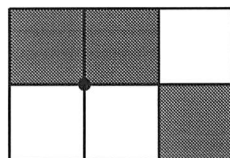
Figur 9.4: Et todimensionalt binært billede udstyret med n -topologien. En kontur er konstrueret som adskiller de høje fra de lave billedpunkt og de mørke skiver markerer skæringspunkterne for denne kontur.



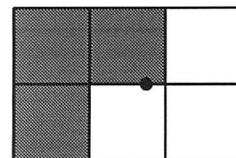
Figur 9.5: Det todimensionale billede på figur 9.4 udstyret med 0-topologien. En kontur er konstrueret for tærskelværdien $T = 1/2$ ved hjælp af den todimensional tegl-algoritme.



A) Lav/høj status af a , b , c og d undersøges.



B) Gravitationen er -2 .



C) Gravitationen er 1 .

Figur 9.6: Omegn af kritisk kant der undersøges for at forskyde skæringspunktet.

skæringspunktet i kanten. På figur 9.6.A er denne omegn illustreret i det tilfælde hvor den kritiske kant er vandret og hvor det høje billedpunkt er over det lave. Der er tre andre tilfælde som er fuldstændig analoge, men for at undgå unødvendige detaljer beskrives kun det illustrerede tilfælde.

9.2.6 Beregning af gravitation. Til at begynde med er skæringspunktets gravitation 0. Denne gravitation modificeres efter følgende system:

1. Hvis a er høj mindskes gravitationen med 1,
2. hvis b er høj øges gravitationen med 1,
3. hvis c er lav mindskes gravitationen med 1, og
4. hvis d er lav øges gravitationen med 1.

Efter dette system vil gravitationen være en af værdierne $\{-2, -1, 0, 1, 2\}$. Denne værdi angiver hvor på kanten skæringspunktet skal befinde sig — er gravitationen -2 skal skæringspunktet være længst til venstre, mens en gravitation på 0 giver en placering midt i kanten og så fremdeles. Der er vist to eksempler på dette på figur 9.6.B og 9.6.C.

9.2.7 Tre dimensioner. Ideen med at forskyde skæringspunktet lader sig uden videre generalisere til tre dimensioner. I dette tilfælde er det sider (det vil sige 2-celler) der kan være kritiske og skæringspunktet kan frit placeres i denne 2-celle. Forskydningen i hver af de to retninger i siden foregår uafhængigt af hinanden. Med andre ord så er skæringspunktets gravitation denne gang med todimensional vektor hvor hvert element antager værdier mellem -2 og 2 .

9.2.8 Anvendelse af tegl-algoritmen. Denne anvendelse af n -topologi har hidtil udelukkende tjent til at demonstrere hvordan skæringspunktet mellem et højt og et lavt billedpunkt kan justeres ud fra værdien af omgivende billedpunkter for at opnå en mere glat kontur. *Det betyder at man er i stand til at anvende en tegl-algoritme til at fastlægge sammenhængen mellem skæringspunkterne.* Kun beregningen af skæringspunktets placering skal modificeres.

Set i det lys er det nærliggende at bibeholde estimationen af skæringspunktets placering mellem det lave og det høje billedpunkt ved lineær interpolation givet ved (6.1). Nu kan skæringspunktet blot osse flyttes i et plan ortogonal på den kritiske kant hvor „kritisk kant“ er en kant i 0-topologien.

9.2.9 Begrænsning af forskydning. For at undgå at facetter bliver selvgenemskaarende, eller på anden måde forvredne, er det nødvendigt at indskrænke „udsvinget til siden“ af skæringspunktet når det ikke er placeret midt mellem det høje og det lave billedpunkt. Hvis man begrænser skæringspunktets placering til en bipyramide med toppunkter i de to billedpunkter og grundplan i n -topologiens kritiske side, kan man se at hvert muligt skæringspunkt ligger i disjunkte mængder givet ved bipyramiden for skæringspunktet.

9.2.10 Mulighed for sammenfald af skæringspunkter. Da bipyramiderne kun har disjunkte indre er der mulighed for sammenfald af skæringspunkter hvis den numeriske værdi af gravitationen er 2 da skæringspunktet så vil ligge på randen af bipyramiden. Det betyder at der kan optræde en ny form for udartede skæringspunkter som ikke umiddelbart kan opdages ved anvendelse af den hash tabel teknik som er beskrevet i bemærkning 7.8.4.

Dette kan løses på forskellige måder: Én metode vil være at benytte et passende produkt af skæringspunktets koordinater afrundet nedad til nærmeste heltal som indeks i hash tabellen. To skæringspunkter med ens indeks som ikke er ens kan skelnes ved en sammenligning af deres eksakte koordinater. Dette introducerer forskellige problemer omkring præcision og sammenligning af flydende-komma tal på computer, men disse problemer er ikke uoverstigelige.

9.2.11 Variationer af algoritmen. Det er flere ting der kan varieres i den ovenfor skitserede algoritme. For eksempel kan man, når man beregner skæringspunktets forskydning, inddrage værdien af de omgivende billedpunkter i stedet for blot deres lav/høj status. Gravitationen kan osse „omvendes“ ved i afsnit 9.2.6 at ombytte lav med høj samtidig med at gravitationerne skifter fortegn. Endelig er det ikke sikkert at ideen med bipyramiden skal anvendes da et skæringspunkt som er „nær“ toppen af en af pyramiderne ikke er forskudt nævneværdigt i forhold til et tilsvarende skæringspunkt placeret af tegl-algoritmen. Disse ideer bør undersøges for at afgøre om de med fordel kan anvendes til at konstruere konturflader.

9.3 Triangulering af tegl

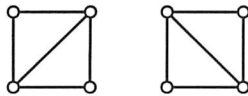
Indtil nu er et plant billede blevet overdækket med kvadratiske tegl og et rumligt billede med kubiske tegl, men det er klart at osse andre tesseleringer er mulige. Den enklest tænkelige tesselering er en overdækning af planet med trekanter og rummet med tetraedre (det vil sige en overdækning med *simpleks*).

9.3.1 Simpliciale tegl-algoritmer. De hidtil beskrevne tegl-algoritmer kan stort set anvendes uændrede når simpleks benyttes som tegl. Dog er der en række forenklinger: Antallet af ækvivalensklasser er reduceret. I to dimensioner er der $2^3 = 8$ stærke ækvivalensklasser og tre svage ækvivalensklasser, mens der i tre dimensioner er $2^4 = 16$ stærke ækvivalensklasser og fire svage ækvivalensklasser. Endvidere optræder der *ingen* flertydige tegl da simpleks ikke har diagonaler mellem hjørner.

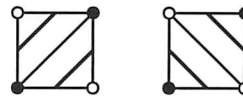
9.3.2 Uorganiserede data. Har man uorganiserede skalardata, det vil sige data som ikke falder på et gitter, kan man foretage Delaunay triangulering af disse data [Okabe et al. 1992] hvorefter en tegl-algoritme med simpliciale tegl kan anvendes. Derved kan en kontur konstrueres.

9.3.3 Triangulering af tegl. Oftest er data dog registreret på et kubisk gitter og så er det ikke naturligt at anvende simpleks som tegl. Alligevel kan man gå fra de sædvanlige tegl til simpliciale tegl ved at triangulere de kubiske tegl.

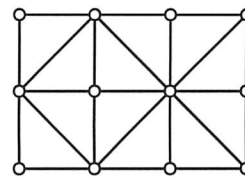
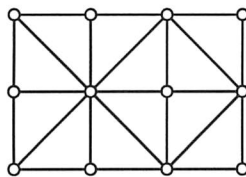
9.3.4 Asymmetrisk triangulering. Det sædvanlige tegl i to dimensioner er et kvadrat. Dette kan trianguleres ved at tilføje en diagonal hvorved der dannes to trekanter som vist på figur 9.7.A. Som man kan se er der to symmetriske trianguleringer som



A) To komplementære trianguleringer af et 2-tegl.

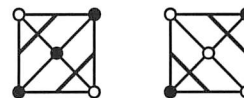


B) Et flertydigt tegl. Det højt og det lavt sammenhængende alternativ vælges afhængigt af trianguleringen.



C) To komplementære trianguleringer af et lille billede.

Figur 9.7: Asymmetrisk triangulering i to dimensioner.



A) Triangulering af et 2-tegl. Bemærk at der introduceres et hjørne i centrum af det oprindelige tegl.

B) Et flertydigt tegl. Flertydighed afgøres i realiteten ved estimationsmetoden.

Figur 9.8: Symmetrisk triangulering i to dimensioner.

man kan kalde for *komplementære*. Dette gør at en triangulering af et helt billede må anvende halvt af hvert alternativ som vist på figur 9.7.C hvor to komplementære trianguleringer er vist. Denne form for triangulering kan kaldes for *asymmetrisk* da der introduceres en asymmetri ved at vælge ét af de to symmetriske alternativer.

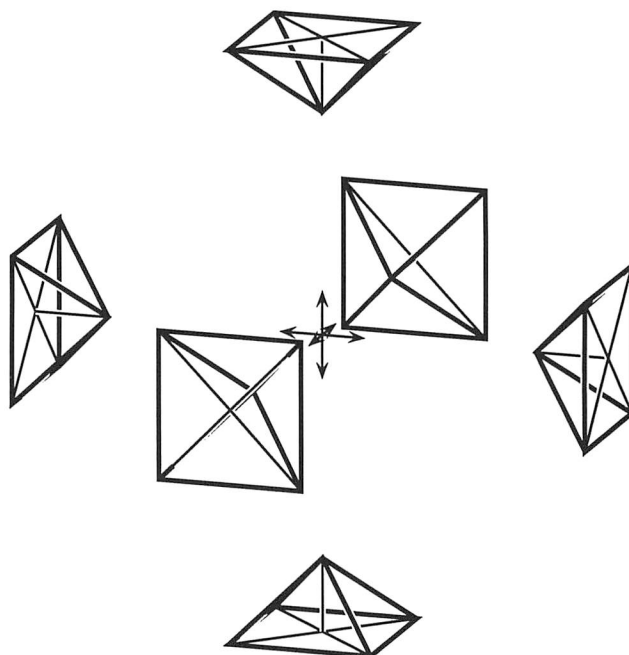
9.3.5 Flertydighed. Et flertydigt tegl som trianguleres asymmetrisk bliver delt op i to ikke-flertydige tegl. Dette er vist på figur 9.7.B hvor de resulterende facetter osse er vist. Man kan se at det højt sammenhængende alternativ bliver anvendt i det ene tilfælde, mens det lavt sammenhængende alternativ bliver anvendt i det andet tilfælde, det vil sige i det komplementære tilfælde.

Det betyder at et (tilfældigt) valg af én af to mulige asymmetrisk trianguleringer afgør flertydighed. Dette kan virke utilfredsstillende, og i stedet kan man foretage en...

9.3.6 Symmetrisk triangulering. Ved at tilføje begge kvadratets diagonaler fås fire trekanter som vist på figur 9.8.A. Problemet er at man må tilføje et hjørne i skæringen af de to diagonaler, og da billedets værdifunktion V ikke er defineret i dette hjørne må man estimere en værdi af V for eksempel ved at beregne gennemsnit af de fire øvrige hjørner. Ikke overraskende kaldes denne form for triangulering for *symmetrisk*.



Figur 9.9: To komplementære trianguleringer af et kubisk tegl som resulterer i fem tetraedre.



Figur 9.10: En eksploderet tegning af en symmetrisk triangulering af et kubisk tegl. Der er i alt 24 tetraedre, og pilene i midten markerer det oprindelige centrum af kubusen hvorfra de seks pyramider hver bestående af fire tetraedre er skubbet ud fra.

9.3.7 Estimationsmetoden. Et flertydigt tegl som trianguleres symmetrisk bliver delt op i fire tegl som ikke kan være flertydige. De facetter der konstrueres kommer dog til at afhænge af lav/høj status af det centrale hjørne der introduceres med trianguleringen. På figur 9.8.B kan man se at et højt (lavt) centralt hjørne giver det der svarer til det højt (lavt) sammenhængende alternativ i det ikke triangulerede tilfælde.

Overvejer man det kan man se at symmetrisk triangulering afgør flertydighed på samme måde som hvis man anvender *estimationsmetoden*.

9.3.8 Komplicerede facetter. Prisen er dog at facetterne kompliceres da der introduceres flere skæringspunkter. Dette gælder såvel i det asymmetriske som i det symmetriske tilfælde. I det symmetriske tilfælde skal man endog estimere værdien af en række nye hjørner. Det er derfor tvivlsomt om man vinder så meget ved at gå over til simpliciale tegl (med mindre data fra starten foreligger på et simplicialt gitter).

9.3.9 Tre dimensioner. De to hidtil beskrevne former for triangulering af et kvadratisk 2-tegl kan nemt generaliseres til tre dimensioner. De to komplementære asymmetriske trianguleringer er vist på figur 9.9 og den symmetriske triangulering er vist på figur 9.10.

9.3.10 Asymmetrisk triangulering. Når et tredimensionalt billede trianguleres asymmetrisk bliver de seks nærmeste naboer til et kubisk tegl trianguleret komplementært for at sikre konsistens. Dette skyldes at de diagonaler der introduceres i siderne af kuberne skal „passe sammen”. Derved opnås automatisk at halvt af hver af de to komplementære alternativer anvendes.

Om flertydighed gælder der at 2-flertydighed afgøres ved valg af én af de to komplementære trianguleringer, mens 3-flertydighed altid bliver afgjort ved anvendelse af det adskillende alternativ.

9.3.11 Symmetrisk triangulering. Anvendes symmetrisk triangulering skal værdien af syv nye hjørner estimeres, men så bliver man osse i stand til at afgøre såvel 2- som 3-flertydighed. Prisen er, ud over de mange estimater der skal beregnes, osse nogle temmelig komplicerede facetter: Hvis for eksempel det oprindelige kubiske tegl har ét højt hjørne er facetten en trekant, men efter en symmetrisk triangulering vil facetten bestå af mindst seks trekanter!

9.3.12 Konklusion. Det er klart at anvendelsen af simpleks som tegl i stedet kvadratiske eller kubiske tegl forenkler tegl-algoritmerne, men sædvanligvis foreligger data nu engang på et kubisk gitter og anvendelsen af simpleks som tegl er derfor ikke helt enkel.

Asymmetrisk triangulering er ikke i stand til at afgøre flertydighed tilfredsstillende i modsætning til symmetrisk triangulering der klarer dette fuldt tilfredsstillende. Desværre bliver „prisen” temmelig høj da der estimeres værdier i nye hjørner, og de resulterende facetter er mere komplicerede. Da antallet af flertydige tegl i reglen ikke er særlig stort ville det være mere ønskeligt om man havde en algoritme der kun introducerede disse „komplikationer” i de flertydige tilfælde. Men sådan en algoritme er allerede beskrevet — det er en tegl-algoritme der afgør flertydighed ved estimationsmetoden.

Man kan osse tænke sig en (tredimensional) algoritme der behandler alle ikke-flertydige kubiske tegl som tidligere beskrevet, men som i de flertydige tilfælde anvender symmetrisk triangulering hvorefter en tegl-algoritme baseret på simpliciale tegl anvendes. En sådan algoritme ville være i stand til at håndtere 3-flertydighed tilfredsstillende.

Anvendelsen af simpliciale tegl i forbindelse med analytiske felter til modellering af syntetiske objekter er noget som kan vise sig at være frugtbart. Endnu mangler dette dog at blive undersøgt i detaljer.

9.4 Marching cubes

Selv om Lorensen og Cline [1987] ikke anvender begreber som *symmetrigruppe* og *ækvivalens* i deres beskrivelse af „marching cubes“ algoritmen så er det essentielt disse formelle begreber de anvender. *Men de anvender en anden symmetrigruppe.*

Når hjørnerne i et kritisk tegl får deres lav/høj status inverteret, således at høj bliver til lav og lav bliver til høj, så kan den samme facet anvendes til at adskille

de høje fra de lave hjørner. Denne symmetrioperation, der permuterer med de øvrige symmetrioperationer, kan tilføjes til G således at gruppen får i alt 92 elementer.

Med denne symmetrigruppe fås fjorten svage ækvivalensklasser (da Lorensen og Cline [1987] ikke benytter operationen μ får de femten svage ækvivalensklasser). Problemet er at konsistens ikke længere kan sikres når de svage ækvivalensklasser benyttes til at konstruere en opslagstabel. I stedet kan der konstrueres konturer som vist på figur 7.3.

Inversion af lav/høj kan anvendes som symmetrioperation i det todimensionale tilfælde uden at der opstår problemer med konsistens (da konsistens her afhænger af 0-facetterne), men det er ikke muligt at anvende den konsekvente metode til afgørelse af flertydighed. Dette medfører at svag ækvivalens i det todimensionale tilfælde ikke kan benyttes til at konstruere en opslagstabel, og inversion af lav/høj har derfor ingen praktiske anvendelse i det todimensionale tilfælde. For den interesserede kan det dog nævnes at antallet af svage ækvivalensklasser bliver fire i stedet for seks.

9.5 Noter og litteratur

I afsnit 4.5.10 er det allerede omtalt hvor i litteraturen tegl-algoritmer er beskrevet, men her er nogle flere tekniske detaljer. Wyvill et al. [1986] beskriver en algoritme som i princippet benytter rekursion over lavere dimensioner og hvor flertydighed afgøres ved estimationsmetoden.

Lorensen og Cline [1987] og Cline et al. [1988] introducerer „marching cubes“ algoritmen som benytter en opslagstabel, og den anvendes på medicinske data der er et meget aktivt forskningsområde. Derfor er algoritmen meget udbredt, dette på trods af, at algoritmen konstruerer inkonsistente konturer. Som tidligere beskrevet er 2-flertydighed ikke løst tilfredsstillende. Dette problem er blevet påpeget af blandt andre Düurst [1988], Baker [1989], Payne og Toga [1990], Wilhelms og Van Gelder [1990b] og Liversage et al. [1992].

Wilhelms og Van Gelder [1990b] konstruerer to analytiske felter som (for en passende tærskelværdi) har en ækvipotentialflade der består af henholdsvis én og to komponenter. Groft sagt kan man sige at de to flader er stort set ens, bortset fra i et område hvor den ene flade er sammenhængende og den anden flade er ikke-sammenhængende og derfor har to komponenter. Når disse felter registreres på et passende gitter vil de for en passende tærskelværdi have nøjagtig de samme høje og lave hjørner og alle tegl vil derfor have samme signatur.

I artiklen viser man at „marching cubes“ algoritmen konstruerer inkonsistente konturer og man forsøger at benytte modificerede algoritmer som konstruerer konsistente konturer. Forskellige former for estimationsmetoder anvendes for at undersøge om det er muligt at skelne mellem de to analytiske flader der har et forskelligt antal komponenter.

Det vises at estimationsmetoden baseret på simpel midling beskrevet her ikke er i stand til at skelne mellem de to flader, mens man ved at benytte fjerde ordens estimerer for centerværdierne bliver i stand til at skelne. Dette er dog ikke overraskende da de analytiske flader er kubiske, og jeg er derfor uenig i artiklens konklusion som peger mod at fjerde ordens metoder skal anvendes. I medicinske data har feltet som regel en meget ulineær karakter og det er ikke sandsynligt at en fjerde ordens approksimation er meget bedre end en første ordens.

Kapitel 10

Højere dimensioner

10.1 Generalisation til fire dimensioner

10.2 Anvendelse af den firedimensionale algoritme

10.3 Eksempler på højere dimensionale data

10.4 Generelle facetter

10.5 En generel tegl-algoritme

I afsnit 7.6 blev det demonstreret hvordan den tredimensionale algoritme rekursivt kan anvende den todimensionale algoritme. Det er derfor naturligt at overveje om den idé kan udvides til vilkårlige dimensioner. Det kan den, men for ikke at fare for voldsomt frem beskrives der i første omgang en firedimensional tegl-algoritme. Her omtales osse hvordan algoritmen kan anvendes i praksis, og der bringes nogle forslag til kilder til højere dimensionale data. Til slut i dette kapitel bringes en beskrivelse af hvordan facetter og tegl-algoritmer kan generaliseres til vilkårlige dimensioner. Da der endnu er nogle huller i den formelle teori har disse afsnit et noget ufærdigt præg. Nogle af manglerne i beskrivelsen vil dog blive påpeget så stoffet her kan tjene som et fundament for videre udviklinger.

10.1 Generalisation til fire dimensioner

10.1.1 Tegl i fire dimensioner. I fire dimensioner er teglet en hyperkube eller *tesseract* som består af én 4-celle, otte 3-celler, 24 2-celler, 32 1-celler og seksten 0-celler. Et kritisk 4-tegl behandles ved at anvende den tredimensionale algoritme på hvert kritisk 3-tegl i 4-teglet. Dette resulterer i en række 2-facetter. Skæringspunkterne i disse 2-facetter er nu punkter i $\mathbb{R}^4$.

Da 2-facetterne er konstrueret således at konsistens mellem nabo-tegl er sikret vil 2-facetterne i 4-teglet „passe sammen“ langs deres kanter. Dermed kan de sættes sammen til en *3-facet*.

10.1.2 Uformel beskrivelse af 3-facetter. Da 2-facetter blev introduceret blev de beskrevet som „ikke-plane polygoner“ i rummet. Betegnelsen er misvisende da polygoner netop er plane. Alligevel er den upræcise betegnelse god til at karakterisere 2-facetter hvis man ikke ønsker at komme med lange udredninger omkring triangulering af facetten og så videre.

På samme måde kan man benytte den upræcise betegnelse „ulinear polyeder“ om 3-facetter. Man kan tænke på en 3-facet som en polyeder (godt nok i $\mathbb{R}^4$) som har fået sine sider „krummet“ en smule.

10.1.3 Triangulering af 3-facetter. Man kan osse præcisere denne beskrivelse ved at triangulere 3-facetten. Her er det mest hensigtsmæssigt at anvende barycentrisk triangulering som er beskrevet i afsnit 8.1.18.

For hver 2-facet i 3-facetten beregnes 2-facettens barycenter og dens mål. I dette tilfælde bliver målet 2-facettens areal. Dette kan benyttes til at tildele 3-facetten et barycenter ved at midle over 2-facetternes barycentre med 2-facettens mål som vægt. For at triangulere 3-facetten skal man først triangulere alle 2-facetterne. Derefter forbindes alle skæringspunkter samt barycentre af 2-facetterne med 3-facettens barycenter. Derved bliver 3-facetten inddelt i tetraedre. Man kan tænke på disse tetraedre som pyramider der som grundplan har en trekant i en 2-facet og som toppunkt har 3-facettens barycenter.

Denne idé kan generaliseres til n dimensioner. Dette bliver gjort her selv om n -facetten endnu ikke er blevet introduceret. Hver n -facet (som ikke er et n -simpleks) tænkes trianguleret ved at indføre barycentret af facetten givet ved følgende rekursive definition.

10.1.4 Definition. (Barycenter.) Lad n -facetten ϕ sammensat af $(n - 1)$ -facetterne $\epsilon_1, \dots, \epsilon_m$ være givet. Barycentret af ϕ er

$$B_n(\phi) = \frac{\sum_{i=1}^m M_{n-1}(\epsilon_i) B_{n-1}(\epsilon_i)}{\sum_{i=1}^m M_{n-1}(\epsilon_i)}$$

hvor $M_k(\phi')$ er det k -dimensionale mål af k -facetten ϕ' . Dette er veldefineret da alle facetterne tænkes trianguleret barycentrisk og M_k bliver derfor en sum over det k -dimensionale volumen af en række k -simpleks.

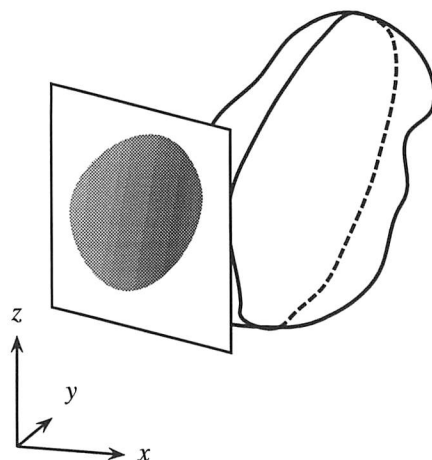
10.1.5 Resultatet af algoritmen. Alle 3-facetterne i hvert kritisk 4-tegl udgør til sammen en lukket 3-kontur som adskiller de lave fra de høje hjørner i 4-billedet. Konturen der fås fra den tredimensionale algoritme er en flade og på samme måde ses 3-konturen at være en tredimensional mangfoldighed i det firedimensionale rum.

Denne 3-kontur kan repræsenteres ved at generalisere ideerne beskrevet i afsnit 7.7. Løst sagt skal man tilføje et „niveau“ af 3-facetter „over“ 2-facetterne i figurerne der ledsager teksten. Dette kan generaliseres til højere dimensioner hvor en mangfoldighed opbygget af n -facetter repræsenteres af et hierarki i n niveauer.

10.2 Anvendelse af den firedimensionale algoritme

Da man ofte betragter tiden som en fjerde dimension er det ikke svært at forestille sig firedimensionale billeder. Man kan tænke på dem som tredimensionale billeder der varierer i tiden. Med andre ord så er et firedimensionalt billede et rumfang der er registreret i en række tidsskridt.

10.2.1 Eksempler er der mange af. Man kan for eksempel forestille sig at man „filmer“ et hjerte der slår, en muskel der trækker sig sammen, eller et stof der transporteres rundt i et organ. Det lyder enkelt, men man skal huske på at det at erhverve et tredimensionalt billede ved hjælp af for eksempel CT eller MRI kan være en relativt langvarig proces. Det nyeste teknologiske udvikling går dog i retning af kortere og kortere skannetider med mulighed for gentagelser hvorved ægte firedimensionale billeder uden bevægelsesstøj måske bliver tilgængelige.



Figur 10.1: Principskitse der viser hvordan en flade projiceres på skærmen, her repræsenteret af rektanglet.

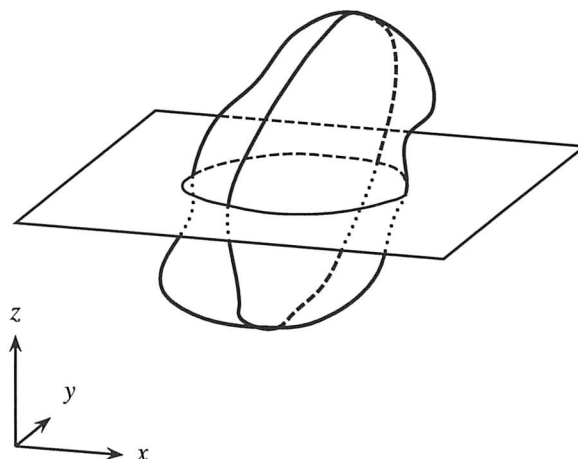
10.2.2 Uddata fra algoritmen. Det „objekt“ der konstrueres af den firedimensionale algoritme er som tidligere nævnt en tredimensional mangfoldighed i det firedimensionale rum, med den egenskab at den adskiller de høje fra de lave hjørner. Mangfoldigheden er opbygget af punkter i det firedimensionale rum som er forbundet med linier. Disse linier udspænder (krumme) flader som igen udspænder (krumme) polyedere. Disse polyedere er byggestenene (facetterne) i mangfoldigheden.

10.2.3 Analogi med tre dimensioner. For at forstå hvorledes den tredimensionale mangfoldighed kan visualiseres er det hensigtsmæssigt først at overveje hvordan en todimensionale mangfoldighed indlejret i det tredimensionale rum kan visualiseres, det vil sige hvordan en flade i rummet kan visualiseres.

10.2.4 Projektion på skærmen. Det mest almindelige form for visualisering er en metode hvor fladen projiceres på et todimensionalt (affint) underrum der svarer til skærmen. Hvis man tilføjer lyskilder til det tredimensionale rum og materialeegenskaber til fladen kan man beregne hvorledes lyset reflekteres fra fladen ind på skærmen, det vil sige det todimensionale underrum. Dette er vist på figur 10.1.

Det menneskelige sanseapparat er indrettet således at når man betragter sådan en projektion (et billede) så er man i stand til at erkende fladens rumlige struktur — specielt hvis man kan flytte det underrum der svarer til skærmen og beregne nye projektioner. Det svarer for eksempel til at bevæge sig rundt om fladen eller ækvivalent at rotere fladen.

10.2.5 Projektion fra fire til to dimensioner. Noget tilsvarende kan man gøre i fire dimensioner: I dette tilfælde projiceres den tredimensionale mangfoldighed i det firedimensionale rum på det todimensionale underrum der repræsenterer skærmen. Det lyder enkelt, men de menneskelige sanser er ikke indrettet på at erkende strukturen af en



Figur 10.2: Principskitse der viser hvordan en flades skæring med et plan er en lukket kurve.

sådan mangfoldighed ud fra en projektion, og det vil derfor være svært at forstå det man ser uden en del øvelse.

Derfor må man søge efter alternative måder at visualisere en sådan mangfoldighed. Igen startes i tre dimensioner:

10.2.6 Skæring med plan. Skæringen mellem et (todimensionalt) plan og fladen i rummet vil i reglen være en eller flere kurver som vist på figur 10.2. Ved at flytte planet ses forskellige kurver og på den måde kan man få et indblik i fladens struktur.

10.2.7 Skæring med hyperplan. I fire dimensioner benyttes et tredimensionalt hyperplan, det vil sige et tredimensionalt (affint) underum. Skæringen mellem den tredimensionale mangfoldighed og hyperplanet er i reglen en eller flere flader, det vil sige todimensionale mangfoldigheder. Den eller de flader er indeholdt i hyperplanet og kan derfor afbildes som almindelige flader i rummet. Men ud fra hvilke kriterier skal hyperplanet vælges?

10.2.8 Valg af hyperplan. Antag at det firedimensionale rum kan opfattes som det sædvanlige tredimensionale rum beskrevet ved x -, y -, og z -koordinater tilføjet tidsdimensionen t . Hvis hyperplanet vælges ortogonal på t -aksen definerer hyperplanet et bestemt tidspunkt. Den flade der er skæringen mellem hyperplanet og den tredimensionale mangfoldighed er en tilnærmelse til ækvipotentiafladen på det tidspunkt hyperplanet definerer.

10.2.9 Nyttens af den firedimensionale algoritme. Ud fra den betragtning kan det virke overflødigt at have en firedimensional tegl-algoritme — har man tidsvarierende data kan man i stedet anvende den tredimensionale algoritme for hver hel t -værdi (med „hel t -værdi“ forstås de t -værdier hvor 3-billedet er registreret). Men det er en overfladisk betragtning. Man kan nemlig vælge hyperplaner i ikke-hele t -værdier — den tredimen-

sionale algoritme vil kun være i stand til at konstruere flader i de to hele t -værdier der „omgiver“ den ikke-hele t -værdi. Derimod vil skæringen mellem hyperplanet fastlagt af den ikke-hele t -værdi og mangfoldigheden bestemt af den firedimensionale algoritme, være en flade som er en slags *interpolation* mellem to flader fra den tredimensionale algoritme.

10.2.10 Analogien i tre dimensioner er at sige at man kun behøver den todimensionale algoritme til at konstruere kurver for hele værdier af z . Det er præcis det man gør i kurvebaserede algoritmer hvor man bestemmer konturer i x - y -planet for hele værdier af z . Modsætningen til dette er isotropiske algoritmer, såsom tegl-algoritmer, der bestemmer en konturflade, osse *mellem* hele z -værdier. Tager man en konturflade og skærer den med et plan ortogonal på z -aksen fås en konturkurve, men gøres dette for en ikke-hel z -værdi er den resulterende kurve ikke en kurve man kunne have fået fra den todimensionale algoritme.

At benytte den firedimensionale algoritme på tidsvarierende data i stedet for at benytte den tredimensionale algoritme for hver hel tidsværdi giver oplysninger om mangfoldighedens forløb, osse *mellem* hele t -værdier.

10.2.11 Et tankeeksperiment. Med den fornødne computerkapacitet kunne man konstruere et program der i sand tid snittede og projicerede mangfoldigheden til en bruger-valgt t -værdi. Brugeren kunne, med for eksempel en knap, justere den valgte t -værdi. På den måde kunne man „spole“ frem og tilbage i tiden som om det var en film, bortset fra at filmen ikke bestod af enkelte billeder, men var kontinuert så selv små bevægelser af knappen ville give en „glat“ animation. Et tilsvarende system baseret på anvendelsen af den tredimensionale algoritme i hvert tidsskridt ville ved en langsom bevægelse af knappen give en ujævn opdatering af billedet i stil med det man oplever når man kører en smalfilm igennem billede for billede.

Et system som det netop beskrevne baseret på den firedimensionale algoritme ville derfor være at foretrække, men den krævede computerkapacitet vil formodentlig ikke være almindeligt tilgængelig i den nærmeste fremtid.

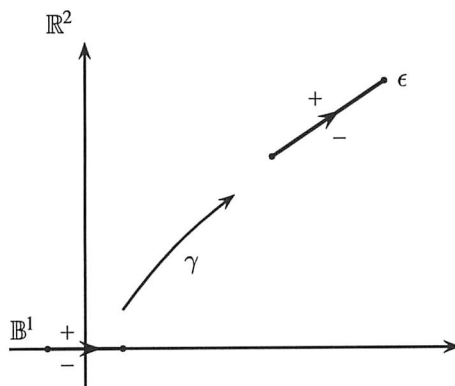
10.2.12 Beregning af skæringen mellem den tredimensionale mangfoldighed og det tredimensionale hyperplan i en ikke-hel t -værdi kræver nogen omtanke. Det kan gøres i hvert tegl for sig, og i tre dimensioner svarer det til at man i et kritisk tegl forsøger at bestemme facettens skæring med et plan som går gennem teglet.

10.3 Eksempler på højere dimensionale data

Ud over tidsvarierende data er der osse andre former for data som har en dimensioner højere end tre. Her er et interessant eksempel.

10.3.1 Skalarum. Inden for billedanalyse og maskin-syn arbejder man med *skalarum*. Ved at lavpas filtrere et billede med en mere og mere „uskarp“ kerne fås en hel serie billeder som bliver mere og mere „udtværede“. Derved går man fra et detaljeret billede til et „overordnet“ billede, og på den måde håber man at blive i stand til at identificere noget af den „overordnede“ information i billedet.

10.3.2 Skalaparameter som ekstra dimension. Man kan lade skalaparameteren være den tredje dimension hvorved serien af billeder bliver et tredimensionalt billede af ska-



Figur 10.3: Illustration af hvorledes en 1-facet tilordnes to sider.

larummet. Med for eksempel en tærskelværdi kan konturerne af objekterne i billedet lokaliseres, og ved at anvende den tredimensionale algoritme kan udviklingen af konturerne i skalarummet studeres. Den mest stabile beskrivelse af et objekt i dette rum kan være i det område hvor dets udvikling under ændring af skalaparameteren er minimal, og dette område kan bestemmes ved at studere fladen konstrueret af tegl-algoritmen [Baker 1989].

10.3.3 Højere dimensioner. Ideen kan generaliseres ved at tage et tredimensionalt billede og tilføje skalaparameteren som en fjerde dimension. Eller man kan tage et firedimensionalt billede, det vil sige et tidsvarierende tredimensionalt billede og tilføje skalaparameteren som den femte dimension. De to sidste forslag forudsætter dog at man er i stand til at anvende resultatet til noget praktisk, det vil blandt andet sige at „forstå“ de tidligere beskrevne højere dimensionale mangfoldigheder.

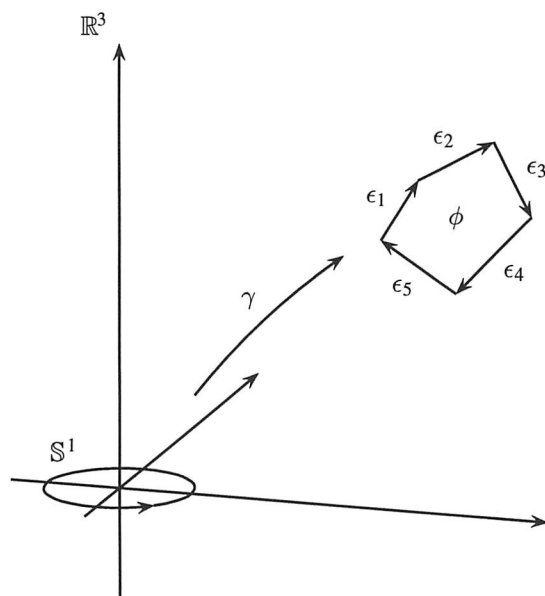
10.4 Generelle facetter

Ved at opstille en generel formalisme i de tidligere kapitler har jeg forsøgt at blive i stand til at komme med nogle præcise udsagn om for eksempel n -facetters orientering. Desværre må jeg erkende at jeg ikke har nået det mål på grund af min manglende formåen inden for matematik. Alligevel vil jeg her forsøge at redegøre for mine ideer, selv om disse ideer kan synes noget upræcise og vanskeligt tilgængelige.

10.4.1 Orientering i lave dimensioner. For at fastlægge orienteringen af 1- og 2-facetter i det to- og tredimensionale tilfælde anvendes regler som involverer højre og venstre. Spørgsmålet er, hvordan disse „højrehåndsregler“ generaliseres til fire og højere dimensioner. Derfor må „højrehåndsreglen“ formaliseres.

10.4.2 Standard orientering. Det euklidiske rum $\mathbb{R}^n$ har to mulige orienteringer. Lad en standard orientering være valgt. I tilfældet $n = 2$ haves en standard omløbsretning mod uret og i tilfældet $n = 3$ haves et „højrehånds“ koordinatsystem. At beskrive orienteringen af en facet kommer ud på at beskrive hvorledes disse standard orienteringer overføres til facetter.

10.4.3 Orientering af 1-facetter. Betragt en 1-facet i $\mathbb{R}^2$ (se figur 10.3). En 1-facet ϵ er et éndimensionalt cellekompleks med én 1-celle og er dermed homeomorf med $\mathbb{B}^1$.



Figur 10.4: Illustration af hvorledes en 2-facet ϕ konstrueres fra 1-facetterne $\epsilon_1, \dots, \epsilon_5$.

Hvis $\mathbb{B}^1$ tænkes indlejret i $\mathbb{R}^2$ ved indlejringen ι defineret i 5.2.2 så kan man løst sagt sige at den inverse karakteristiske afbildning $\gamma^{-1}: \mathbb{B}^1 \rightarrow \epsilon$ „flytter“ facetten ϵ ned i en standard placering i $\mathbb{R}^2$. I denne „standard placering“ har man det positive og det negative halvrum $\mathbb{R} \times \mathbb{R}_+$ og $\mathbb{R} \times \mathbb{R}_-$. Løst sagt kan disse halvrum flyttes „op til“ $\epsilon \in \mathbb{R}^2$ via γ hvorved man får en *positiv* og en *negativ* side af ϵ .

Denne „overførsel“ af positiv og negativ side til facetten svarer egentlig til at udvide γ til en afbildning $\gamma: \mathbb{R}^2 \rightarrow \mathbb{R}^2$. Denne afbildning skal være *orienteringsbevarende* for at positiv og negativ ikke skal ombyttes.

10.4.4 Orientering af 2-facetter. Næste problem bliver at forstå hvordan en 2-facet sammensættes af 1-facetter og på den måde får en orientering.

Ved anvendelse af den todimensionale algoritme fås en række 1-facetter som til sammen udgør en lukket kurve (se figur 10.4). Denne kurve er homeomorf med $\mathbb{S}^1$ som er indlejret i $\mathbb{R}^3$ ved indlejringen ι . For at få 2-facetten fra den lukkede kurve kan man uformelt tænke sig at kurven „flyttes ned“ til $\mathbb{S}^1$. Her „tilføjes“ det indre af $\mathbb{S}^2$, det vil sige $\mathbb{B}^2$, hvorefter man „flytter tilbage“ til den oprindelige position. På den måde *adjungeres* en 2-celle til den lukkede kurve og en 2-facet opstår. Samtidig får facetten automatisk en orientering, det vil sige en positiv og en negativ side som beskrevet ovenfor med den modifikation at det positive og det negative halvrum i dette tilfælde er $\mathbb{R}^2 \times \mathbb{R}_+$ og $\mathbb{R}^2 \times \mathbb{R}_-$.

10.4.5 Generelle facetter fås ved rekursion. Konstruktionen skitseret ovenfor kan fortsættes ad infinitum. Mere præcist så fås en n -cykel fra et antal n -facetter som til sammen udgør en mængde homeomorf med $\mathbb{S}^n$. Hver af n -facetterne skal have en orientering der stemmer overens således at det er meningsfyldt at tale om n -cyklens orientering, og n -facetterne må kun overlappe på deres $(n-1)$ -skeletter. En $(n+1)$ -facet fås fra en n -cykel ved adjungering af en $(n+1)$ -celle. Orienteringen af facetten fås fra orienteringen af indlejringen af $\mathbb{S}^n$ i $\mathbb{R}^{n+1}$ ved ι .

10.4.6 De forbigåede problemer. Den kritiske læser vil nok føle at der er en del huller i ovenstående beskrivelse af hvordan en n -facet konstrueres, og det er ganske rigtigt.

Formelt mangler der blandt andet et bevis for at man er i stand til sammensætte $(n-1)$ -facetter til et cellekompleks som er homeomorft med $\mathbb{S}^{n-1}$. Specielt at $(n-1)$ -facetter konstrueret med en $(n-1)$ -dimensional tegl-algoritme giver facetter der har en „konsistent“ orientering.

Det forudsættes osse at den karakteristiske afbildning γ kan udvides til $\neg\gamma$ som oven i købet skal være orienteringsbevarende. Dette sidste krav betyder essentielt at afbildningen skal have positiv determinant, men det er i almindelighed meningsløst at tale om determinanten af en kontinuert afbildning med mindre afbildningen er lineær. Begrebet orienteringsbevarende kan dog udvides til differentiable afbildninger — her skal man se på fortegnet af Jacobi determinanten. Endvidere kan man i stedet for lineære afbildninger anvende stykkevis lineære afbildninger osse kaldet *PL-afbildninger* (for *Piecewise Linear*).

10.4.7 Anvendelse af PL-homeomorfier. Matematikere finder det er ofte hensigtsmæssigt at arrangere tingene således at de karakteristiske afbildninger for celler i cellekomplekser er PL. Derved undgås nogle af de ellers patologiske matematiske konstruktioner man kan opstille hvis de karakteristiske afbildninger ikke opfylder andre krav end at de er homeomorfier.

Ved anvendelse af trianguleringen skitseret i definition 10.1.4 er det klart hvorledes n -facetten får en PL-struktur. Problemet er at „standard cellerne“ $\mathbb{B}^n$ i almindelighed ikke er PL-homeomorfe med simpleks eller cellekomplekser sammensat af simpleks. Alternativt kan som „standardcelle“ benyttes $[-1; 1]$ der er homeomorf (men ikke PL-homeomorf) med $\mathbb{B}^n$.

Hele problematikken kan formodentlig omformuleres inden for rammerne af algebraisk topologi idet opbygningen af n -facetter ved hjælp af $(n-1)$ -facetter udgør en *eksakt sekvens*. Inden for algebraisk topologi har man en generel beskrivelse af orientering som formodentlig kan anvendes, men alt dette ligger langt uden for mine kundskaber.

10.5 En generel tegl-algoritme

10.5.1 Algoritmen i korte træk. Umiddelbart er algoritmen enkel: hvert kritisk n -tegl dekomponeres rekursivt indtil den to- eller tredimensionale algoritme kan anvendes. Dette giver 1- eller 2-facetter som samles via konstruktion af cykler indtil en eller flere $(n-1)$ -facetter er konstrueret. En $(n-1)$ -kontur opbygget af $(n-1)$ -facetter vil være et hierarki af facetter af forskellige dimensionalitet forbundet i en træstruktur. Denne struktur er for tre dimensioner beskrevet i afsnit 7.7.

10.5.2 Orientering af tegl. En ting skal man være opmærksom på og det er hvorledes $(n-1)$ -teglene fås fra n -teglet i rekursionen. Hvert $(n-1)$ -tegl i n -teglet kan nemlig have to forskellige orienteringer. Hvorledes man vælger den korrekte orientering i tre dimensioner er beskrevet i afsnit 7.6. Mere generelt kan n -teglet opfattes som et n -dimensionalt cellekompleks med én n -celle. Dette cellekompleks er homeomorft med $\mathbb{B}^n$, og randen som er homeomorf med $\mathbb{S}^{n-1}$ er sammensat af $(n-1)$ -tegl. Kuglen $\mathbb{B}^n$ har samme orientering som $\mathbb{R}^n$ og derved har $\mathbb{S}^{n-1}$ en orientering. Hvert $(n-1)$ -tegl

skal have samme orientering, og af de to mulige vælges den som stemmer overens med S^{n-1} 's orientering.

10.5.3 Flertydighed. Som beskrevet i afsnit 7.3 findes der $n-1$ former for flertydighed i n -teglet, men som nævnt er det kun 2-flertydighed det er betydningsfuldt at tages hensyn til. Dette kan gøres ved anvendelse af estimationsmetoden i den todimensionale algoritme som anvendes på sidste trin af rekursionen over dimensioner.

10.5.4 Uafklarede emner. Det er et åbent spørgsmål om de konstruerede facetter er gyldige i vilkårlige dimensioner. En ugyldig facet vil for eksempel være en facet som er selvgennemskærende eller som skærer en anden facet i teglet. Et andet interessant spørgsmål er hvilke krav man må stille til et tegl for at konstruktionen af facetter bliver gyldig. Hidtil er det kun kubiske (og simpliciale) tegl der er blevet beskrevet. Et naturligt krav er at teglet skal være konvekst, men skal dette krav yderligere skærpes?

10.5.5 Implementering. Beskrivelsen af den n -dimensionale algoritme har været ret abstrakt. I princippet kan man konstruere en algoritme der kan håndtere alle endelige dimensioner, men dette kræver nogle meget komplicerede datatyper da dimensionen af billedet ikke er kendt på forhånd.

I stedet kan man for en given værdi af n foretage en implementering. Det er det der i detaljer har beskrevet i tre dimensioner. For n større end tre (måske fire) er det dog så kompliceret at man ikke kan gøre det fejlfrit i hånden. I stedet kan man konstruere en kodegenerator der er i stand til at generere den krævede kode. I de fleste praktiske tilfælde vil det nok være nødvendigt da man må forudse at den ovenfor omtalte generelle implementering vil være alt for langsom.

Det er derfor klart at der endnu resterer noget arbejde inden jeg for alvor kan tillade mig at sige at jeg har konstrueret en n -dimensional algoritme. Vejen er dog vist.

Kapitel 11

Afslutning

11.1 Resumé

11.2 Fordele ved tegl-algoritmer

11.3 Ulemper ved tegl-algoritmer

11.4 Konklusion

I dette afsnit vil jeg benytte lejligheden til at se tilbage på hvad jeg har opnået. Første del af dette speciale indeholder stof der for det meste er beskrevet i litteraturen. Det er derfor ikke nødvendigt at rekapitulere dette. Derimod er de i anden del mange nye emner og ideer, og her vil jeg godt fremhæve nogle af de vigtigste. Jeg vil osse opveje fordele og ulemper ved de her beskrevne tegl-algoritmer, især sammenlignet med rumfangsgengivelse som osse kan anvendes til tredimensional visualisering.

11.1 Resumé

Ved at „låne“ cellekomplekser fra kombinatorisk og algebraisk topologi får jeg et stærkt værktøj til beskrivelse af digitale billeder og flader. Jeg opstiller en matematisk abstraktion af et n -dimensionalt billede og demonstrerer hvordan billedet naturligt kan udstyres med to forskellige topologier: 0- og n -topologien. Dette gør det enkelt at beskrive tegl og facetter som er centrale begreber for tegl-algoritmer.

I forbindelse med tegl-algoritmer analyseres osse flertydighed, og jeg viser at der findes flere slags flertydighed — én for hver dimension. Endvidere beskriver jeg forskellige metoder til løsning af 2-flertydighed som er den mest bemærkelsesværdige form for flertydighed.

Ved en systematisk analyse af symmetrier af tegl opstiller jeg en generel teknik der kan benyttes til at konstruere opslagstabeller som giver en effektiv implementering af tegl-algoritmen. Dermed bringer jeg for første gang en teoretisk velfunderet forklaring på hvorfor forskellige forfattere når frem til et forskelligt antal svage ækvivalensklasser. Det skyldes at man har valgt forskellige symmetrigrupper.

Som alternativ til anvendelse af opslagstabel viser jeg hvorledes man kan foretage rekursion over dimensionen. Dette giver en algoritme der i princippet kan anvendes i vilkårlige dimensioner. Denne algoritme kan osse kombineres med algoritmen baseret på anvendelsen af en opslagstabel. Derved får man en effektiv algoritme som samtidig kan anvende estimationsmetoden i de flertydige tilfælde. Det er den algoritme jeg har implementeret.

Jeg viser osse flere metoder til triangulering af facetter, og diskuterer hvordan man kan estimere normaler i facetternes skæringspunkter. Dette kræver nogen omtanke hvis det underliggende billede er udstyret med en koordinatfunktion som ikke er regulær.

Endvidere studerer jeg trappeeffekten i detaljer, og jeg beskriver hvordan kubiske tegl kan trianguleres så simpliciale tegl i stedet kan anvendes. Her er konklusionen at det ikke er specielt nyttigt at triangulere tegl selv om det umiddelbart ser ud til at flertydighed på den måde kan løses — komplikationerne er alt for store i forhold til det man opnår, og flertydighed er løst tilfredsstillende i den algoritme jeg har implementeret. Endelig er der nogle foreslag til hvordan n -topologien kan anvendes i forbindelse med en ny algoritme. Her resterer der dog endnu noget arbejde.

11.2 Fordele ved tegl-algoritmer

Et af de væsentlige karakteristika ved tegl-algoritmer er at konturen som konstrueres, den estimeres inden for dimensioner som er mindre end billedpunkterne i billedet. Set fra et signalteoretisk synspunkt er dette i en vis forstand ugyldigt. Nyquists sætning siger at man ikke korrekt kan rekonstruere et signal der indeholder frekvenser der er højere end halvdelen af den frekvens signalet registreres med. Omsat til et praktisk medicinsk eksempel betyder det, løst sagt, at hvis afstanden mellem to tomogrammer i en medicinsk skanning er 2 mm, så kan man ikke være sikker på at detaljer i konturen som er mindre end 4 mm er korrekte.

Høje frekvenser i billedet kan fjernes ved en lavpas filtrering hvorved den konstruerede kontur kan hævdes at være mere „korrekt“. På den anden side så ville en bruger formodentlig synes at resultatet virker „udtværet“ og har for få detaljer.

Jeg mener at det er vigtigt at bevare detaljerigdommen som tegl-algoritmerne tilbyder. Selv om konturerne ikke er fuldstændig korrekte er de ofte tæt på virkeligheden, og brugerne (for eksempel lægerne) er ofte i stand til at identificere interessante strukturer som ikke var synlige på en tilsvarende rumfangsgengivelse (se for eksempel figur 7.17 på side 99). Det er naturligvis vigtigt at brugeren ikke forledes til at tro at det som ses er en fuldstændig korrekt gengivelse af naturen. Ofte vil man jo anvende medicinsk visualisering i forbindelse med kirurgisk behandlingsplanlægning og her kan fejlfortolkninger være meget alvorlige.

En anden fordel ved anvendelsen af tegl-algoritmer sammenlignet med rumfangsgengivelse er at man går fra *data-rum* til *objekt-rum*. Dermed bliver man i stand til at manipulere objekterne i billedet i stedet for de enkelte billedpunkter.

Dette kan for eksempel benyttes til at *matche* forskellige modeller af samme patient. Forskellene kan være opstået som følge af vækst [Kreiborg et al. 1992a] eller som følge af et operativt indgreb.

Algoritmer til rumfangsgengivelse forudsætter ofte at de enkelte voxels enten er kubiske eller i de mindste har ens dimensioner overalt i rumfanget (svarende til en regulær koordinatfunktion). Medicinske data har meget sjældent kubiske voxels og har ofte osse varierende snitafstand. Dette betyder at anvendelsen af rumfangsgengivelse til visualisering ofte kræver reinterpolation af data på et kubisk gitter. Dette ekstra, ofte tidskrævende, trin er *ikke* nødvendigt hvis tegl-algoritmer anvendes til visualisering.

Endelig kan man ved visualiseringen af en konturflade benytte en lang række avancerede teknikker kendt fra *computer graphics*, for eksempel flere lyskilder og „realistiske“ lysberegninger, som man på den måde får „foræret“.

11.3 Ulemper ved tegl-algoritmer

En af manglerne ved anvendelse tegl-algoritmer til visualisering af for eksempel medicinske data er at man normalt kun kan visualisere én vævstype ad gangen. Ved rumfangsgengivelse har man mulighed for at anvende transparente voxels hvorved man kan studere sammenhængen mellem forskellige typer væv. Ved anvendelse af α -blanding kan to konturflader godt afbildes transparent foran hinanden, men det væv der er mellem to flader bidrager ikke til det endelige billede, og løsningen er derfor ikke tilfredsstillende sammenlignet med de tilsvarende muligheder man har til rådighed ved anvendelse af rumfangsgengivelse.

En mulighed er at kombinere rumfangsgengivelse og visualisering af konturflader. De særlige problemer der opstår i forbindelse med dette er blandt andet beskrevet af Kaufman et al. [1990] og Max et al. [1990].

Et andet problem i forbindelse med tegl-algoritmer er de forholdsvis store computerkapacitet der kræves — både til beregning og til lagring af konturerne — men dette problem støder man på inden for mange områder af medicinsk billedbehandling da de underliggende datamængder er så omfattende. Udviklingen går imod hurtigere og hurtigere systemer så derfor vil det gradvist blive nemmere at håndtere store mængder medicinske data. Endvidere er tegl-algoritmer velegnede til parallelisering da forskellige tegl kan behandles uafhængigt af hinanden.

11.4 Konklusion

Konklusionen er naturligvis at tegl-algoritmer og rumfangsgengivelse er metoder til visualisering der supplerer hinanden. Alt efter anvendelsen skal man vælge den metode der er mest hensigtsmæssig. Dette speciale indeholder en detaljeret beskrivelse af hvordan tegl-algoritmer kan anvendes.

Til slut kan jeg nævne at *General Electrics* efter sigende har et patent på „marching cubes“ algoritmen. Det har desværre ikke været muligt for mig at finde ud hvad dette patent omfatter. Endvidere har *Cyberware* patent på algoritmen beskrevet af Schroeder et al. [1992] til forenkling af flader konstrueret med „marching cubes“.

Man kan se at der er store forretningsmæssige interesser inden for nogle af de områder jeg har beskrevet i dette speciale — nu må jeg bare håbe at mit speciale ikke er omfattet af noget patent.

Bibliografi

- [Agoston 1976]: Max K. Agoston. *Algebraic Topology*. Monographs and Textbooks in Pure and Applied Mathematics. Marcel Dekker, Inc., 1976.
- [Altschuler 1979]: M. D. Altschuler. Reconstruction of the global-scale three-dimensional solar corona. I [Herman 1979], kapitel 4, siderne 105–145.
- [Analyze 1993]: Biomedical Imaging Resource, Mayo Foundation. *Analyze, Reference Manual, Version 6.2*, 1993.
- [Artzy et al. 1981]: E. Artzy, G. Frieder, og G. T. Herman. The theory, design, implementation, and evaluation of a three-dimensional surface detection algorithm. *Computer Graphics and Image Processing*, 15:1–24, januar 1981.
- [Baker 1989]: H. Harlyn Baker. Building surfaces of evolution: The weaving wall. *International Journal of Computer Vision*, 3:51–71, 1989.
- [Barrett 1984]: Harrison H. Barrett. The Radon transform and its applications. I E. Wolf (redaktør), *Progress in Optics XXI*, kapitel III, siderne 217–286. North-Holland Physics Publishing, 1984.
- [Bergtin et al. 1993]: E. Bergtin, F. Parazza, og J. M. Chassery. Segmentation and measurement based on 3D voronoi diagram: Application to confocal microscopy. *Computerized Medical Imaging and Graphics*, 17(3):175–182, 1993.
- [Bloomenthal 1988]: Jules Bloomenthal. Polygonization of implicit surfaces. *Computer Aided Geometric Design*, 5(4):341–355, november 1988.
- [Bogomolny 1988]: A. Bogomolny. Digital geometry may not be discrete. *Computer Vision, Graphics, and Image Processing*, 43(2):205–220, august 1988.
- [Boissonnat 1988]: Jean-Daniel Boissonnat. Shape reconstruction from planar cross sections. *Computer Graphics and Image Processing*, 44(1):1–29, oktober 1988.
- [Boman 1990]: Jan Boman. Helgason's support theorem for Radon transforms — a new proof and a generalization. I [Herman et al. 1990], siderne 1–5.
- [Bracewell 1979]: R. N. Bracewell. Image reconstruction in radio astronomy. I [Herman 1979], kapitel 3, siderne 81–104.
- [Bransden og Joachain 1983]: B. H. Bransden og C. J. Joachain. *Physics of Atoms and Molecules*. Longman Scientific & Technical, 1983.
- [Budinger et al. 1979]: T. F. Budinger, G. T. Gullberg, og R. H. Huesman. Emission computed tomography. I [Herman 1979], kapitel 5, siderne 147–246.
- [Cartwright 1990]: Mark Cartwright. *Fourier Methods: for Mathematicians, Scientists and Engineers*. Mathematics and its Applications. Ellis Horwood, 1990.
- [Censor 1990]: Yair Censor. On variable block algebraic reconstruction techniques. I [Herman et al. 1990], siderne 133–140.
- [Chang et al. 1991]: Long-Wen Chang, Hown-Wen Chen, og Ju-Rone Ho. Reconstruction of 3D medical images: A nonlinear interpolation technique for reconstruction of 3D medical images. *Graphical Models and Image Processing*, 53(4):382–391, juli 1991.
- [Cline et al. 1988]: H. E. Cline, W. E. Lorensen, S. Ludke, C. R. Crawford, og B. C. Teeter. Two algorithms for the three-dimensional reconstruction of tomograms. *Medical Physics*, 15(3):320–327, 1988.
- [Coatrieux og Barillot 1990]: Jean-Louis Coatrieux og Christian Barillot. A survey of 3D display techniques to render medical data. I [Höhne et al. 1990b], siderne 175–195.

- [Colchester og Hawkes 1991]: A. C. F. Colchester og D. J. Hawkes (redaktører). *Information Processing in Medical Imaging*, bind 511 af *Lecture Notes in Computer Science*. Springer-Verlag, 1991.
- [Cormack 1963]: A. M. Cormack. Representation of a function by its line integrals, with some radiological applications. *Journal of Applied Physics*, 34(9):2722–2727, september 1963. Første artikel i en serie på to.
- [Cormack 1964]: A. M. Cormack. Representation of a function by its line integrals, with some radiological applications. II. *Journal of Applied Physics*, 35(10):2908–2913, oktober 1964. Anden artikel i en serie på to.
- [Cormack 1980]: Allan M. Cormack. Early two-dimensional reconstruction and recent topics stemming from it. *Science*, 209:1482–1486, september 1980. Nobelpris tale.
- [Cormack 1983]: Allan M. Cormack. Computed tomography: Some history and recent developments. I [Shepp 1983a], siderne 35–42.
- [Deans 1983]: Stanley R. Deans. *The Radon Transform and some of its Applications*. John Wiley & Sons, Inc., 1983.
- [Dold 1972]: Albrecht Dold. *Lectures on Algebraic Topology*. Springer-Verlag, 1972.
- [Drebin et al. 1988]: Robert A. Drebin, Loren Carpenter, og Pat Hanrahan. Volume rendering. *Computer Graphics*, 22(4):65–74, august 1988.
- [DTI 1992]: Dansk Teknologisk Institut. *Rapid Prototyping*, Århus, september 1992.
- [Düurst 1988]: Martin J. Düurst. Additional reference to „marching cubes“. *Computer Graphics*, 22(2):72–73, april 1988.
- [Ell og Holman 1982]: P. J. Ell og B. L. Holman (redaktører). *Computed emission tomography*. Oxford University Press, 1982.
- [Foley et al. 1990]: James D. Foley, Andries van Dam, Steven K. Feiner, og John F. Hughes. *Computer Graphics: Principles and Practice*. The Systems Programming Series. Addison-Wesley Publishing Company, anden udgave, 1990.
- [Fritsch og Piccinini 1990]: Rudolf Fritsch og Renzo A. Piccinini. *Cellular Structures in Topology*. Nummer 19 i Cambridge Studies in Advanced Mathematics. Cambridge University Press, 1990.
- [Fuchs et al. 1977]: H. Fuchs, Z. M. Kedem, og S. P. Uselton. Optimal surface reconstruction from planar contours. *Communications of the ACM*, 20(10):693–702, 1977.
- [Gindi et al. 1991]: G. Gindi, M. Lee, A. Rangarajan, og I. G. Zubal. Bayesian reconstruction of functional images using registrered anatomical images as priors. I [Colchester og Hawkes 1991], siderne 121–131.
- [Globus 1990]: Al Globus. Octree optimization. Technical report RNR-90-011, NASA Ames Research Center, 1990.
- [Gordon og Udupa 1989]: Dan Gordon og Jayaram K. Udupa. Fast surface tracking in three-dimensional binary image. *Computer Vision, Graphics, and Image Processing*, 45(2):196–214, februar 1989.
- [Grassin et al. 1990]: Patricia Grassin, Bernard Duchene, og Walid Tabbara. Diffraction tomography: Some applications and extension to 3-D ultrasound imaging. I [Herman et al. 1990], siderne 98–105.
- [Gustafson 1985]: D. E. Gustafson. Single photon emission-computed tomography. I [Robb 1985a], kapitel 1, siderne 1–39.
- [Hall og Warren 1990]: Mike Hall og Joe Warren. Adaptive polygonalization of implicitly defined surfaces. *Computer Graphics & Applications*, 10(6):33–42, november 1990.

- [Hanrahan 1990]: Pat Hanrahan. Three-pass affine transforms for volume rendering. *Computer Graphics*, 24(5):71–78, november 1990.
- [Hejtmánek 1981]: J. Hejtmánek. The problem of reconstructing objects from projections as an inverse problem in scattering theory of the linear transport operator. I [Herman og Natterer 1981], siderne 29–35.
- [Helgason 1983]: Sigurdur Helgason. Ranges of Radon transforms. I [Shepp 1983a], siderne 63–70.
- [Herman 1979]: G. T. Herman (redaktør). *Image Reconstruction from Projections: Implementations and Applications*, bind 32 af *Topics in Applied Physics*. Springer-Verlag, 1979.
- [Herman 1985]: Gabor T. Herman. X-ray computed tomography — basic principles. I [Robb 1985b], kapitel 3, siderne 61–79.
- [Herman 1990]: Gabor T. Herman. On topology as applied to image analysis. *Computer Vision, Graphics, Image Processing*, 52(3):409–415, december 1990.
- [Herman et al. 1990]: G. T. Herman, A. K. Louis, og F. Natterer (redaktører). *Mathematical Methods in Tomography*, bind 1497 af *Lecture Notes in Mathematics*. Springer-Verlag, 1990.
- [Herman og Bucholtz 1991]: G. T. Herman og C. A. Bucholtz. Shape-based interpolation using a chamfer distance. I [Colchester og Hawkes 1991], siderne 314–325.
- [Herman og Natterer 1981]: G. T. Herman og F. Natterer (redaktører). *Mathematical Aspects of Computerized Tomography*, bind 8 af *Lecture Notes in Medical Informatics*. Springer-Verlag, 1981.
- [Herman og Webster 1981]: Gabor T. Herman og Dallas Webster. Surfaces of organs in discrete three-dimensional space. I [Herman og Natterer 1981], siderne 204–224.
- [Hertle 1981]: Alexander Hertle. On the problem of well-posedness for the Radon transform. I [Herman og Natterer 1981], siderne 36–44.
- [Hill og Hinshaw 1985]: Barbara C. Hill og Waldo S. Hinshaw. Fundamentals of NMR imaging. I [Robb 1985a], kapitel 4, siderne 79–124.
- [Hilton og Wylie 1960]: P. J. Hilton og S. Wylie. *Homology Theory*. Cambridge University Press, 1960.
- [Höhne et al. 1990a]: Karl Heinz Höhne, Michael Bomans, Andreas Pommert, Martin Riemer, Ulf Tiede, og Gunnar Wiebecke. Rendering tomographic volume data: Adequacy of methods for different modalities and organs. I [Höhne et al. 1990b], siderne 197–215.
- [Höhne et al. 1990b]: Karl Heinz Höhne, Henry Fuchs, og Stephen M. Pizer (redaktører). *3D Imaging in Medicine: Algorithms, Systems, Applications*, bind 60 af *NATO ASI Series F*. Springer-Verlag, 1990.
- [Höhne og Bernstein 1986]: K. Höhne og R. Bernstein. Shading 3D-images from CT using gray-level gradients. *IEEE Transactions on Medical Imaging*, MI-5(1):45–47, marts 1986.
- [Holm 1988]: Søren Holm. *Dynamic SPECT of the Human Brain*. PhD afhandling, University of Copenhagen, Physics Laboratory, H. C. Ørsted Institute, Copenhagen, 1988.
- [Hounsfield 1973]: G. N. Hounsfield. Computerized transverse axial scanning (tomography): Part 1. description of system. *British Journal of Radiology*, 46:1016–1022, 1973.
- [Hounsfield 1980]: G. N. Hounsfield. Computed medical imaging. *Science*, 210:22–28, 1980. Nobelpris tale.
- [Kalender 1993]: W. A. Kalender. Quo vadis CT? CT in the year 2000. *Electro Medica*, 61(2):30–39, 1993.
- [Kaufman et al. 1990]: Arie Kaufman, Roni Yagel, og Daniel Cohen. Intermixing surface and volume rendering. I [Höhne et al. 1990b], siderne 217–227.
- [Kepple 1975]: E. Kepple. Approximating complex surfaces by triangulation of contour lines. *IBM Journal of Research and Development*, 19:2–11, 1975.

- [Kovalevsky 1989]: V. A. Kovalevsky. Finite topology as applied to image analysis. *Computer Vision, Graphics, and Image Processing*, 46(2):141–161, maj 1989.
- [Kreiborg et al. 1992a]: Sven Kreiborg, Martin Liversage, M. Michael Cohen, Jr., Hans Pedersen, og Flemming Skovby. Ny teknik til tredimensional analyse af kraniofacial vækst ud fra CT-scannings data. *Tandlægebladet*, 96(18):858–860, december 1992.
- [Kreiborg et al. 1992b]: Sven Kreiborg, Martin Liversage, Hans Pedersen, og Karsten Lumbye. Anvendelse af 3D-teknikker og stereolitografi-modeller i udforskning og behandling af alvorlige medfødte ansigtsmisdannelse. I [DTI 1992].
- [Krueger 1990]: Wolfgang Krueger. Volume rendering and data feature enhancement. *Computer Graphics*, 24(5):21–26, november 1990.
- [Laur og Hanrahan 1991]: David Laur og Pat Hanrahan. Hierarchical splatting: A progressive refinement algorithm for volume rendering. *Computer Graphics*, 25(4):285–288, juli 1991.
- [Leahy og Yan 1991]: R. Leahy og X. Yan. Incorporation of anatomical MR data for improved functional imaging with PET. I [Colchester og Hawkes 1991], siderne 105–120.
- [Levoy 1988]: M. Levoy. Display of surfaces from volume data. *IEEE Computer Graphics & Applications*, 8(3):29–37, maj 1988.
- [Lin et al. 1989]: Wei-Chung Lin, Shih-Yung Chen, og Chin-Tu Chen. A new surface interpolation technique for reconstruction 3D objects from serial cross-sections. *Computer Vision, Graphics, and Image Processing*, 48(1):124–143, oktober 1989.
- [Liversage et al. 1992]: Martin Liversage, Gulab Bhatia, og Michael Vannier. Marching cubes: Algorithmic improvements. *Automedica*, 14:297–310, 1992.
- [Lorensen og Cline 1987]: William E. Lorensen og Harvey E. Cline. Marching cubes: A high resolution 3D surface construction algorithm. *Computer Graphics*, 21(4):163–169, juli 1987.
- [Lyndon 1985]: Roger C. Lyndon. *Groups and Geometry*, bind 101 af *London Mathematical Society Lecture Notes Series*. Cambridge University Press, 1985.
- [Maass 1990]: Peter Maass. Singular value decompositions for Radon transforms. I [Herman et al. 1990], siderne 6–14.
- [Malandain et al. 1991]: G. Malandain, G. Bertrand, og N. Ayache. Topological classification in digital space. I [Colchester og Hawkes 1991], siderne 300–313.
- [Marr et al. 1981]: Robert D. Marr, Ching-Nien, og Paul C. Lauterbur. On two approaches to 3D reconstruction in NMR zeugmatography. I [Herman og Natterer 1981], siderne 225–240.
- [Massey 1967]: William S. Massey. *Algebraic Topology: An Introduction*. Springer-Verlag, 1967.
- [Max et al. 1990]: Nelson Max, Pat Hanrahan, og Roger Crawfis. Area and volume coherence for efficient visualization of 3D scalar functions. *Computer Graphics*, 24(5):27–33, november 1990.
- [Mortenson 1989]: M. E. Mortenson. *Computer Graphics: An Introduction to the Mathematics and Geometry*. Industrial Press Inc., 1989.
- [Müller og Stark 1993]: Heinrich Müller og Michael Stark. Adaptive generation of surfaces in volume data. *The Visual Computer*, 9(4):182–199, januar 1993.
- [Natterer 1980]: Frank Natterer. A Sobolev space analysis of picture reconstruction. *SIAM Journal of Applied Mathematics*, 39(3):402–411, december 1980.
- [Niblack 1985]: Wayne Niblack. *An Introduction to Digital Image Processing*. Strandberg Publishing Company, 1985.
- [Niland 1982]: R. A. Niland. Maximum reliable information obtainable by tomography. *Journal of Optical Society of America*, 72(12):1677–1682, december 1982.

- [Niland og Schmidt-Harms 1985]: R. A. Niland og C. A. Schmidt-Harms. Reduction of a time of flight experiment to tomography. *Nuclear Instruments and Methods in Physics Research*, A241:221–235, 1985.
- [Okabe et al. 1992]: Atsuyuki Okabe, Barry Boots, og Kokichi Sugihara. *Spatial Tessellations: Concepts and Applications of Voronoi Diagrams*. Wiley Series in Probability and Mathematical Statistics. John Wiley & Sons, 1992.
- [Payne og Toga 1990]: Bradley A. Payne og Arthur W. Toga. Surface mapping brain function on 3D models. *IEEE Computer Graphics & Applications*, siderne 33–41, september 1990.
- [Pierro 1990]: Alvaro R. De Pierro. Multiplicative iterative methods in computed tomography. I [Herman et al. 1990], siderne 167–186.
- [Ponce og Faugeras 1987]: Jean Ponce og Olivier Faugeras. An object centered hierarchical representation for 3D objects: The prism tree. *Computer Vision, Graphics, and Image Processing*, 38(1):1–28, april 1987.
- [Quinto 1990]: Eric Todd Quinto. Computed tomography and rockets. I [Herman et al. 1990], siderne 261–268.
- [Radon 1983]: Johann Radon. Über die bestimmung von funktionen durch ihre integralwerte längs gewisser mannigfaltigkeiten. I [Shepp 1983a], siderne 71–86. Genoptryk af Radons originale artikel fra 1917.
- [Rees et al. 1981]: Charles Sparks Rees, S. M. Shah, og Č. V. Stanojević. *Theory and Applications of Fourier Analysis*, bind 59 af *Monographs and Textbooks in Pure and Applied Mathematics*. Marcel Dekker, Inc., 1981.
- [Robb 1985a]: Richard A. Robb (redaktør). *Three-Dimensional Biomedical Imaging*, bind II. CRC Press, 1985.
- [Robb 1985b]: Richard A. Robb (redaktør). *Three-Dimensional Biomedical Imaging*, bind I. CRC Press, 1985.
- [Robb 1985c]: Richard A. Robb. X-ray-computed tomography: Advanced systems and applications in biomedical research and diagnosis. I [Robb 1985b], kapitel 5, siderne 107–164.
- [Robb 1985d]: Richard A. Robb. X-ray-computed tomography: Implementation and applications. I [Robb 1985b], kapitel 4, siderne 81–106.
- [Robb og Barillot 1989]: R. A. Robb og C. Barillot. Interactive display and analysis of 3-d medical images. *IEEE Transactions on Medical Imaging*, 8(3):217–226, september 1989.
- [Robinson og Greenleaf 1985]: B. S. Robinson og J. F. Greenleaf. Computerized ultrasound tomography. I [Robb 1985a], kapitel 3, siderne 57–78.
- [Rosenfeld og Kak 1982]: Azriel Rosenfeld og Avinash C. Kak. *Digital Picture Processing*, bind 2 af *Computer Science and Applied Mathematics*. Academic Press, 1982.
- [Rourke og Sanderson 1982]: Colin P. Rourke og Brian J. Sanderson. *Introduction to Piecewise-Linear Topology*. Springer-Verlag, 1982.
- [Rowland 1979]: S. W. Rowland. Computer implementations of image reconstruction formulas. I [Herman 1979], kapitel 2, siderne 9–79.
- [Ruud 1992]: Nils Aksel Ruud. Pasienttilpassing av hofteproteser. I [DTI 1992].
- [Sabella 1988]: Paolo Sabella. A rendering algorithm for visualizing 3D scalar fields. *Computer Graphics*, 22(4):51–58, august 1988.
- [Schmidt 1993]: Michael F. W. Schmidt. Cutting cubes — visualizing implicit surfaces by adaptive polygonization. *Visual Computer*, 10(2):101–115, 1993.
- [Schomberg 1981]: Hermann Schomberg. Nonlinear image reconstruction from projections of ultrasonic travel times and electric current densities. I [Herman og Natterer 1981], siderne 270–291.

- [Schroeder et al. 1992]: William J. Schroeder, Jonathan A. Zarge, og William E. Lorensen. Decimation of triangle meshes. *Computer Graphics*, 26(2):65–70, juli 1992.
- [Shepp 1983a]: Lawrence A. Shepp (redaktør). *Computed Tomography*, bind 27 af *Proceedings of Symposia in Applied Mathematics*, Providence, Rhode Island, 1983. American Mathematical Society.
- [Shepp 1983b]: Lawrence A. Shepp. Scope of pure and applied tomography. I [Shepp 1983a], siderne 1–6.
- [Shu og Krueger 1991]: Renben Shu og Richard C. Krueger. A 3D surface construction algorithm for volume data. I N. M. Patrikalakis (redaktør), *Scientific Visualization of Physical Phenomena*, siderne 251–266. Computer Graphics Society, Springer Verlag, 1991.
- [Siemens 1991]: Siemens, Erlangen. *Somatom Plus: Case Studies*, 1991.
- [Smith 1983]: Kennan T. Smith. Reconstruction formulas in computed tomography. I [Shepp 1983a], siderne 7–23.
- [Smith et al. 1977]: Kennan T. Smith, Donald C. Solmon, og Sheldon L. Wagner. Practical and mathematical aspects of the problem of reconstructing objects from radiographs. *Bulletin on the American Mathematical Society*, 83(6):1227–1270, 1977.
- [Steenstrup 1990]: S. Steenstrup. Reconstructing SPECT images using the maximum entropy principle and including the effect of collimation. *Physica Medica*, VI(3–4):273–276, 1990.
- [Swaelens 1992]: Bart Swaelens. Rapid prototyping of medical models for CT-images with stereolithography. I [DTI 1992].
- [Ter-Pogossian 1985]: Michel M. Ter-Pogossian. Positron emission tomography (PET). I [Robb 1985a], kapitel 2, siderne 41–55.
- [Tretiak 1983]: Oleh J. Tretiak. Attenuated and exponential Radon transforms. I [Shepp 1983a], siderne 25–33.
- [Turk 1992]: Greg Turk. Re-tiling polygonal surfaces. *Computer Graphics*, 26(2):55–64, juli 1992.
- [Upson og Keeler 1988]: Craig Upson og Michael Keeler. V-BUFFER: Visible volume rendering. *Computer Graphics*, 22(4):59–64, august 1988.
- [Wilhelms og Van Gelder 1990a]: Jane Wilhelms og Allen Van Gelder. Octrees for faster isosurface generation: Extended abstract. *Computer Graphics*, 24(5):57–62, november 1990.
- [Wilhelms og Van Gelder 1990b]: Jane Wilhelms og Allen Van Gelder. Topological considerations in isosurface generation: Extended abstract. *Computer Graphics*, 24(5):79–82, november 1990.
- [Wilhelms og Van Gelder 1991]: Jane Wilhelms og Allen Van Gelder. A coherent projection approach for direct volume rendering. *Computer Graphics*, 25(4):275–284, juli 1991.
- [Wood et al. 1979]: E. H. Wood, J. H. Kinsey, R. A. Robb, B. K. Gilbert, L. D. Harris, og E. L. Ritman. Applications of high temporal resolution, computerized tomography to physiology and medicine. I [Herman 1979], kapitel 6, siderne 247–279.
- [Wyvill et al. 1986]: Geoff Wyvill, Craig McPheeters, og Brian Wyvill. Data structure for soft objects. *Visual Computer*, 2:227–234, 1986.
- [Xu og Lu 1988]: S. B. Xu og W. X. Lu. Surface reconstruction of 3D objects in computerized tomography. *Computer Vision, Graphics, and Image Processing*, 44(3):270–278, december 1988.
- [Yagel et al. 1991]: Roni Yagel, Daniel Cohen, og Arie Kaufman. Context sensitive normal estimation for volume imaging. I N. M. Patrikalakis (redaktør), *Scientific Visualization of Physical Phenomena*, siderne 211–234. Computer Graphics Society, Springer Verlag, 1991.
- [Yale 1968]: Paul B. Yale. *Geometry and Symmetry*. Holden-Day Series in Mathematics. Holden-Day, 1968.
- [Zeitler 1974]: E. Zeitler. The reconstruction of objects from their projections. *Optik*, 39(4):396–415, 1974.

Indeks

Sidenumre sat med **fed** skrift henviser til definition af opslaget.

A

A-billeder 34
 adjungering
 af celle 136
 af transformation 14
 adskillende alternativ **82**, 83
 afstand mellem tomogrammer ... *se* snitafstand
 afstands-gradient toning 45
 afstandstoning 45
 aktiv tomografi 28
 α -blanding *se* transparente voxels
 algebraic reconstruction technique 20
 algebraisk topologi *se* topologi, algebraisk
 Ambartsumian, V. A. 36
 analytiske felter *se* implicit flade
 ANALYZE 3, 42
 anatomi 25, 29
 animation 41
 ART *se* algebraic reconstruction technique
 assymetri mellem høj og lav *se* facet,
 udartning

B

$\mathbb{B}^n$ *se* kugle
 B-billeder 34
 baglæns differens *se* gradient, baglæns
 differens
 bane 70, 85
 barycenter *se* facet, barycenter
 billede *se* digitalt billede
 billedpunkt 23, 28, 39
 omegn 54
 billedtransmission 38
 biokemiske processer *se* funktionalitet af
 organer
 bløddede 25, 31, 99
 Bracewell, R. N. 21
 brain scanner *se* EMI brain scanner
 bånd begrænsning 18
 bølgeligning 35
 bølgeudbredelse 37

C

CAT *se* computer assisted tomography
 celle **56**, 57
 siderelation *se* siderelation
 cellekompleks 55, **56**
 central differens *se* gradient, central
 differens

Compton spredning *se* spredningsforsøg
 computed tomography 25
 computer assisted tomography 25
 computerized axial tomography 25
 convolution *se* foldning
 Cormack, Allan Meeleod 10, 21
 cosinus lov 45
 CT *se* computed tomography *eller*
 røntgentomografi
 CT-enheder *se* Hounsfield enheder
 cutting cubes 51
 CW-kompleks 56
 cyclotron 31
 cyklisk gruppe 70

D

data-rum 140
 degenereret facet *se* facet, udartning
 degenereret skæringspunkt ... *se* skæringspunkt,
 udartning
 del-rumfangs effekt 28, 45
 Delaunay triangulering 48, 52, 125
 depth shading *se* afstandstoning
 depth-gradient shading ... *se* afstands-gradient
 toning
 diagonal **82**
 differentialkvotient *se* gradient
 diffraktionstomografi 35
 digital topologi *se* topologi, digital
 digitalt billede 5, 23, **59**
 dimension
 billede **59**
 celle **56**
 cellekompleks **56**
 kubisk gitter **57**
 DSR-skanner 27
 DT *se* diffraktionstomografi
 dynamisk billeddannelse 29

E

ECT *se* emission computed tomography
 ekko *se* spin-ekko
 eksakt sekvens 137
 eksponentiel Radon transformation .. *se* Radon
 transformation eksponentiel
 elastostatik 38
 elektronmikroskopi 37
 elektrostatik 38
 EMI brain scanner 11, 20, 27
 emission computed tomography 28

emissionstomografi 25, 28–29, 36
 kombineret med CT eller MRI . . *se* kombination af skanneformer
 empiriske data 6, 113
 estimationsmetoden 67, 74, 82, 126, 129
 implementering 76–77, 92
 Explorer 96

F

facet
 0-facet 63
 1-facet 64
 2-facet 79
 3-facet 130–131
 barycenter 104, 114, 131
 mål 104, 131
n-facet 136
 orientering 64, 79, 89, 91, 100, 135–137
 repræsentation 92–94, 131
 symbolsk 73, 86, 89–92
 triangulering 100–106, 131, 137
 udartning 64, 95
 fare ved skanning *se* risiko
 filter respons-funktion 16–18
 filtreret tilbageprojektion
 diskret 18
 todimensional 16
 tredimensional 17, 34
 flad toning 107
 flade-sporing 49–50
 flertydighed 65–67, 81, 82, 92, 138
 afgørelse 67, 74, 82, 129
 triangulering af tegl 126
 årsag til inkonsistens 80
 foldning 59
 forbindende alternativ 82
 forenkling af flade 50, 112–113
 nedefra og op 50, 113–116
 oppefra og ned 50–52, 113
 forgreningsproblemet 48
 forlæns differens *se* gradient, forlæns differens
 fotonenergi
 CT 26
 PET 30
 SPECT 29
 Fourier transformation 12–13, 14
 funktional 12
 funktionalitet af organer 25, 29, 30

G

gantry 23
 CT 26
 PET 30
 SPECT 30

Gauß kerne *se* kerne, Gauß
 generation af skanner . . . *se* skanner generation
 generic patch 105
 geometrisk normal 107
 gitter *se* kubisk gitter
 Gouraud toning 107
 gradient
 baglæns differens 109
 central differens 109
 foldning 110–111
 forlæns differens 109
 ikke-regulær koordinatfunktion . . 109–110
 interpolation 108
 kontekstafhængig metode 111
 matematisk 108
 regulær koordinatfunktion 109
 sammenhæng med normal 45, 107
 Sobel kerne *se* kerne, Sobel
 gradient normal 108, 112
 gradientfelt 33
 gravitet og ultralyd 34
 gravitation 122–124
 gruppeteori 60, 69–72, 83–85
 gruppevirkning 70, 75, 85

H

halv-gennemsigtige voxels *se* transparente voxels
 hash tabel 94
 hastighedsfordeling af stjerner . . . *se* stjerners hastighedsfordeling
 helisk skanner *se* spiral skanner
 hierarkisk rumfangsgengivelse 47
 hjørne 61, 63, 78
 højt 62
 lavt 62
 mængden af høje hjørner 69
 nummer *se* nummerering af hjørner
 holografi *se* interferometri
 homeomorfi 55
 PL 137
 Hounsfield enheder 26, 40
 Hounsfield, Godfrey Newbold 10, 21
 HU *se* Hounsfield enheder
 hyperplan 133
 hærdning 27
 høj og lav assymetri *se* facet, udartning
 høje hjørner *se* hjørne, højt
 højpas filtrering 16
 højrehåndsregel *se* tommelfingerregel
 højt sammenhængende alternativ . . 66, 76, 80, 88

I

identisk koordinatfunktion 59

ikke-invasiv 10, 23, 34, 38
 implicit flade 6, 51, 113
 indeks for normal 112
 indeks for skæringspunkt ... *se* skæringspunkt,
 indeks
 Indigo² 95
 inkonsistens 80
 marching cubes *se* marching cubes
 interferometri 37
 interpolation af konturer 48
 interpolation mellem tomogrammer ... 44, 140
 Inventor 96
 ioniserende stråling 25, 28, 35
 energi *se* fotonenergi
 Iris 4D/20 95
 isotoper 29–31
 isotropigruppe 71
 isotropisk algoritme 47, 49
 iterativ metode 18–20

K

kant 61, 63, 78
 kritisk **63**
 nummer *se* nummerering af kanter
 kantdetektion 40, 108
 karakteristisk afbildning 56, 57
 kemiske processer *se* funktionalitet af
 organer
 kerne
 Gauß 119
 repræsentation 60
 Sobel 110
 kernemagnetisk resonans 25, 31–34, 36
 kombineret med emissionstomografi
 *se* kombination af skanneformer
 kernespin 32
 kollimation
 CT 26, 27
 PET 30
 SPECT 30
 kombination af skanneformer 28
 kombinatorisk topologi *se* topologi,
 algebraisk
 konsekvent metode **67**, 74, 92
 konsistens **65**, 80, 86
 kontur *se* konturflade *eller* konturkurve
 konturflade **40–41**, 78
 konstruktion 47–52
 krumning *se* krumning
 lukning 105
 toning *se* toning, konturflade
 konturkurve 40, 48, 62
 interpolation *se* interpolation af
 konturer
 koordinatfunktion **58**

korona 37
 kritisk kant *se* kant, kritisk
 kritisk tegl *se* tegl, kritisk
 krumning 114–115
 kubisk gitter **57**
 rand 57
 kubiske voxels *se* interpolation mellem
 tomogrammer
 kugle **56**
 kurvebaseret algoritme 47–48, 134
 kvadratisk integrabel 12
 kvalitet af skannebillede *se* skannebillede,
 kvalitet

L

Laplace operator 38
 Larmor frekvens 32
 lav og høj assymetri *se* facet, udartning
 lave hjørner *se* hjørne, lavt
 lavineprognose 38
 lavpas filtrering 16, 118–120, 140
 lavt sammenhængende alternativ ... **66**, 74, 76,
 80, 88
 ligevægt *se* termisk ligevægt
 lineær interpolation af skæringspunkt ... *se* skæ-
 ringspunkt, estimation
 lineær transportteori *se* transportteori
 linieintegral ... 11, 12, 18, 23, 26, 27, 30, 35,
 36
 longitudinal relaksation *se* relaksation
 luftforurening 38
 lukning af kontur *se* konturflade, lukning

M

mangfoldighed 131, 132
 marching cubes 49–50, 80, 128–129
 matematisk spredningsteori ... *se* spredningsteori
 materialekontrol 38
 medicinsk skanning 5, 22
 metabolisme *se* funktionalitet af organer
 mikrobølgespredning 37
 MRI *se* kernemagnetisk resonans
 mål *se* facet, mål
 mængde som datastruktur 8
 mønstergenkendelse 38

N

n-topologi *se* topologi, *n*-topologi
 neutrinoer 38
 NMR-spektroskopi 31, 33
 nobelpris 10, 20–22
 normal
 estimation 107–108
 orientering 112
 trappeeffekten 121
 ækvipotentialflade 45

notation 8
nuclear magnetic resonance spectroscopy
..... *se* NMR-spektroskopi
nuclear som „farligt“ ord 31
0-topologi *se* topologi, 0-topologi
nummerering af hjørner 68, 83
nummerering af kanter 73, 86
Nyquists sætning 140
næsten overalt 12

O

objekt-rum 140
operator 12
operator metode 18
opløsning af skannebillede .. *se* skannebillede,
opløsning
opslagstabel 74–76, 86
organers funktionalitet *se* funktionalitet af
organer
orientering
facet *se* facet, orientering
normal *se* normal, orientering
standard- 135

P

partial volume effect .. *se* del-rumfangs effekt
partielle differentiaalligninger 38
partikelsystemer 46–47
passiv tomografi 28
Perl 116
PET *se* proton emission tomography
Phong toning 107
picture element *se* billedpunkt
PIXAR 46
pixel *se* billedpunkt
PL-homeomorfi *se* homeomorfi, PL
planintegral 16, 33
plasmafysik 38
Polya-Burnsides formel 70
polære koordinater 14, 15
prismetret 50, 113
projektion 132
projektionssætning
todimensional 15
tredimensional 17
proton emission tomography ... 25, 30–31, 36
pseudokode 8

R

radioastronomi 21, 36
Radon rum
todimensionalt 13
tredimensionalt 16, 33

Radon transformation 36
eksponentiel 28, 36
invers todimensional 15
invers tredimensional 17
todimensional 13
tredimensional 17
Radon, Johann 11, 20, 21
raketter 38
rand af kubisk gitter ... *se* kubisk gitter, rand
Reality Engine² 41
reconstructive tomography 25
regulær celle 56
regulær koordinatfunktion 58, 140
rekursion *se* tegl-algoritme, rekursion
relaksation 33
resonansfrekvens 32, 33
rettriangulering 50
risiko
CT 25
emissionstomografi 25, 28
MRI 31
ultralyd 34
RT *se* reconstructive tomography
rumfang 39
„typisk“ 40, 95
rumfangsgengivelse 41–47
hierarkisk *se* hierarkisk
rumfangsgengivelse
partikelsystemer *se* partikelsystemer
toning *se* toning, rumfangsgengivelse
røntgenabsorption 11, 27
røntgenbillede 22, 23
røntgenstråling 10, 25, 38
energi *se* fotonenergi
røntgentomografi 10, 12, 23, 25–28, 36
kombineret med emissionstomografi
..... *se* kombination af skanneformer
røntgentomogram 23

S

$\mathbb{S}^n$ *se* sfære
sammenhængsgraf 54
sammenhængsparadokset 54, 60
sand tid
CT 27
MRI 34
ultralyd 34
scintigrafi 22
seismologi 38
sekvens som datastruktur 8
sfære 56
shading *se* toning
side 78
siderelation 57

signal-støj forhold
 CT 28
 emissionstomografi 28
 filteret tilbageprojektion 18
 MRI 33
 PET 31
 SPECT 30
 ultralyd 34
 signalteori 16, 140
 signatur 69
 simpliciale tegl 52
 simulerede data 6
 single photon emission computed tomography ..
 25, 29–30, 36
 singulær facet *se* facet, udartning
 singulært skæringspunkt *se* skæringspunkt,
 udartning
 skadevirkning *se* risiko
 skalarfelt 45, 61
 skalarum 134–135
 skankonvertering
 trekant 107
 voxel *se* rumfangsgengivelse
 skannebillede 23
 kvalitet 25
 opløsning 24, 26
 skanneformer 25
 skanner 23
 DSR *se* DSR-skanner
 generation 27
 helisk *se* spiral skanner
 risiko *se* risiko
 spiral *se* spiral skanner
 skannetid
 CT 26
 MRI 33, 34
 skanning *se* medicinsk skanning
 skelet 56
 Skylab 37
 skæringspunkt 63
 estimation 63, 74, 94–95, 124
 indeks 94–95
 repræsentation 73
 udartning 95
 snedybder 38
 snitafstand 26, 39
 snittykkelse 27
 Sobel kerne *se* kerne, Sobel
 solens korona *se* korona
 SPECT .. *se* single photon emission computed
 tomography
 spin *se* kernespin
 spin-ekko 33
 spin-flip 32
 spin-gitter relaksation *se* relaksation

spin-spin relaksation *se* relaksation
 spiral skanner 27
 splat *se* hierarkisk rumfangsgengivelse
 splitting-box 51, 105, 113
 spredningsforsøg 37
 spredningsteori 38
 stabilisator *se* isotropigruppe
 standardorientering ... *se* orientering, standard-
 statistisk billedannelse 29
 statistik 38
 stereolitografi 42, 105, 121
 stjerners hastighedsfordeling 36
 stressanalyse 38
 stykkevis lineær homeomorfi .. *se* homeomorfi,
 PL
 stærk ækvivalens *se* ækvivalens, stærk
 størrelsesorden 8
 svag ækvivalens *se* ækvivalens, svag
 svækkelseskoefficient 11, 26, 27
 symbolsk facet *se* facet, symbolsk
 symmetrier af kubus 85
 symmetrier af tegl *se* tegl, symmetri
 syntetiske data *se* implicit flade

T

T_1 *se* relaksation
 T_1 -vægtning *se* vægtning
 T_2 *se* relaksation
 T_2 -vægtning *se* vægtning
 TCT ... *se* transmission computed tomography
 tegl 62–63, 78, 130
 kritisk 63
 nummerering af hjørner .. *se* nummerering
 af hjørner
 orientering 89, 91, 137–138
 repræsentation af 1-tegl 73
 repræsentation af 2-tegl 68
 symmetri 70, 83–84, 128–129
 triangulering 125–128
 tegl-algoritme 49
 generel 137–138
 n -topologi 124
 rekursion 89, 136
 simplicial 125
 todimensional 73
 tredimensional 86, 95–96
 Terman 21
 terminologi 8
 termisk ligevægt 33
 tid for skanning *se* skannetid
 tidsvarierende data 131
 tilbageprojektion
 todimensional 14, 16
 tredimensional 17
 time of flight analysis 38

todimensional tegl-algoritme
 *se* tegl-algoritme, todimensional
 tommelfingerregel 79, 89, 135
 tomografi 6, 22
 første anvendelse 36
 tomogram 10, 22
 toning
 konturflade 107
 rumfangsgengivelse 45
 topologi
 algebraisk 54, 60, 137, 139
 digital 54–55, 60
 komplementaritet 59
 n-topologi 58, 122–124
 0-topologi 58, 122, 124
 transformation 12
 diskret 18
 transmission computed tomography 28
 transparente voxels 43, 45–46, 141
 transportteori 38
 transversal 71–72, 74, 85
 transversal relaksation *se* relaksation
 trappeeffekten 117–122
 triangulering af facetter *se* facet,
 triangulering
 tværsnitsbillede *se* tomogram
 tykkelse af snit *se* snittykkelse
 tærskelværdi 40, 43, 48, 61, 78

U

udartet facet *se* facet, udartning

udartet skæringspunkt *se* skæringspunkt,
 udartning
 ultralydstomografi 25, 34–36

V

V-BUFFER 46
 volume *se* rumfang
 volume element *se* voxel
 volume rendering *se* rumfangsgengivelse
 voxel 28, 39
 kubisk *se* interpolation mellem
 tomogrammer
 transparent *se* transparente voxels
 vægtning 33
 værdifunktion 57, 58

W

weaving wall 49
 winged-edge 94

X

X-ray computed tomography 25
 XCT *se* X-ray computed tomography

Æ

ækvipotentialflade 5, 40, 41, 78
 ækvipotentialkurve 61
 ækvivalens
 stærk 68, 69, 83
 svag 68, 70, 74, 83, 88