

Aard : afstuderverslag

Omvang : 50 blz.

Datum : november 1985

Vakgroep : Transmissie van Informatie
Schrijver : H.M. Wielandt
Titel : Onderzoek naar datareductie van
radarbeelden ten behoeve van het
site diversity onderzoek

Korte inhoud :

Het afbeelden van radardata op een monitor is een onderdeel van het rondzoekradarsysteem, dat in de nabije toekomst gegevens zal verschaffen ter bepaling van de intensiteit en geometrie van regenbuien in de omgeving van Delft. Deze data dient tevens on-line te worden verzonden naar het DNL en te worden opgeslagen voor latere off-line verwerking. Ten gevolge van de beperkte bandbreedte van het transmissiekanaal en de geheugencapaciteit is datareductie noodzakelijk. Om onder het product van beschikbare transmissietijd en kanaalcapaciteit te blijven, is een algoritme geschreven dat het beeld aan de zenzijde analyseert en beschrijft met behulp van een beperkt aantal parameters. Aan de ontvangzijde wordt uit deze parameters het beeld weer gereconstrueerd. Het is een mogelijkheid om, alleen voor het diversity onderzoek significante, informatie beschikbaar te laten komen.

Hoogleraar Prof.ir. J.L. de Kroes

Mentoren ir. L.R. Nieuwkerk, ir. J.S. van Sintruyen

INHOUDSOPGAVE

<u>Inleiding</u>	2
1. <u>Systeemopbouw</u>	
1.1 Inleiding	4
1.2 Het radarsysteem	5
1.3 Het communicatiesysteem	7
2. <u>Het beschrijven van het beeld met behulp van een aantal parameters</u>	
2.1 Inleiding	9
2.2 De te verwachten beelden	9
2.3 Het beschrijven van het beeld	12
2.3.1 Het beschrijven van het beeld zonder vervorming	13
2.3.2 Het beschrijven van het beeld met vervorming	13
2.3.3 Vergelijking tussen de verschillende coderingstechnieken	15
2.4 Het verkrijgen van de contourpunten	16
2.5 Het vinden van contouren	19
2.6 Het volgen van contouren	23
3. <u>Gerealiseerde beeldverwerkingsalgorithmen</u>	
3.1 Inleiding	30
3.2 Het markeren van de contourpunten	30
3.3 Het bepalen van het zwaartepunt van een (deel-) plateau	32
3.4 De structuurdiagrammen	36
4. <u>Aspecten betreffende de communicatie tussen de grondstations</u>	45
5. <u>Conclusies</u>	48
<u>Literatuurlijst</u>	49
<u>Bijlage: Listings</u>	

Inleiding

De troposfeer heeft bij straal- en satellietverbindingen een grote invloed op de transmissiekwaliteit en zal hierom bij het ontwerp van een verbinding betrokken moeten worden.

Wordt voor frequenties tot 10 GHz bij straalverbindingen veel hinder ondervonden van meerwegfading, voor frequenties boven 10 GHz zal de kwaliteit belangrijk worden beïnvloed door extra demping.

De extra demping wordt veroorzaakt door de in de troposfeer aanwezige gassen en (water-) deeltjes. Van de hydrometeoren is de invloed van regen het grootst ten gevolge van de grotere equivalente waterhoeveelheid en het afwijkend dielectrisch gedrag. De demping ten gevolge van gassen is, voor de in de naaste toekomst in gebruik te nemen 20-30 GHz frequentieband, gering vergeleken met de demping ten gevolge van regen.

Naast extra demping treedt ook kruispolarisatie-ontkoppeling op ten gevolge van de niet-bolsymmetrische regendruppels die door het vallen enigszins worden afgeplat. Bij gebruik van twee orthogonale polarisatierichtingen treedt dan overspraak op.

De extra demping wordt bij straalverbindingen gecompenseerd door het kleiner maken van de trajectlengtes waardoor tevens de meerwegfading relatief aan invloed inboet. Bij satellietverbindingen kan verbetering worden bereikt door toepassing van "site-diversity". Bij site-diversity wordt gebruik gemaakt van twee of meer grondstations die op zodanige afstand van elkaar worden geplaatst dat de kans gering is dat de grondstations tegelijkertijd onbruikbaar worden. De kans dat de stations tegelijkertijd onbruikbaar zijn wordt hierdoor verminderd.

De extra demping op een satellietpad kan met behulp van een radar bepaald worden, daar demping en reflectivity aan elkaar gerelateerd zijn.

Om meer inzicht te krijgen betreffende de fenomenen die zich in de troposfeer afspelen en om tot een beter model van de troposfeer te komen, vindt er op de Technische Hogeschool radiopropagatieonderzoek plaats. Hierbij wordt ten behoeve van een 20-30 GHz diversity testconfiguratie, in samenwerking met de PTT/Dr. Neher Laboratorium (DNL), een rondzoekradar ontwikkeld die in de omgeving van

satellietbakenontvangers de geometrische structuren van de intensiteiten in regenbuien dient te bepalen. Door variërende meteorologische omstandigheden kan de reflectivity, in de tijd gezien, snel veranderen, wat gepaard gaat met een overeenkomstig snelle dempingverandering. De rondzoekradar zal om deze reden permanent in bedrijf moeten zijn.

De gegevens die hierbij worden verkregen, dienen on-line te worden verzonden naar het DNL en opgeslagen te worden voor latere off-line verwerking. Ten gevolge van de beperkte bandbreedte van het transmissiekanaal en opslagruimte zal datareductie nodig zijn, zodat alleen voor het diversity onderzoek significante informatie beschikbaar komt.

Datareductie wordt, naast beeldcoderingsmethoden die steeds de intensiteit van alle beeldpunten --direct of via transformaties--codeert, bereikt door het beeld aan de zenzijde te analyseren en te beschrijven met behulp van een aantal parameters. Deze parameters worden vervolgens gecodeerd, waarna aan de ontvangtzijde vanuit deze parameters het beeld weer wordt opgebouwd. Een methode om een regenbui van gelijke intensiteit vereenvoudigd weer te geven, is bijvoorbeeld door middel van een cirkel waarbij de oppervlakte van de cirkel de uitgebreidheid van de bui weergeeft. Er hoeven dan slechts vier parameters overgezonden te worden, namelijk: de intensiteit, de plaatscoördinaten van het zwaartepunt en de straal van de cirkel. Om de vorm beter weer te kunnen geven, zullen er meer parameters nodig zijn. Hierbij moet tevens rekening gehouden worden met de volgende aspecten:

- de verwerkingstijd
- het beperkt houden van het aantal parameters
- de beschikbare geheugenruimte van de computer
- de beschikbare apparatuur

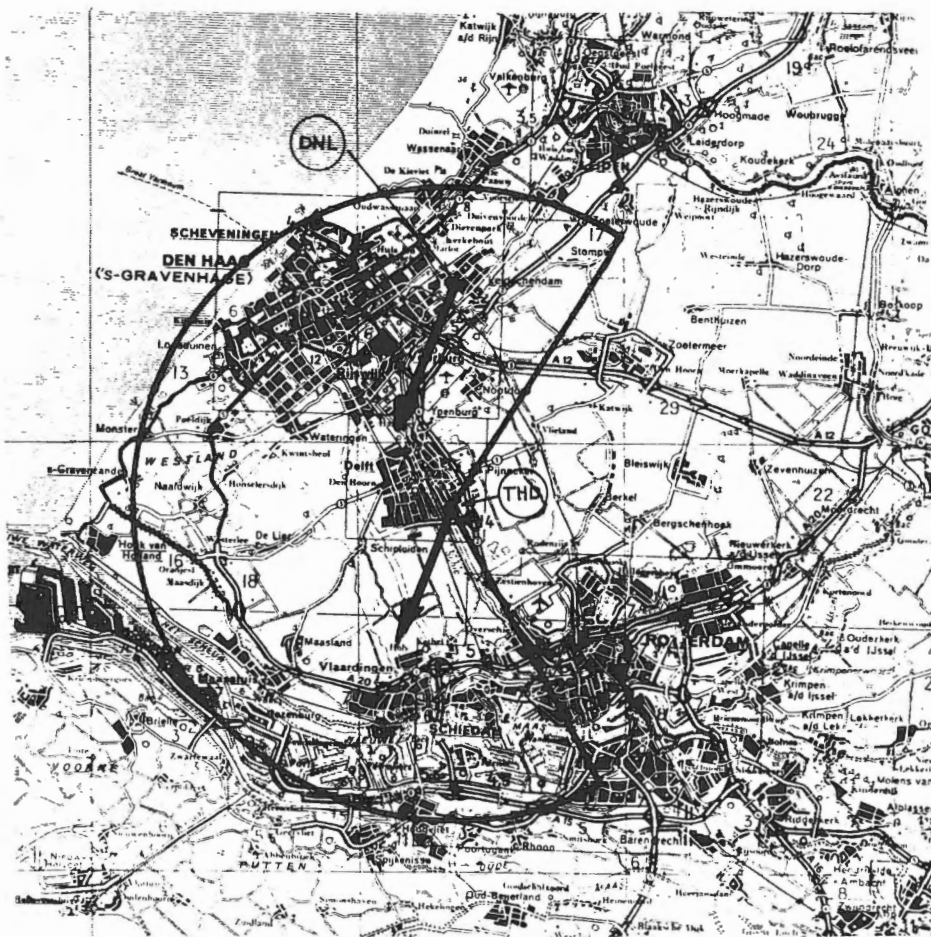
Voor het weergeven van de beelden kan gebruik worden gemaakt van een zwart-wit monitor waarbij maximaal 16 (0-15) intensiteiten onderscheiden kunnen worden.

1. Systeemopbouw

1.1 Inleiding

Om de geometrische structuur van de intensiteit van regenbuien te bepalen, dient de rondzoekradar een geschikt gedeelte van de omgeving, die de ontvangst van de satellietbakenontvangers bepalen, te bestrijken.

Een satellietbakenontvanger wordt geplaatst bij het DNL, terwijl een tweede satellietbakenontvanger alsmede de rondzoekradar geplaatst worden bij de THD. De gegevens afkomstig van de rondzoekradar alsmede de Delftse satellietbakenontvanger worden on-line verzonden naar het DNL. In figuur 1.1.1 is een kaart van de omgeving van Leidschendam-Delft weergegeven waarbij tevens het afstands bereik en de bestreken sectorhoek van de radar alsmede de ontvangrichting van de satellietbakens zijn aangegeven. Deze keuze maakt het mogelijk de belangrijke regenbuien komend uit het Zuid-Westen en het Noord-Westen te detecteren.



figuur 1.1.1

1.2 Het radarsysteem

Voor het verrichten van metingen wordt gebruik gemaakt van een FM-CW¹⁾ radar. Het principe van de FM-CW radar is dat continu een signaal wordt uitgezonden waarvan de frequentie gedurende een tijdsinterval T_m lineair met de tijd toe- of afneemt en de amplitude constant blijft. Het in de richting van het radarsysteem gereflecteerde signaal wordt, indien de benodigde ontkoppeling tussen zender en ontvanger niet gehaald wordt, door een tweede, naast de zendantenne opgestelde antenne opgevangen.

Wordt verondersteld dat de "mount", waarop de antennes geplaatst zullen zijn, stil staat en dat in de bundel van de zendantenne één reflecterend doel aanwezig is, dan zal het gereflecteerde signaal ten gevolge van de afgelegde weg met een tijdsvertraging τ worden ontvangen. De momentele zend- en ontvangsfrequenties zullen hierdoor niet aan elkaar gelijk zijn. De verschilfrequentie, ook wel beatsignaal genoemd, welke dus evenredig is met afstand radarsysteem-reflecterend doel, wordt verkregen door de signalen aan een mixer toe te voeren en de verschilfrequentiecomponent uit te filteren.

Het ontvangen vermogen kan met behulp van de radarvergelijking bepaald worden:

$$P_o = \frac{P_z \cdot G_z \cdot G_o \cdot \lambda^2}{(4\pi)^3 \cdot d^4} \cdot \sigma$$

waarin: P = vermogen

G = Gain antenne

λ = golflengte

d = afstand radarsysteem-doel

σ = radardoorsnede

De radardoorsnede, ook wel "Radar Scattering Cross Section" genoemd is: $\sigma = \eta \cdot \Delta d \cdot A$

waarin: η = reflectivity (radardoorsnede per m³)

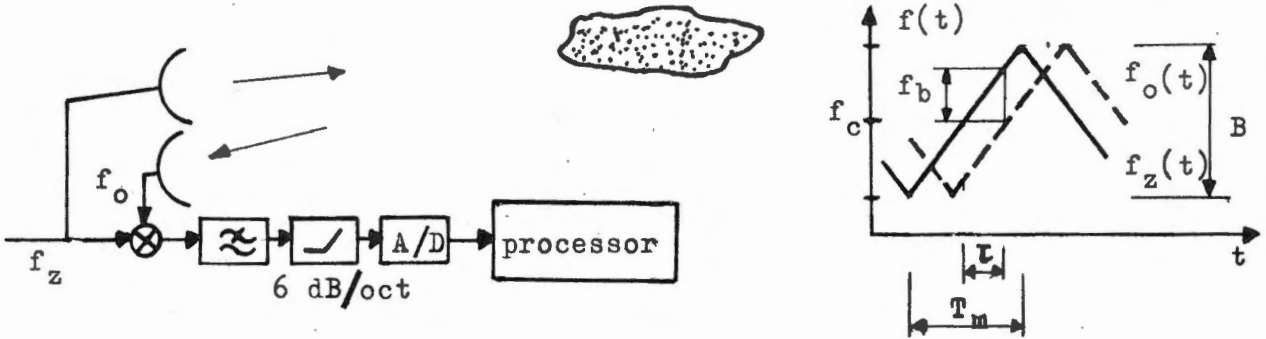
Δd = dikte van het reflecterende volume element in de richting van de antenne

A = het door de radar belichte oppervlak van het reflecterende volume element

In het algemeen zijn de troposferische volume elementen zodanig dat A evenredig is met d^2 . Ten gevolge van deze ruimtelijke uitbreiding is de amplitude van het ontvangen signaal omgekeerd evenredig met

1) Frequency Modulated-Continuous Wave

de afstand tot het reflecterende volume element. Dit wordt in de ontvanger gecompenseerd door middel van een 6 dB/oct correctie van het beatsignaal. De ontvangerruis wordt echter hiermee evenredig versterkt. Een en ander is weergegeven in figuur 1.2.1



figuur 1.2.1

De beatfrequentie is vervat in de relatie:

$$f_b = \frac{B}{T_m} \cdot \tau \quad \text{waarbij} \quad \tau = \frac{2d}{c}$$

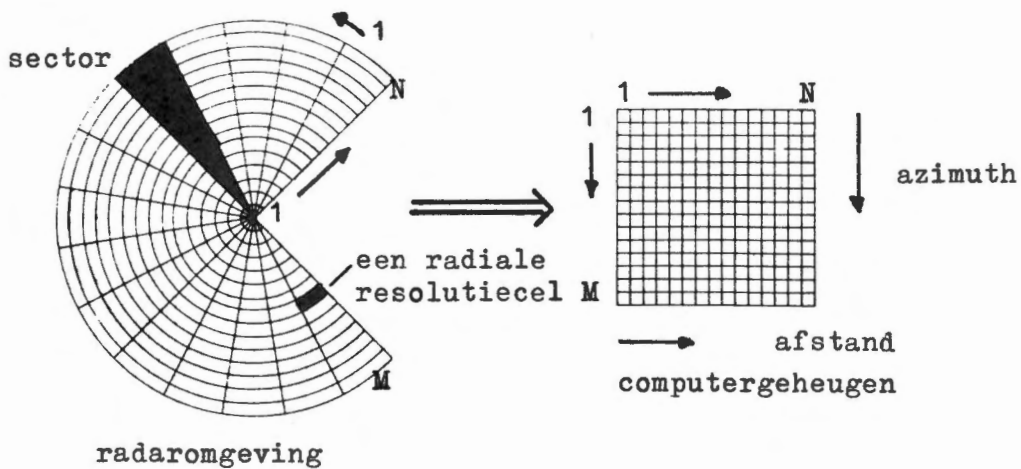
waarin: B = FM-frequentiezwaai
 c = lichtsnelheid

Het maximale afstands bereik wordt begrensd door de halve periodetijd van de driehoekige modulatie ($\tau_{\max} = T_m$). De minimale afstand is gelijk aan nul ($\tau_{\min} = 0$), wat een groot voordeel is ten opzichte van de pulsradar. Bij pulsradar is namelijk bij een monostatische opstelling $\tau_{\min} = t_{\text{pulse}}$.

Indien de reflecterende laag zich naar het radarsysteem toe beweegt dan zal, ten gevolge van het Doppler-effect, het beatfrequentieverloop veranderen. Uit het zend-en ontvangfrequentieverloop kan de Doppler-frequentie bepaald worden welke een maat is voor de snelheid van het doel.

Indien meerdere reflecties aanwezig zijn dan bestaat het beatsignaal uit de som van een aantal sinusvormige signalen. Informatie over de plaats en grootte van de reflecties wordt dan verkregen door het bepalen van het vermogensdichtheidsspectrum van het beatsignaal. Om het beatsignaal digitaal te kunnen verwerken, wordt het frequentiespectrum door middel van een Fast Fourier Transformatie (FFT), hetgeen een snel algoritme is om de Discrete Fourier Transformatie (DFT) te bepalen, met behulp van een computer bepaald.

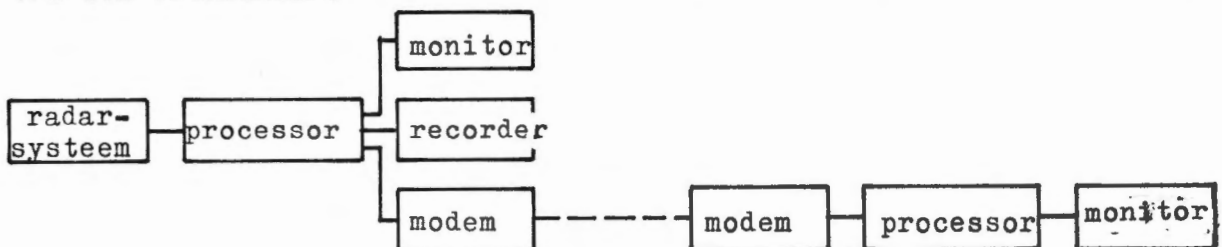
Om het discrete vermogensdichtheidsspectrum te verkrijgen wordt van de spectrale data, die per radiale resolutiecel beschikbaar komt, het vermogen berekend. Tenslotte wordt voor het weergeven de waarden gekwantiseerd tot een eindig aantal discrete niveaus. Dit levert een array-rij met N woorden op die een maat zijn voor de reflectiviteit van de overeenkomstige resolutiecellen. Indien nu de schotelantenne een pencil beam patroon heeft en de omwentelings-snelheid zo wordt gekozen dat de aangrenzende sectoren elkaar net overlappen zodat een aansluitende belichting van de omgeving wordt verkregen, dan wordt een 2-dimensionale matrix (M,N) verkregen. Figuur 1.2.2 illustreert dit. Uit de figuur zien we tevens dat er een conversie plaatsvindt die circulair-rechthoekig is.



figuur 1.2.2

1.3 Het communicatiesysteem

In figuur 1.3.1 is het communicatiesysteem weergegeven in de vorm van een blokschema.



figuur 1.3.1

Het communicatiesysteem dient de boodschap (M.N matrix), welke is afgegeven door het radarsysteem, zo goed mogelijk over te dragen naar de bestemming (on-line naar het DNL en off-line voor later verwerking). Door de beperkte bandbreedte van het transmissiekanaal (telefoonkanaal) en de opslagruimte (recorder) is het nodig een selectie toe te passen waarbij de irrelevante informatie wordt verwijderd. Er vindt dus datareductie plaats waar we in hoofdstuk 2 nog uitgebreid op ingaan. Ten einde de kanaalcapaciteit optimaal te benutten wordt naar een vermindering van de redundantie in de boodschap gestreefd (boodschapcodering). Om invloed van storingen op het transmissiekanaal te voorkomen, is het wellicht nodig de verkregen code aan te vullen, feitelijk de redundantie te vergroten, om op deze wijze de boodschap foutloos te kunnen ontvangen, ook wanneer niet elk ontvangen kanaalcodesymbool foutloos wordt gedetecteerd (kanaalcodering). De zo verkregen code dient te worden omgezet in een signaal dat door het communicatiekanaal kan worden getransporteerd (modulatie).

Het ontvangen, door ruis gestoorde signaal moet zodanig worden geïnterpreteerd dat weer de oorspronkelijke kanaalcodesymbolen worden verkregen (demodulatie). De ontvangen kanaalcodesymbolen moeten worden ontdaan van kunstmatig toegevoegde redundantie, nadat hiervan gebruik is gemaakt om opgetreden fouten te detecteren en eventueel te corrigeren (kanaaldecodering). De code dient weer te worden omgezet in de gewenste boodschap (boodschapdecodering). De informatie zal tenslotte moeten worden gebracht in een voor de bestemming geschikte vorm (boodschapreconstructie) waar we in hoofdstuk 3 nog nader op ingaan.

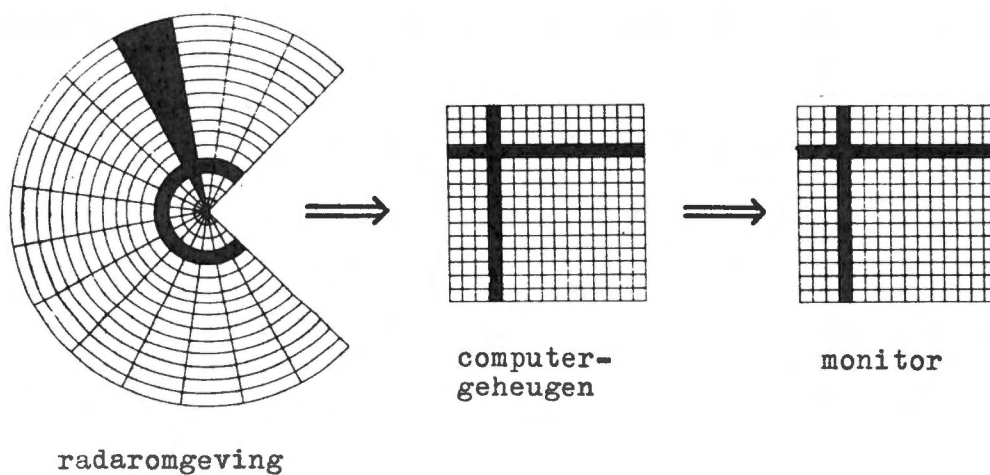
2. Het beschrijven van het beeld met behulp van een aantal parameters

2.1 Inleiding

Zoals reeds eerder vermeld is, wordt datareductie bereikt door het beeld aan de zenzijde te analyseren en te beschrijven met behulp van een beperkt aantal parameters. Door de beperkte bandbreedte van het transmissiekanaal en de beperkte opslagruimte dient m.b.v. de parameters het beeld later zo goed mogelijk gereconstrueerd te kunnen worden. In paragraaf 2.2 worden de karakteristieken van de te verwachten, uit de radar afkomstige, beelden behandeld. In paragraaf 2.3 wordt dan uit de verschillende behandelde methoden de geschikste gekozen. Uit paragraaf 2.3 blijkt dat voor het bepalen van de parameters van een plateau de contourpunten van het plateau nodig zijn. Het verkrijgen van de contourpunten wordt in paragraaf 2.4 besproken. Nadat in paragraaf 2.4 nog eerst een keuze is gemaakt, wordt in paragraaf 2.5 het vinden van plateaus behandeld. Paragraaf 2.6 bespreekt twee verschillende contour-volg-algorithmen. Tevens wordt in deze paragraaf nog een keuze gemaakt uit deze contour-volg-algorithmen.

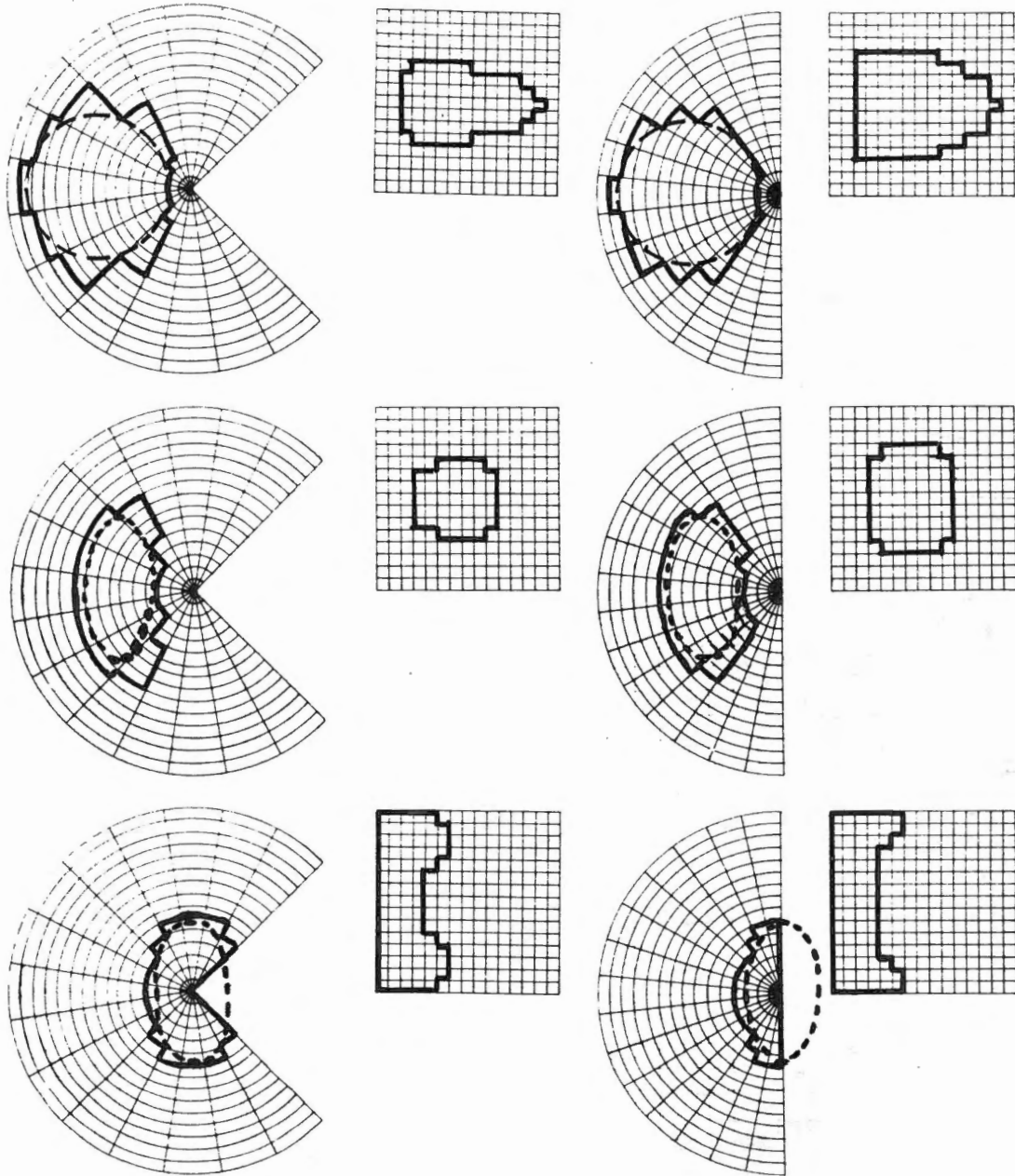
2.2 De te verwachten beelden

In het vorige hoofdstuk hebben we gezien dat er een conversie plaatsvindt van circulaire naar rechthoekige coördinaten (figuur 2.2.1).



figuur 2.2.1

Omdat nog niet bekend is hoe groot de sectorhoek wordt, geeft onderstaand figuur 2.2.2 verschillende matrixvullingen weer bij verschillende sectorhoeken. Hierbij is in de figuur aangenomen dat de reflecties (regenbuien) één intensiteit bezitten.



figuur 2.2.2

In werkelijkheid bezit de kern een hogere intensiteit. Tevens zal het beeld weergegeven op een monitor een ruisachtige indruk maken aangezien de amplitudes van de beeld-elementen grote variaties vertonen.

Het van een resolutiecel ontvangen signaal Γ_r wordt namelijk verkregen door sommatie van alle reflecties Γ_i van in de resolutiecel aanwezige deeltjes.

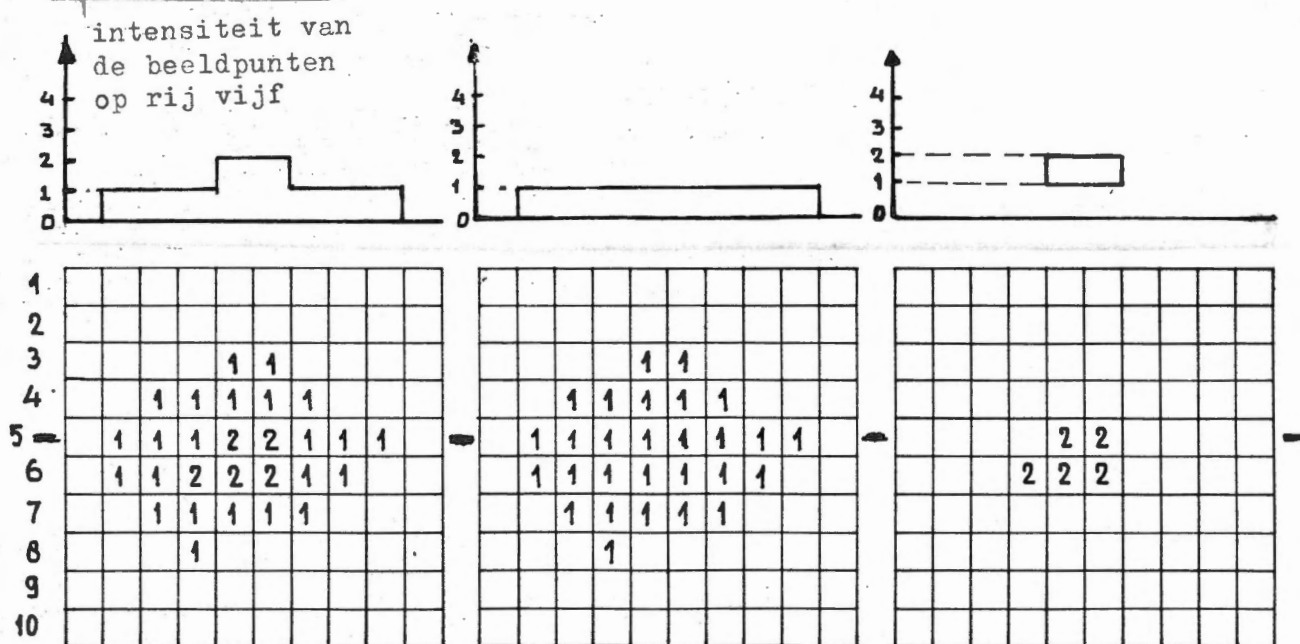
$$\Gamma_r = \sum \Gamma_i e^{j\psi_i}$$

Elk deeltje geeft dus een reflectie Γ_i met een fase draaiing ψ_i . Vanwege het grote aantal deeltjes in een resolutiecel kan worden aangenomen dat de amplitude van de reflectiecoëfficiënt van een resolutiecel ten gevolge van regen een verdeling heeft die tussen een Gaussische en een Rayleigh verdeling inligt [4]. De faseverdeling wordt uniform aangenomen. Aangezien de afmetingen van een resolutiecel veel groter zijn dan de golflengte van de draaggolf, zal de resulterende reflectiecoëfficiënt ontstaan door vectoriële optelling van de reflecties van alle deeltjes in een cel. Verbetering kan worden bereikt door de intensiteitsschaal te modificeren en een effeningsfilter toe te passen. Tevens ontstaat er bij rondraaien van de antenne onscherpte ten gevolge van de eindige afmetingen van de antennediameter. Als voorbeeld geeft een puntdoel een verticale streepje in het beeld (zie figuur 2.2.1). Voor een reconstructie van het oorspronkelijke beeld is dan kennis omtrent de antennediagram vereist. Het verbeteren van het beeld en reconstructie van het oorspronkelijke beeld valt echter buiten het bestek van mijn onderzoek.

2.3 Het beschrijven van het beeld

Het beeld kan gerepresenteerd worden door een functie $f(x,y)$ van de plaatscoördinaten x en y in het beeldvlak. Aangezien er een eindig aantal discrete niveaus zijn, kan de functie weer worden gegeven door een aantal plateaus met de plateauhoogte gelijk aan de intensiteitswaarde. Kennis van de hoogte, locatie en vorm van alle plateaus is gelijk aan kennis van het beeld.

Naast coderingsmethoden die steeds de intensiteit van alle beeldpunten -direct of via transformaties- codeert, bijvoorbeeld met pulse-code-modulatie (PCM), is het beeld aan de zenzijde te analyseren en te beschrijven met behulp van een aantal parameters. Deze parameters worden vervolgens gecodeerd, waarna aan de ontvangstzijde vanuit deze parameters het beeld weer wordt opgebouwd. Efficiënt kan het zijn punten te bepalen waar een intensiteitswisseling optreedt. Deze punten zijn uit coderingsopzicht interessant vanwege de sterke correlatie tussen deze beeldpunten en belangrijk omdat intensiteitswisselingen samenvallen met de contouren van in het beeld voorkomende doelen. Een doel wordt dan zodanig doorsneden dat er plateaus ontstaan met de eigenschap dat elk plateau overal dezelfde intensiteit heeft. Als voorbeeld heeft figuur 2.3.1a, het oorspronkelijke beeld, twee intensiteitsniveaus. Voor het beschrijven van het origineel wordt het zodanig doorsneden dat er twee schijven ontstaan: één schijf met intensiteit 1 en één (kleinere) schijf met intensiteit 2 (zie figuur 2.3.1b en 2.3.1c).



figuur 2.3.1a

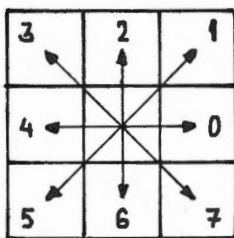
figuur 2.3.1b

figuur 2.3.1c

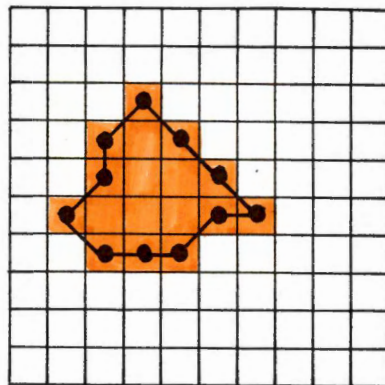
figuur 2.3.1

2.3.1 Het beschrijven van het beeld zonder vervorming

De contour van een plateau zou gecodeerd kunnen worden door van ieder contourpunt de coördinaten op te geven. Er wordt dan echter geen gebruik gemaakt van de geometrische afhankelijkheid van de contourpunten: tenminste één van de acht buurpunten van een contourpunt is wederom een contourpunt. Wanneer een eerste contourpunt bekend is, behoeft slechts de richting waarin een volgend contourpunt zich bevindt gecodeerd te worden. De acht mogelijke richtingen kunnen worden genummerd zoals aangegeven in figuur 2.3.1.1. Uitgaande van een bepaald startpunt, bijvoorbeeld het bovenste punt, is de contour van figuur 2.3.1.2 te beschrijven met de volgende zogenaamde "kettingcode": 5 6 5 7 0 0 1 0 3 3 3



figuur 2.3.1.1



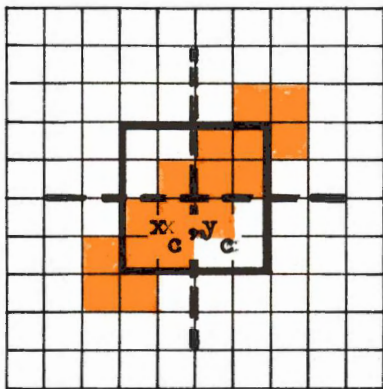
figuur 2.3.1.2

Naast de coördinaten van het startpunt vereist de kettingcode 3 bit per contourpunt om de 8 mogelijke richtingen te coderen. Echter wanneer het aantal contourpunten groot is zal het aantal bits ook groot zijn. Verlaging van het aantal bits is dan slechts te realiseren door vervorming toe te staan.

2.3.2 Het beschrijven van het beeld met vervorming

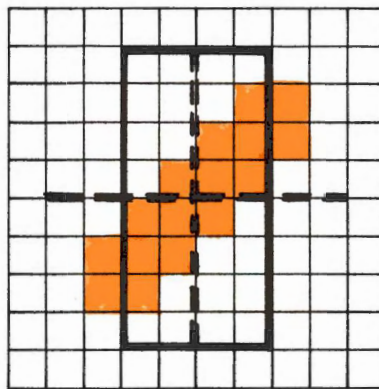
Een benadering van de vorm van een plateau kan worden verkregen door deze met een eindig aantal vierkanten weer te geven. De positie van een plateau kan daarbij gegeven worden door zijn zwaartepunt. De vorm van een plateau kan worden benaderd door een vierkant waarbij de oppervlakte van het vierkant de uitgebreidheid van het plateau weergeeft (figuur 2.3.2.1). Gekozen is voor een

vierkant en niet voor een cirkel omdat door de keuze van een matrix het rekentechnisch gemakkelijker is en voor de parameters met integers in plaats van met floating point getallen gewerkt kan worden. Er treden echter afwijkingen op van de gewenste vorm (vervorming). Om minder vervorming te verkrijgen zullen er meer gegevens nodig zijn. Het gebruik van meer parameters houdt tevens echter ook in dat de gemiddelde codewoordlengte in bits per beeldpunt, die immers de kanaalcapaciteit en geheugenruimte bepaalt, groter wordt. Rekentechnisch is het gemakkelijk een plateau op te delen in twee of vier deeloppervlakken. Er hoeft dan slechts bekeken te worden of een punt zich links of rechts en boven of onder het zwaartepunt van het gehele plateau bevindt (figuur 2.3.2.2 en 2.3.2.3).

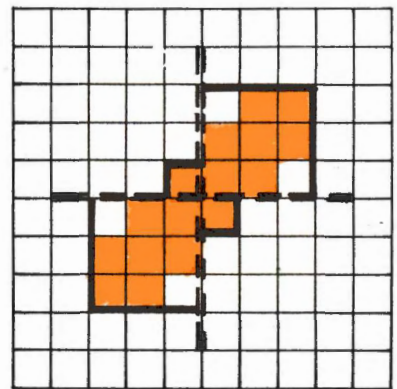


x_c, y_c = zwaartepunt van het plateau.

figuur 2.3.2.1



figuur 2.3.2.2



figuur 2.3.2.3

Het zwaartepunt van een vlakke figuur met oppervlakte m welke is opgebouwd uit n vierkante elementen wordt in cartesische coördinaten geschreven als:

$$x_c = \frac{\sum_{i=1}^n m_i \cdot x_i}{m}$$

$$y_c = \frac{\sum_{i=1}^n m_i \cdot y_i}{m}$$

waarbij:

n = totaal aantal elementen

m_i = oppervlakte van element m_i

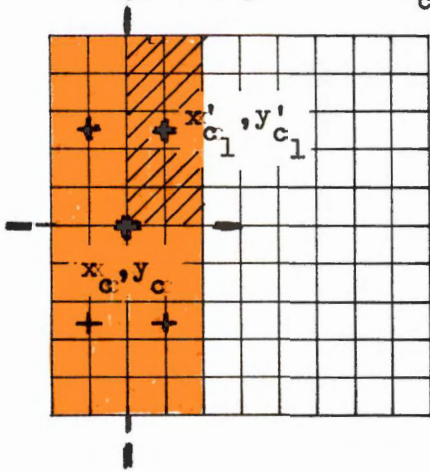
x_i, y_i plaatscoördinaten van element m_i

$m = \sum_{i=1}^n m_i$ = totale oppervlakte van de groep van n elementen

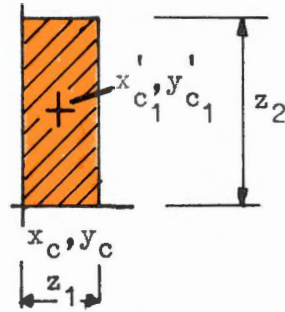
Indien we voor de oppervlakte van een element één nemen dan is m het totaal aantal elementen van een plateau.

Het nadeel om een deeloppervlak door een vierkant weer te geven is dat bij langwerpige plateaus (figuur 2.3.2.4) de vervorming groot is. Vermindering van de vervorming wordt verkregen als de samenstellende elementen door één rechthoek wordt weergegeven waarvan de zijden

worden bepaald door de oppervlakte van een deelplateau m' en de richtingscoëfficiënt r die bepaald wordt door de zwaartepunten van het gehele plateau (x_c, y_c) en van een deelplateau (x'_{c_1}, y'_{c_1}) (figuur 2.3.2.5).



figuur 2.3.2.4



figuur 2.3.2.5

$$z_1 \cdot z_2 = m' \quad (1)$$

$$(y'_{c_1} - y_c) = r \cdot (x'_{c_1} - x_c) \quad (2)$$

$$z_1 = \sqrt{\frac{m'}{r}}$$

$$z_2 = \frac{m'}{z_1}$$

De deelopervlakken kunnen pas bepaald worden wanneer het zwaartepunt van het gehele plateau bekend is. Bij het bepalen van een deelopervlak is voor het bepalen van het zwaartepunt van een deelplateau dan slechts de plaatscoördinaten van de elementen nodig.

2.3.3 Vergelijking tussen de verschillende coderingstechnieken

Een tweetal criteria zijn van belang.

- Een criterium is de grootte van de eventueel geïntroduceerde vervorming. Bij het beschrijven van het beeld met vervorming is gekozen voor een in vieren te delen plateau en de deelplateaus met rechthoeken weer te geven. Voor deze methode is gekozen omdat deze het oorspronkelijke beeld nog redelijk benadert. Per plateau hebben we dan 11 parameters namelijk: de intensiteit van het plateau, de plaatscoördinaten van het zwaartepunt en de 8 rechthoekszijden
- Een tweede criterium is de gemiddelde codewoordlengte in bits per beeldpunt die immers de vereiste kanaalcapaciteit en geheugenruimte bepaalt. Voor de efficiëntie nemen we een $M \cdot N$ matrix waarbij $M=2^x$ en $N=2^y$. Als voorbeeld wordt $x=7$ en $y=6$ genomen zodat een 128×64 matrix wordt verkregen. Om de positie van een contourpunt of het zwaartepunt aan te geven zijn er dan $7+6=13$ bits nodig. Verder nemen we aan dat er één plateau aanwezig is die het hele beeld vult en dat het beeld in twee discrete niveaus is gekwantiseerd. We krijgen

dan het maximum aantal bits voor:

PCM	128.64	= 8192 bits
coördinaten van contourpunten	$(2 \cdot 128 + 2 \cdot 62)13$	= 4940 bits
kettingcode	$13 + (2 \cdot 128 + 2 \cdot 62)3$	= 1153 bits
methode met vervorming	$13 + (7+6)4$	= 65 bits

Het is het maximum omdat de boodschap, opgebouwd uit de symbolen van het bronalfabet, nog over het kanaal moeten worden verzonden in symbolen die eigen zijn aan het kanaal. Dit betekent dat de mogelijke boodschappen van de bron moeten worden omgezet in woorden, opgebouwd uit symbolen van het kanaalalfabet (boodschapcodering). Bij dit proces wordt gestreefd naar een optimaal gebruik van het kanaal. Een efficiënte variabele lengte codering is de Huffman code [6].

2.4 Het verkrijgen van de contourpunten

Voor het bepalen van de parameters van een gebied zijn de contourpunten nodig. Een methode is om in het oorspronkelijke beeld, waarvan de gebieden expliciet en daarmee de contourpunten impliciet zijn aangegeven, de contourpunten op te sporen. Een andere methode, die expliciet de contourpunten van de gebieden opspoor, is om het oorspronkelijke beeld door een gradiënt-operator te sturen. Een functie van twee variabelen kan namelijk gereconstrueerd worden wanneer in ieder punt de gradiëntcomponenten $\frac{\delta f}{\delta x}$ en $\frac{\delta f}{\delta y}$ bekend zijn. De randen in het beeld worden hierdoor versterkt en met een drempeltoets wordt beslist of een punt al dan niet een randpunt is. Een rand is hier de overgang tussen twee elkaar rakende gebieden met onderling verschillende maar overigens constante intensiteitwaarden. De gradiënt van een plaats- en amplitudecontinu beeld $f(x,y)$ is:

$$\nabla f(x,y) = \left(\frac{\delta f}{\delta x}, \frac{\delta f}{\delta y} \right)$$

De modulus van de gradiënt vector is dan:

$$|\nabla f(x,y)| = \sqrt{\left(\frac{\delta f}{\delta x}\right)^2 + \left(\frac{\delta f}{\delta y}\right)^2}$$

Waarbij de richting van de gradiënt vector

$$\theta(x,y) = \arctan\left(\frac{\delta f/\delta y}{\delta f/\delta x}\right) \text{ is.}$$

Wordt nu teruggekeerd naar gedigitaliseerde beelden, waar discrete differenties worden gehanteerd in plaats van differentialen:

$$\Delta_x f(i,j) = f(i,j) - f(i-1,j)$$

$$\Delta_y f(i,j) = f(i,j) - f(i,j-1)$$

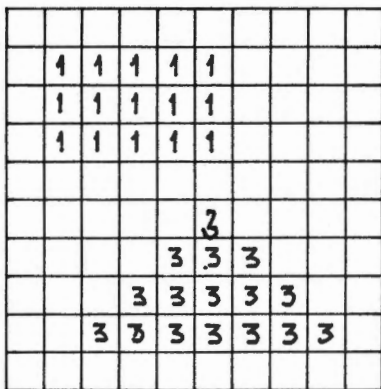
De modulus van de discrete gradiënt wordt gedefinieerd als:

$$|\nabla f(i,j)| = \sqrt{(\Delta_x f(i,j))^2 + (\Delta_y f(i,j))^2}$$

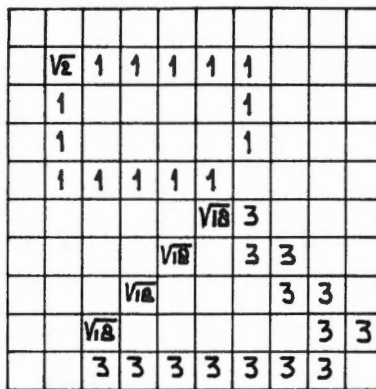
Om redenen van rekentijd kan deze worden benadert door middel van

$$|\Delta_x f(i,j)| + |\Delta_y f(i,j)|$$

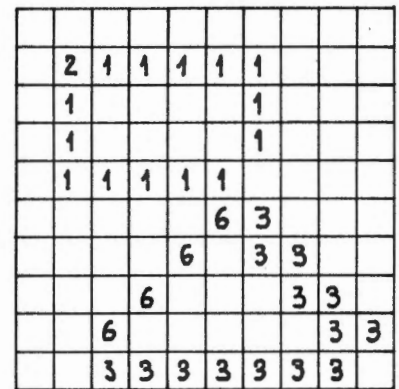
Er treedt een fout op die nul is voor $\Delta_x = 0$ of $\Delta_y = 0$ en maximaal is voor $\Delta_x = \Delta_y$. De drempel, welke beslist of een punt al dan niet een contourpunt wordt, zal echter kleiner moeten worden genomen dan het 'gewenste' intensiteitsverschil tussen twee elkaar rakende gebieden. Er treedt dan geen fout op. Als voorbeeld is figuur 2.4.1a door een gradiënt-operator gestuurd en de gedefinieerde als de benaderde moduli berekend welke in figuur 2.4.1b en 2.4.1c zijn weergegeven.



figuur 2.4.1a



figuur 2.4.1b



figuur 2.4.1c

figuur 2.4.1

Bekijken we nu ook het frequentiegedrag van deze operator en nemen we als voorbeeld de frequentieresponsie van de discrete differentie-operator $\Delta_y f(i,j) = f(i,j) - f(i,j-1)$ welke gerepresenteerd kan worden als een convolutie met een operator $[1 \ -1]$.

De bewerking komt overeen met het convolueren van de rijen van het beeld $f(i,j)$ met een operator die als impulsresponsie $h(k) = \delta(k) - \delta(k-1)$ heeft. Transformatie naar het z -domein door middel van de eenzijdige z -transformatie

$$H(z) = \sum_{k=0}^{\infty} h(k) \cdot z^k$$

geeft als overdrachtsfunctie

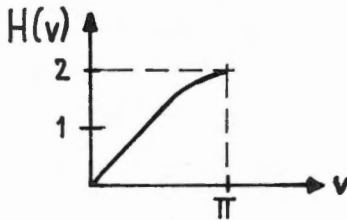
$$H(z) = 1 - z^{-1}$$

De frequentieresponsie wordt verkregen door substitutie van $z = e^{jv}$ in $H(z)$, hetgeen leidt tot $H(v) = 1 - e^{-jv} = 1 - \cos v + j \sin v$

$$|H(v)| = \left| 2 \cdot \sin\left(\frac{v}{2}\right) \right|$$

$$\arg H(v) = \arctan \frac{\sin v}{1 - \cos v} = \frac{\pi - v}{2}$$

In figuur 2.4.2 is de modulus van de responsie van Δy geschetst.



figuur 2.4.2

Dit geldt tevens ook voor de discrete differentie-operator $\Delta_x f(i,j)$. We zien hieruit dat differentiërende operatoren in het beeld domein overeen komen met filters die hoge spatiële frequenties ophalen zodat daardoor tevens eventuele "ruis" wordt versterkt.

Vergelijken we nu de twee methoden met elkaar dan heeft de methode die het oorspronkelijke beeld door een gradiënt-operator stuurt verschillende nadelen ten opzichte van de methode die in het oorspronkelijke beeld de contourpunten opspoort, namelijk:

- de langere bewerkingstijd
- dat niet de goede parameters van een plateau worden bepaald (zie figuur 2.4.1)
- de ruis wordt versterkt.

2.5 Het vinden van contouren

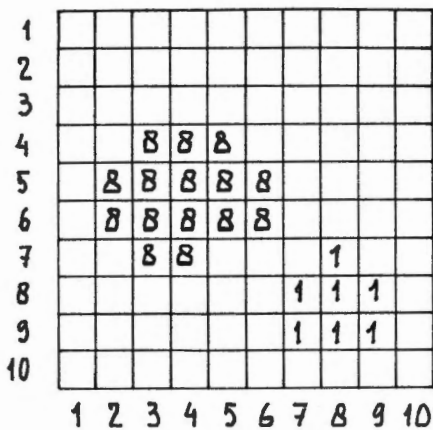
Voor het bepalen van de parameters van de plateaus zijn de contourpunten van de plateaus nodig. Om van alle plateaus de parameters te bepalen, zullen alle matrixelementen onderzocht moeten worden. Een methode is om de matrix rij voor rij te onderzoeken. Wanneer bij het onderzoeken een contourpunt van een plateau is gevonden dan wordt, voor het bepalen van de parameters, de contour van dat plateau een aantal keren gevolgd. Als dit gedaan is en het gevonden contourpunt zich bevindt op de plaatscoördinaten i, j , wordt het zoeken naar volgende contouren op de plaatscoördinaten $i, j+1$ hervat. Deze methode zal zeker alle contouren vinden, echter zullen de contouren meerdere keren worden gevonden hetgeen overbodig is. Om dit probleem op te lossen wordt elk contourpunt dat op een rij het begin- of eindpunt is van het laatst gevonden plateau, gemarkeerd terwijl de contour van dat plateau gevolgd wordt. Onder markeren wordt verstaan dat aan een matrixelement een waarde wordt gegeven anders dan de waarden in de oorspronkelijke matrix. Hierdoor kunnen de reeds gevonden contouren worden herkend. Linker- en rechtermarkeringen zijn nodig om het begin- en eindpunt op een rij aan te geven. Uit de waarden die zijn gegeven aan de linker- en rechtermarkeringen zullen tevens moeten worden bepaald welke hoogte een reeds gevonden plateau heeft.

Aangezien er 16 (0-15) waarden in de oorspronkelijke matrix kunnen voorkomen, zullen de waarden die aan de linker- en rechtermarkeringen worden gegeven buiten het gebied 0-15 moeten zijn. Een manier om uit de gegeven waarde tevens de hoogte van het reeds gevonden plateau te bepalen, is het aantal waarden die in de oorspronkelijke matrix kunnen voorkomen bij de plateauhoogte op te tellen. Als het een linkermarkering is en de hoogte van het plateau is bijvoorbeeld 8 dan wordt de linkermarkering $8+16=24$. Bij het vinden van deze markering hoeft dan alleen het aantal waarden ervan te worden afgetrokken om de hoogte van het reeds gevonden plateau te verkrijgen. De rechtermarkering zal nu buiten het gebied 0-31 moeten zijn. Een manier is om tweemaal de aantal waarden bij de plateauhoogte op te tellen. We krijgen dan als rechtermarkering $2 \cdot 16 + 8 = 40$.

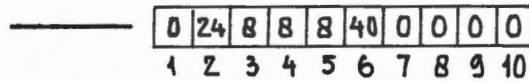
Als voorbeeld wordt de matrix van figuur 2.5.1a onderzocht. Het zoeken in de eerste rij en eerste kolom begint en rij voor rij wordt onderzocht. Hierbij wordt aangenomen dat de matrixelementen binnen

een contour dezelfde waarde hebben als de contourpunten zodat binnen een contour zich geen andere contouren bevinden.

Voordat het zoeken begint, zal eerst de waarde moeten worden bepaald die moet worden genegeerd. Indien er geen plateaus aanwezig zijn, zijn alle elementen 0 (nulniveau). De beginwaarde die niet mag worden gevonden is dan 0. Hetzelfde geldt bij het begin van het onderzoeken van een rij. Bij het onderzoeken wordt op rij 4 en kolom 3 een contourpunt gevonden. Dit contourpunt is het beginpunt voor het volgen van de contour. De contourpunten die op een rij het begin- of eindpunt zijn worden gemarkeerd, terwijl de contour gevolgd wordt. Verondersteld wordt nu dat het zoeken zover is dat het zoeken op rij 5 begint. Van het eerste plateau met hoogte 8 zijn de parameters al bepaald. De contourpunten die op een rij het begin- en eindpunt zijn van dat plateau, zijn ook gemarkeerd. Figuur 2.5.1b geeft de inhoud van de kolomelementen van rij 5. Wanneer op kolom 2 wordt aangekomen, kan uit de gegeven waarde worden herkend dat deze linker gemarkeerd is. Uit de gegeven waarde wordt dan bepaald welke waarde moet worden genegeerd. In dit geval is het $24-16=8$. Wordt nu kolom 6 bereikt dan wordt herkend dat deze rechter gemarkeerd is. Dit duidt erop dat de waarde nul (nulniveau) niet mag worden gevonden. Indien zich binnen een contour geen andere contouren bevinden, hoeft, zoals blijkt, de rechtermarkeringen niet voor elke plateauhoogte anders te zijn.



figuur 2.5.1a



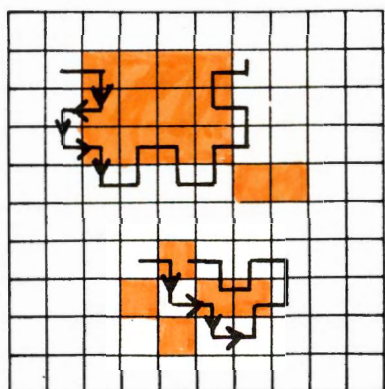
figuur 2.5.1b

Aangezien contouren zich in een contour kunnen bevinden, zijn de rechtermarkeringen nodig waaruit de plateauhoogten kunnen worden bepaald. Tevens is een extra matrix, ter grootte van $[1.((N/2)+1)]$, en een pointer nodig. In de eerste kolom van de extra matrix is de waarde 0 (nulniveau) geplaatst die tijdens het verwerken van de beeldmatrix niet meer verandert. In de volgende kolommen komen de bepaalde waarden uit de linkermarkeringen te staan. De pointer wijst een kolomelement van de extra matrix aan. De inhoud van het aangewezen kolomelement mag niet worden gevonden. Voor het hoe en het waarom nemen we figuur 2.5.2a als voorbeeld. Aangenomen wordt weer dat het zoeken zover is dat het zoeken in rij 6 begint. De parameters van de plateaus zijn al bepaald. De plateaus waaruit de parameters zijn bepaald, zijn in figuur 2.5.2d weergegeven. Figuur 2.5.2c geeft de waarden weer van de matrixelementen op rij 6. Het nulniveau is de waarde die niet mag worden gevonden wanneer het zoeken in een rij begint. De pointer wijst dan op het eerste kolomelement van de extra matrix. Wanneer bij kolom 2 wordt aangenomen dan wordt herkend dat deze linker gemarkeerd is. De pointer wordt met één opgehoogd. De bepaalde plateauhoogte (1) wordt in de aangewezen kolom geplaatst. Bereiken we kolom 4 dan wordt herkend dat deze linker gemarkeerd is. De pointer wordt met één verhoogd en de bepaalde waarde (2) wordt in de aangewezen kolom geplaatst. Bij kolom 5 wordt herkend dat deze rechter gemarkeerd is. De bepaalde waarde (2) wordt nu vergeleken met de inhoud van het aangewezen kolomelement. In dat geval zijn ze aan elkaar gelijk. De pointer wordt dan met één verlaagd. Wordt nu kolom 9 bereikt dan zien we dat voor de kolom kleiner dan 9 en voor de kolom groter dan 9 dezelfde waarde niet mag worden gevonden. De pointer mag dan niet met één verlaagd worden. Dit kan worden bereikt als de inhoud van het aangewezen kolomelement niet gelijk is aan de bepaalde waarde (3). Contourpunten van plateaus die één kolom breed zijn, zullen dan rechter gemarkeerd moeten worden. Wordt tenslotte op kolom 10 aangekomen dan wordt herkend dat deze rechter gemarkeerd is. De inhoud van het aangewezen kolomelement is gelijk aan de bepaalde waarde (1). De pointer wordt met één verlaagd. We zijn nu weer in de beginsituatie teruggekomen. Hierdoor kan de extra matrix, zonder bewerkingen uit te voeren, weer worden gebruikt bij het onderzoeken van de volgende rij. Figuur 2.5.2b geeft de posities van de pointer en de extra matrix weer.

2.6 Het volgen van contouren

Voor het bepalen van de parameters van een plateau wordt de contour van dat plateau een aantal keren gevolgd. Voor het volgen van de contour worden hier twee contour-volg-algorithmen besproken.

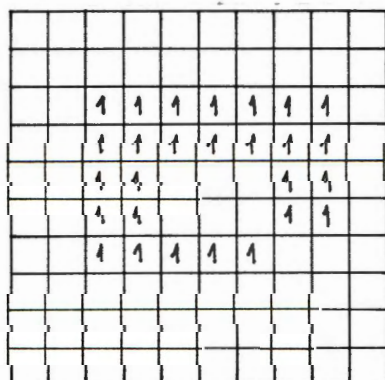
- De eerste procedure (A) is als volgt. Wordt een contourpunt gevonden, waarvan de contour nog niet is gevolgd, dan slaan we rechtsaf als het



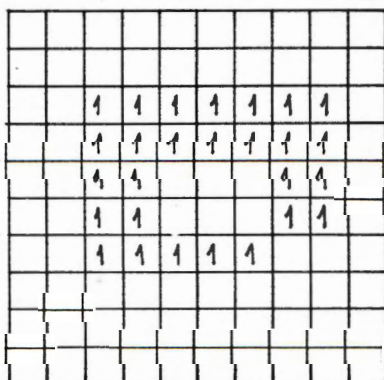
laatst doorlopen punt dezelfde waarde heeft als het beginpunt en linksaf als het laatste punt niet dezelfde waarde heeft. Uit figuur 2.6.1 blijkt dat er situaties kunnen voorkomen waarin niet de gehele contour gevolgd wordt zodat meerdere plateaus worden gelocaliseerd dan er in werkelijkheid zijn.

figuur 2.6.1

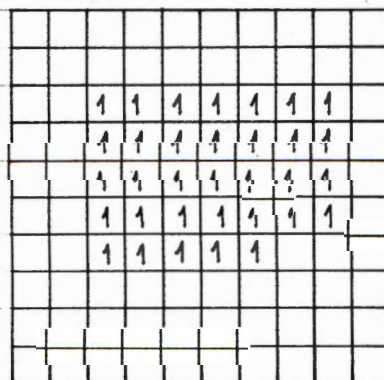
De parameters van een plateau worden bepaald door de punten die binnen- of op de contour liggen. Er zijn echter situaties dat niet het gehele plateau beschikbaar komt. In onderstaande figuren is een voorbeeld gegeven. Figuur 2.6.2a is de oorspronkelijke matrix. Figuur 2.6.2b is het plateau weergegeven dat met de procedure is verkregen. Figuur 2.6.2c is het gehele plateau.



figuur 2.6.2a



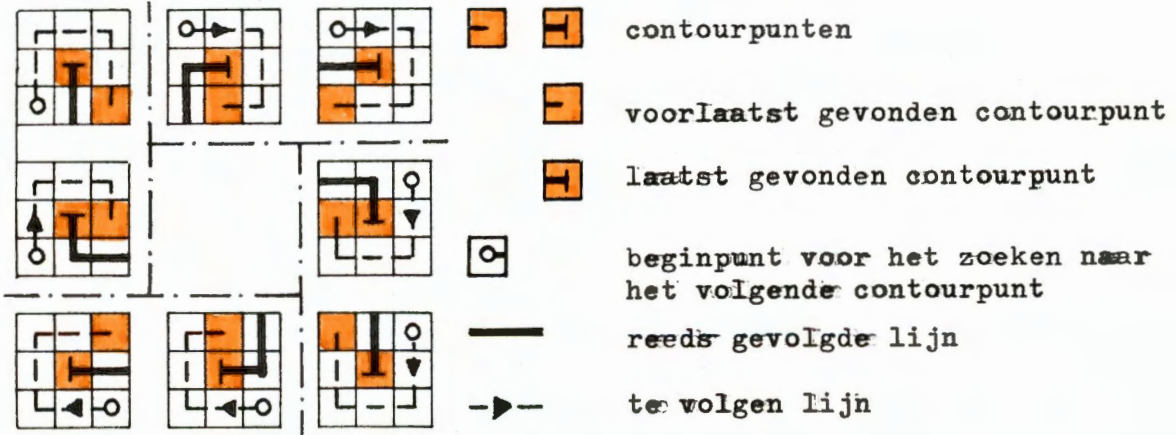
figuur 2.6.2b



figuur 2.6.2c

figuur 2.6.2

- Bij de tweede procedure (B) wordt het zoeken naar een volgend contourpunt ook bepaald door de plaatscoördinaten van de twee laatst gevonden contourpunten. Figuur 2.6.3 laat het principe zien. Hierbij wordt de contour met de klok mee gevolgd.



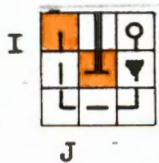
figuur 2.6.3

Voor het uitleggen van het principe nemen we één van de 8 figuren (figuur 2.6.4a). Het punt op i, j is het laatst gevonden contourpunt.



I
J
figuur 2.6.4a

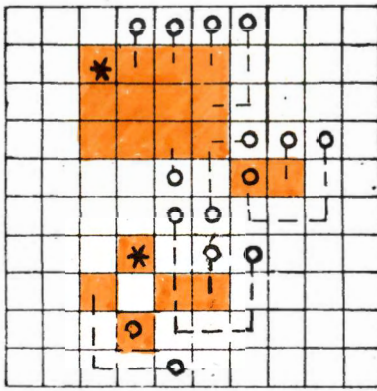
Het punt op $i, j-1$ is het voorlaatst gevonden contourpunt. Als de contour met de klok mee wordt gevolgd dan zijn de punten $i-1, j-1$ en $i-1, j$ geen contourpunten. Er is al onderzocht of het contourpunten zijn.



I
J
figuur 2.6.4b

Om de gehele contour te volgen, zal punt $i-1, j+1$ het beginpunt moeten zijn voor het zoeken naar het volgende contourpunt. Indien $i-1, j+1$ geen contourpunt is, wordt onderzocht of punt $i, j+1$ een contourpunt is. Indien het geen contourpunt is, wordt onderzocht of het punt $i+1, j+1$ een contourpunt is. Indien het een contourpunt is dan krijgen we de situatie zoals hiernaast is weergegeven (figuur 2.6.4b). Het laatst gevonden punt bevindt zich nu op $i+1, j+1$. Het voorlaatste punt bevindt zich op i, j . We weten nu zeker dat het punt $i, j+1$ geen contourpunt is.

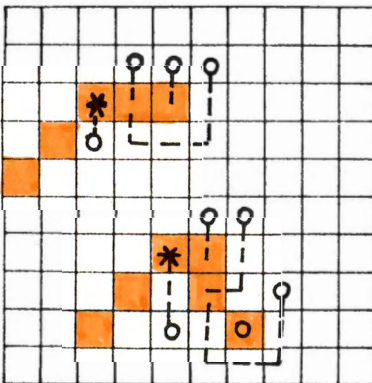
Het beginpunt voor het zoeken naar het volgend contourpunt begint dan op $i, j+2$. We kunnen dit toepassen tot het beginpunt weer bereikt is. Wanneer een contourpunt van een nog niet gevolgde contour is gevonden dan hebben we de situatie zoals in figuur 2.6.4a is weergegeven. Figuur 2.6.5 laat zien hoe de contouren van figuur 2.6.1 gevolgd worden.



* beginpunt voor het volgen van de contour

figuur 2.6.5

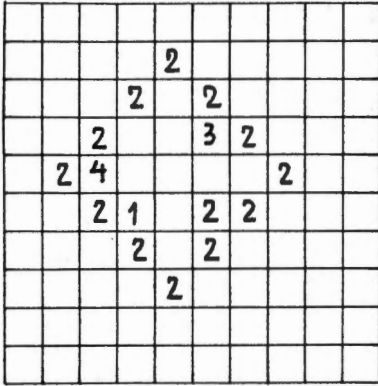
Aangenomen wordt nu dat het beginpunt voor het volgen van de contour op de coördinaten i, j bevindt. Wanneer de contour is gevolgd, is het beginpunt het laatst gevonden contourpunt. Als het voorlaatste gevonden contourpunt zich bevindt op $i, j+1$ of $i+1, j+1$ zal onderzocht moeten worden of de contour op $i+1, j-1$ verder gaat. In figuur 2.6.6 wordt dit geïllustreerd. Hierbij zijn de met een * gemerkte punten beginpunten voor het volgen van de contouren



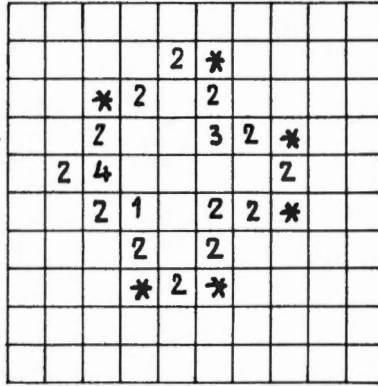
figuur 2.6.6

Zowel procedure A als procedure B hebben bepaalde nadelen. De nadelen van procedure B worden echter veroorzaakt doordat zich in een contour contouren bevinden. Voor het uiteenzetten van één van deze nadelen nemen we figuur 2.6.7a als voorbeeld. Bij het zoeken naar contouren zal op 2,5 een contourpunt worden gevonden waarvan de contour nog niet gevolgd is. Voor het bepalen van de parameters van het plateau wordt de contour een aantal keren gevolgd. Indien dit gedaan is, wordt het zoeken van volgende contouren op 2,6 hervat. Op 3,5 vinden we een tweede contourpunt waarvan de contour nog niet

gevolgd is. We bepalen uit figuur 2.6.3 het beginpunt voor het zoeken naar het volgende contourpunt. Het beginpunt bevindt zich op 2,6. Dit punt heeft een gelijke intensiteit als het beginpunt en is dus een contourpunt. Het volgende contourpunt bevindt zich dan op 1,7. We krijgen dan niet de goede parameters. Om dit te voorkomen, worden de punten die met een * in figuur 2.6.7b zijn gemerkt, bij het volgen van de eerste contour zodanig gemarkeerd dat het volgen van de tweede contour correct kan plaatsvinden.



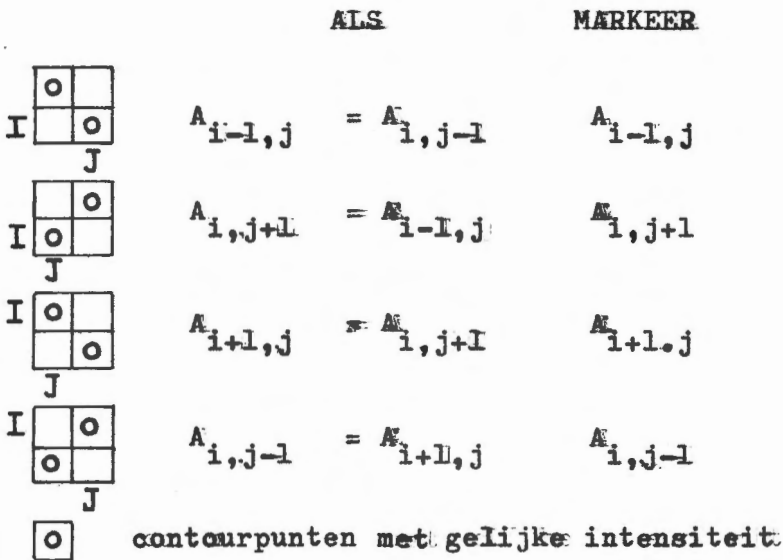
figuur 2.6.7a



figuur 2.6.7b

figuur 2.6.7

Figuur 2.6.8 geeft de situaties weer wanneer er gemarkeerd moet worden. Hierbij is A de beeldmatrix en i, j de coördinaten van het laatst gevonden contourpunt.



figuur 2.6.8

Bekijken we nu ook figuur 2.6.9a dan worden de punten die aangegeven zijn met een * in figuur 2.6.9b ook gemarkeerd bij het volgen van de contour van het eerst gevonden plateau.

			1				
		1	1	1		3	
		1	1	1	1	1	3 3
	1	1	1	1	3	3	1 3 3
1	1	1	1	3	3	3	3 1 3
	1	1	1	3	3	3	1 3 3
		1	1	1	3	3	3 3
			1			3	3 3

figuur 2.6.9a

			1				
			1	1	1		3
			1	1	1	1	*3 3
		1	1	1	1	3	3 1 *3 3
1	1	1	1	3	3	3	3 1 3
	1	1	1	3	3	3	1 *3 3
		1	1	1	3	1	*3 3 3
			1	1	1	*3	3 3 3
				1			3 3 3

figuur 2.6.9b

Er zijn nu twee mogelijkheden om deze punten te markeren namelijk door deze punten als aparte of als niet aparte plateaus te beschouwen. Als ze als aparte plateaus worden beschouwd dan heeft figuur 2.6.9a vijf plateaus. De vormen zijn weergegeven in figuur 2.6.10a tot en met 2.6.10d. Wanneer ze niet als aparte plateaus worden beschouwd dan heeft figuur 2.6.9a drie plateaus.

			1				
		1	1	1			
		1	1	1	1	1	
	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
		1	1	1	1	1	
			1	1	1		
				1			

figuur 2.6.10a

						3	
							3
							3
							3
							3 3
							3 3 3
							3 3 3 3
							3 3 3

figuur 2.6.10b

						3	3
						3	3 3 3
						3	3 3
							3

figuur 2.6.10c

							3
							3
							3
							3
							3

6 figuur 2.6.10d

Een deelplateau wordt weergegeven door een rechthoek waarvan de zijden worden bepaald door de oppervlakte van het deelplateau en de richtingscoëfficiënt die wordt bepaald door de zwaartepunten van het gehele plateau en het deelplateau. Tevens vinden er afrondingsfouten plaats aangezien voor de parameters met integers gewerkt wordt. Hierdoor zal de invloed gering zijn op het weer opgebouwde beeld wanneer ze niet als aparte plateaus worden beschouwd.

De plaatscoördinaten van de twee laatst gevonden contourpunten zijn bekend. Bij het volgen van de contour moet dus steeds getest worden of het beginpunt is bereikt. Dit testen kan op twee manieren gebeuren. De eerste manier is het testen op de plaatscoördinaten. De tweede manier, die eenvoudiger is, is het beginpunt te markeren vóór het volgen van de contour begint. Omdat zeker is dat één van de acht buurpunten van een beginpunt geen beginpunt van een contour met de hoogte gelijk aan het beginpunt kan zijn, hoeft de markering voor het beginpunt niet voor elke contour anders te zijn. Een manier om het beginpunt te markeren is het aantal waarden (16) af te trekken van de plateauhoogte. Als bijvoorbeeld het beginpunt de waarde 8 heeft, wordt het beginpunt gemarkeerd met: $8-16 \Rightarrow 8$. De contourpunten van een gevonden contour kunnen hierdoor 4 waarden aannemen, namelijk de waarde van:

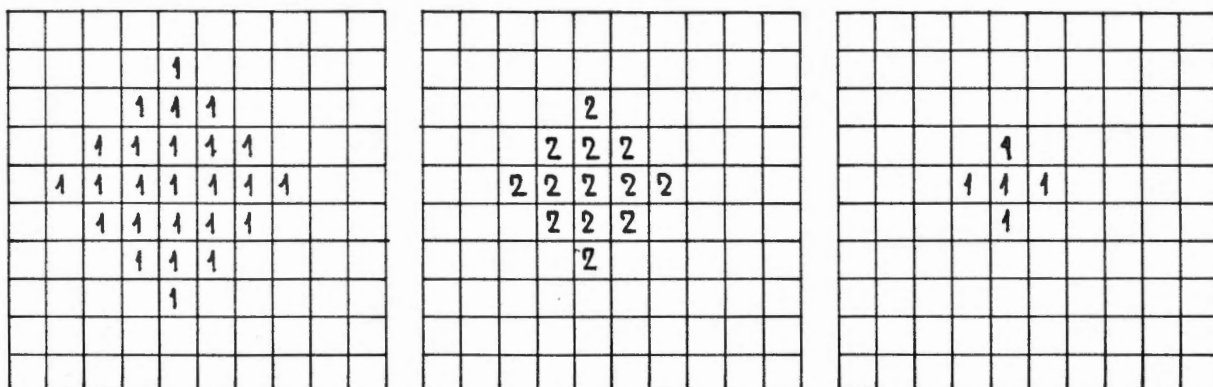
- het beginpunt;
- de plateauhoogte;
- de linkermarkering;
- de rechtermarkering.

Bij het onderzoeken of een punt een contourpunt is, zal nagegaan moeten worden of het één van de boven vermelde waarden heeft. Heeft het één van deze waarden dan is het een contourpunt. We komen nu bij het tweede probleem. Voor het uiteenzetten van dit probleem nemen we weer een voorbeeld dat in figuur 2.6.11 is weergegeven.

1										
2				1						
3			1	2	1					
4		1	2	1	2	1				
5	1	2	1	1	1	2	1			
6		1	2	1	2	1				
7			1	2	1					
8				1						
9										
10										
	1	2	3	4	5	6	7	8	9	10

figuur 2.6.11

Er zijn drie plateaus waarvan twee plateaus dezelfde hoogte hebben. De plateaus zijn in figuur 2.6.12 weergegeven. Verondersteld wordt dat het vinden van contouren zover is gevorderd dat op rij 4 en kolom 5 is aangekomen. De contourpunten van de twee gevonden contouren die op een rij het begin- of eindpunt zijn, zijn gemarkeerd. Uit figuur 2.5.3 bepalen we het beginpunt voor het zoeken naar het volgende contourpunt, Het beginpunt bevindt zich op 3,6.



figuur 2.6.12

Dit punt is rechter gemarkeerd. Tevens is het een contourpunt. We volgen dan de verkeerde contour. Dit kan worden voorkomen door elke contour zijn eigen linker- en rechtermarkering te geven. Een manier om elke contour zijn eigen markering te geven en waaruit tevens de plateauhoogte kan worden bepaald, is door een contour een eigen nummer te geven en gebruik te maken van het aantal waarden die in een oorspronkelijke matrix kunnen voorkomen. Dit gaat als volgt:

$$A_{i,j} = A_{i,j} + \text{nummer} \cdot \text{aantal} \quad \text{voor een linkermarkering}$$

$$A_{i,j} = A_{i,j} + \text{nummer} \cdot \text{aantal} + \text{factor} \quad \text{voor een rechtermarkering}$$

waarin: A = beeldmatrix

nummer = nummer van het plateau $E_{1,2,\dots}$

aantal = aantal waarden die in de oorspronkelijke matrix kunnen voorkomen (16)

De "factor" is een maat voor het maximum aantal plateaus dat in de oorspronkelijke matrix kan voorkomen. Is het een $M \cdot N$ matrix dan zijn er maximaal $M \cdot N / 2$ rechtermarkeringen. Het verschil moet dan minimaal $M \cdot N / 2 \cdot 16 = 8MN$ zijn tussen een linker- en een rechtermarkering.

Vergelijken we nu deze procedures met elkaar dan hebben we gezien dat voor beide procedures situaties kunnen voorkomen dat voor het bepalen van de parameters van een plateau niet het gehele plateau beschikbaar komt. Worden de te verwachten beelden hierbij betrokken, dan kunnen we verwachten dat met procedure B betere parameters worden verkregen dan met procedure A. Een tweede groot nadeel van procedure A is dat in een ruisig beeld veel meer plateaus worden gevonden dan met procedure B.

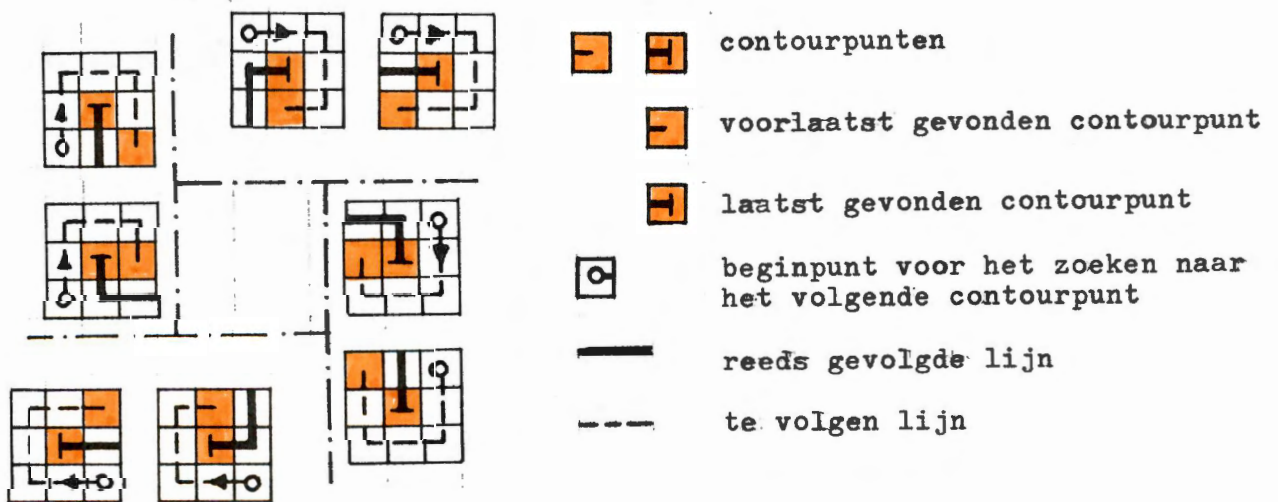
3. Gerealiseerde beeldverwerkingsalgorithmen

3.1 Inleiding

Terwijl de contour wordt gevolgd, wordt elk contourpunt gemarkeerd, dat in een rij het begin- of het eindpunt is. Het markeren van die punten wordt toegelicht in paragraaf 3.2. In paragraaf 3.3 wordt het bepalen van het zwaartepunt van een (deel-)plateau behandeld. In paragraaf 3.4 worden structuurdiagrammen voor het bepalen van de parameters die een beeld beschrijven, besproken, alsmede een methode om het algoritme te versnellen. Ook wordt in paragraaf 3.4 het structuurdiagram behandeld, dat nodig is om vanuit deze parameters het beeld weer te reconstrueren.

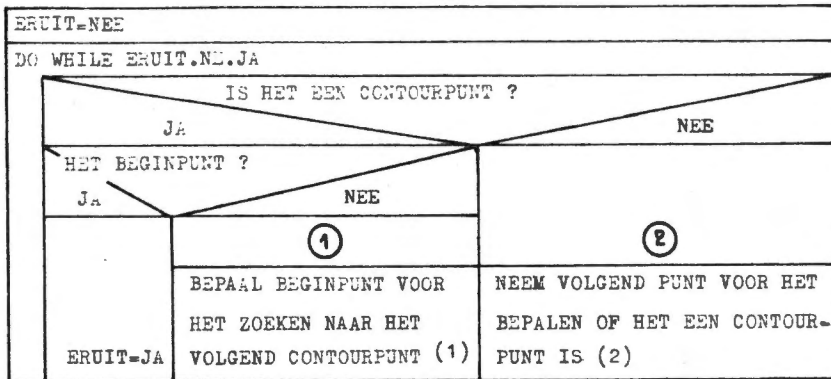
3.2 Het markeren van de contourpunten

Om te vermijden dat contouren meerdere keren worden gevolgd, wordt elk contourpunt dat op een rij het begin- of eindpunt is van het laatst gevonden plateau, gemarkeerd terwijl de contour van dat plateau gevolgd wordt. Voor het bepalen of een contourpunt moet worden gemarkeerd, wordt figuur 2.6.3 anders weergegeven en wel door figuur 3.2.1. Hierbij wordt aangenomen dat het laatst gevonden punt zich bevindt op de plaatscoördinaten i, j .



figuur 3.2.1

Tevens wordt voor de uitleg van de methode het vereenvoudigde structuurdiagram gebruikt (figuur 3.2.2).



- (1) Als bijvoorbeeld het voorlaatst gevonden punt zich bevindt op $i-1, j-1$, is het beginpunt voor het zoeken naar het volgende contourpunt $i-1, j+1$
- (2) Als bijvoorbeeld $i-1, j+1$ geen contourpunt is, wordt $i, j+1$ het volgende te onderzoeken punt

figuur 3.2.2

Contourpunten van plateaus die één kolom breed zijn, moeten rechter gemarkeerd zijn. Hierdoor mag een contourpunt dat al rechter gemarkeerd is, niet meer linker gemarkeerd worden.

We beginnen met de omcirkelde 1 (dit is een contourpunt).

- Indien het voorlaatst gevonden punt zich bevindt op de plaatscoördinaten $i-1, j$ of op $i-1, j+1$, wordt het punt i, j rechter gemarkeerd. Immers, het punt $i, j+1$ is al onderzocht.
- Indien het voorlaatst gevonden punt zich bevindt op de coördinaten $i+1, j$ of op $i+1, j-1$, wordt het punt i, j linker gemarkeerd tenzij het rechter gemarkeerd is.

De omcirkelde 2

- Indien op het punt $i, j+1$ is aangekomen, wordt het punt i, j rechter gemarkeerd tenzij het een beginpunt voor het volgen van de contour is. Bij het volgen van de contour zal bij zo'n situatie steeds moeten worden getest of het voorlaatst gevonden punt het beginpunt is. Om dit te vermijden, wordt als het punt i, j het beginpunt is, vóór het volgen van de contour begint onderzocht of het punt $i, j+1$ een contourpunt is. Als dit het geval is, wordt aan de kolom van het beginpunt een kolom toegevoegd zodat het punt $i, j+1$ het laatst gevonden contourpunt wordt.
- Indien op het punt $i, j-1$ is aangekomen, wordt het punt i, j linker gemarkeerd tenzij het al rechter gemarkeerd is.

3.3 Het bepalen van het zwaartepunt van een (deel-)plateau

De positie van een plateau wordt aangegeven door zijn zwaartepunt. Het zwaartepunt van een vlakke figuur met oppervlakte m dat is opgebouwd uit n vierkante elementen wordt in cartesische coördinaten geschreven als:

$$x_c = \frac{\sum_{i=1}^n m_i x_i}{m}$$

$$y_c = \frac{\sum_{i=1}^n m_i y_i}{m}$$

waarin:

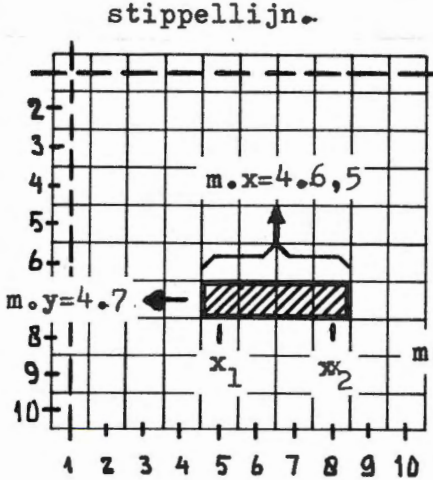
n = totaal aantal elementen

m_i = oppervlakte van element m_i

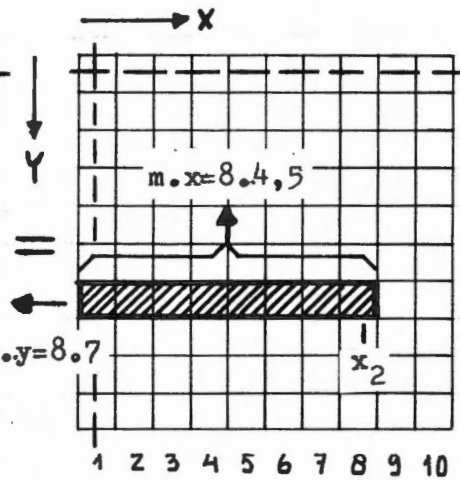
x_i, y_i = plaatstcoördinaten van element m_i

$m = \sum_{i=1}^n m_i$ = totale oppervlakte van de groep van n elementen

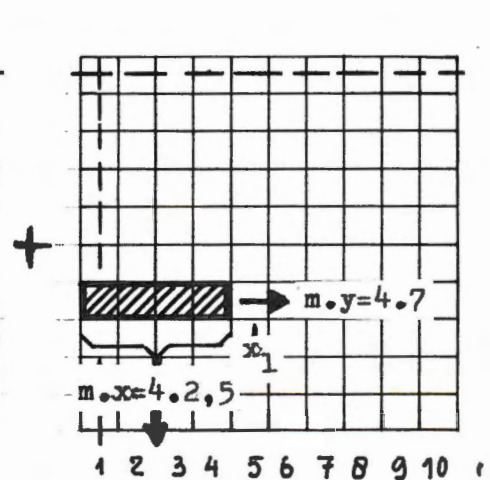
Voor het bepalen van het zwaartepunt van een plateau zullen eerst $\sum m_i x_i, \sum m_i y_i$ en $\sum m_i$ bekend moeten zijn. Deze worden bij het eerste keer volgen van de contour bepaald. De werkwijze is in figuur 3.3.1 weergegeven. Hierbij wordt gebruik gemaakt van twee referentieniveaus, één in de x - en één in de y - richting. De referentieniveaus kunnen willekeurig worden gekozen. In dit geval zijn ze op de eerste rij en eerste kolom van de matrix gekozen. Ze zijn aangegeven met een stippelijne.



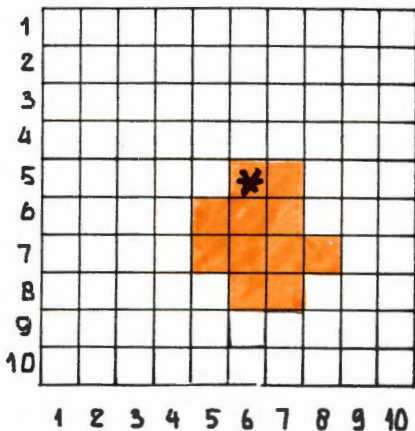
figuur 3.3.1a



figuur 3.3.1b



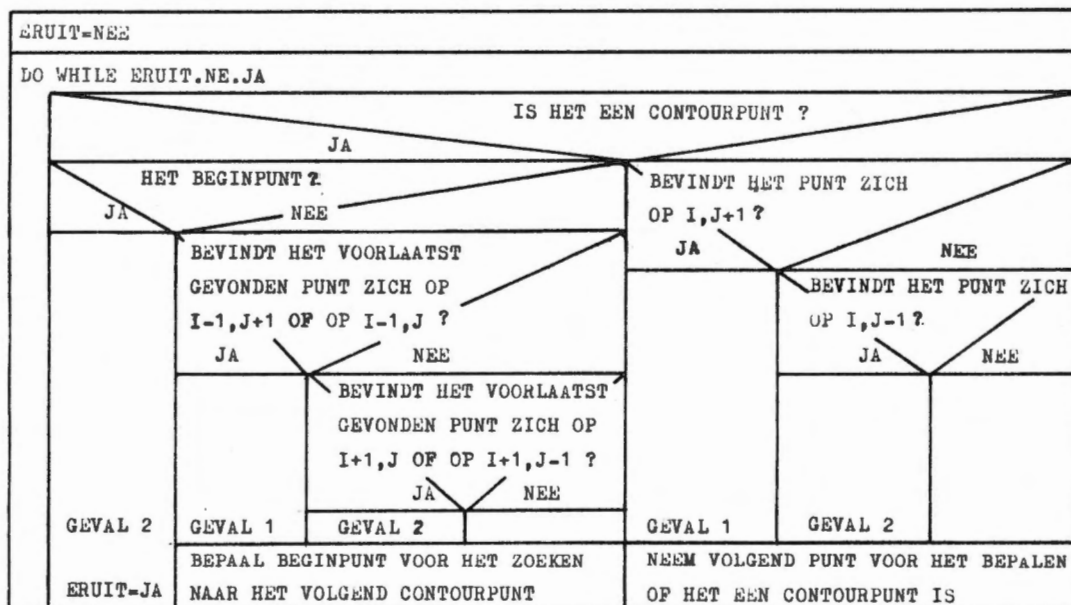
figuur 3.3.1c



figuur 3.3.1d

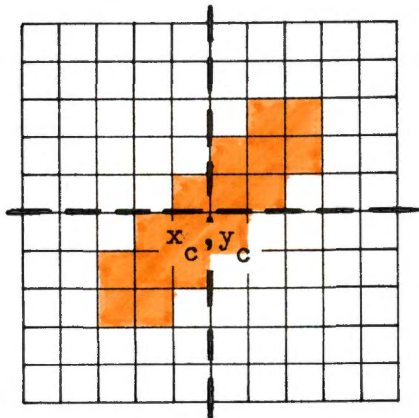
***** beginpunt voor het volgen van de contour

In figuur 3.3.1a is het punt op 7,5 het beginpunt en het punt op 7,8 het eindpunt in rij 7 van het plateau in figuur 3.3.1d. Er wordt gebruik gemaakt van de plaatscoördinaten van de gemarkeerde contourpunten. Wanneer een contourpunt rechter gemarkeerd wordt, krijgen we de situatie zoals in figuur 3.3.1b is weergegeven. Bij $\sum m_i$ wordt x_2 opgeteld. Bij $\sum m_i x_i$ wordt $x_2(x_2+1)/2$ opgeteld. In het voorbeeld van figuur 3.3.1d wordt het punt op 7,8 rechter gemarkeerd zodat $m=8$ en $m.x=8(8+1)/2$. Wanneer een contourpunt linker gemarkeerd is, krijgen we de situatie zoals in figuur 3.3.1c is weergegeven. Bij $\sum m_i$ wordt $-(x_1-1)$ opgeteld. Bij $\sum m_i x_i$ wordt $-(x_1-1)x_1/2$ opgeteld. In het voorbeeld wordt het punt op 7,5 linker gemarkeerd zodat $m=-(5-1)$ en $m.x=-(5-1)5/2$. Hetzelfde geldt voor $\sum m_i y_i$, hierbij is echter $y_i=y=$ constant. Het structuurdiagram van figuur 3.2.2 is dan uitgebreid. Het diagram is in figuur 3.3.2 weergegeven. Hierbij is figuur 3.3.1b geval 1 genoemd en figuur 3.3.1c geval 2. Het laatst gevonden punt bevindt zich op i,j .



figuur 3.3.2

Uit de gegevens die met de DO-loop zijn verkregen, wordt het zwaartepunt van het gehele plateau bepaald. Door de contour een tweede keer te volgen, kunnen de zwaartepunten van de deelplateaus worden bepaald. Hierbij wordt dezelfde methode als voor het bepalen van het zwaartepunt van het gehele plateau toegepast. De referentieniveaus worden echter naar het zwaartepunt van het gehele plateau verschoven (figuur 3.3.3).



figuur 3.3.3

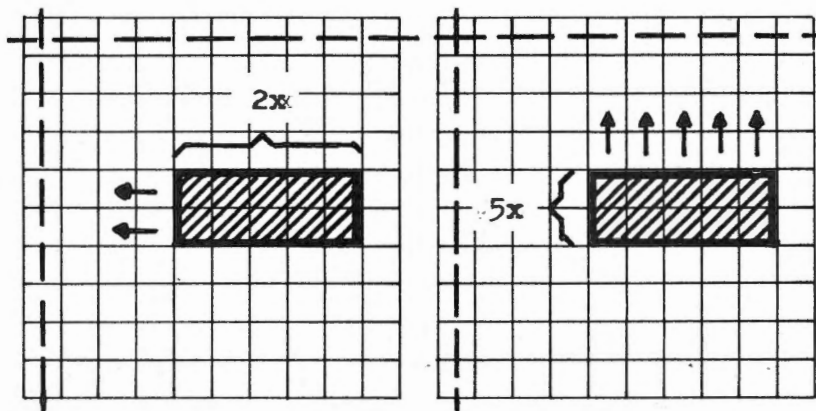
Nadat de contour voor de tweede keer gevolgd is, worden uit de reeds verkregen gegevens, de zwaartepunten van de deelplateaus bepaald. Tenslotte wordt met het zwaartepunt van het gehele plateau dat met het voor de eerste keer volgen van de contour is verkregen, de parameters van het plateau bepaald. Het verloop voor het bepalen van de parameters is in figuur 3.3.4 weergegeven.

1 ^e keer volgen
- Bepaal $\sum m_i x_i$, $\sum m_i y_i$ en $\sum m_i$ van het gehele plateau
Bepaal het zwaartepunt van het gehele plateau
2 ^e keer volgen
- Bepaal $\sum m_i x_i$, $\sum m_i y_i$ en $\sum m_i$ van elk deelplateau
- Markeer de contourpunten die moeten worden gemarkeerd
- Markeer de punten zoals in paragraaf 2.6 beschreven zijn
Bepaal de zwaartepunten van de deelplateaus
Bepaal met het zwaartepunt van het gehele plateau, de parameters van het plateau

figuur 3.3.4

Voor het bepalen van de parameters van een plateau zal dus twee keer de contour moeten worden gevolgd. Indien mocht blijken dat de vervorming te groot is, kan vermindering van de vervorming worden bereikt door een deelplateau als een apart plateau te beschouwen. Voor het bepalen van de parameters van de vier verkregen plateaus zal de contour dan nog eens moeten worden gevolgd.

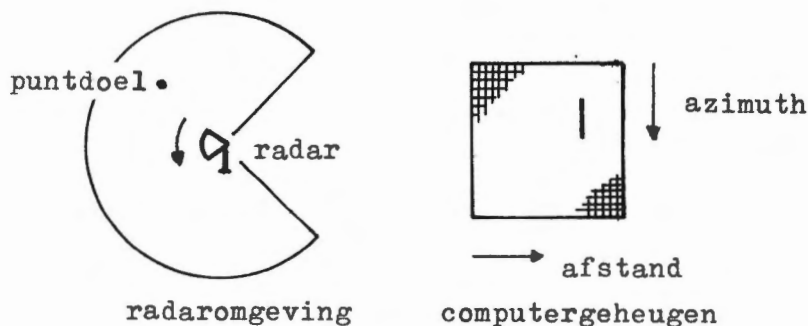
In het bovenstaande hebben we gebruik gemaakt van de werkwijze zoals in figuur 3.3.5a is weergegeven. De werkwijze kan ook zijn zoals in figuur 3.3.5b is weergegeven. De matrix wordt rij voor rij onderzocht.



figuur 3.3.5a

figuur 3.3.5b

Ten gevolge van de eindige afmetingen van de antennediameter zal bij het ronddraaien van de antenne een puntdoel een verticaal streepje in het beeld geven (figuur 3.3.6).

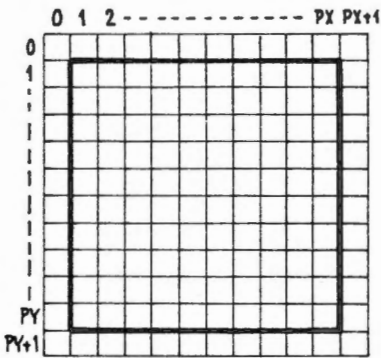


figuur 3.3.6

De matrix wordt voor het vinden van contouren dan kolom voor kolom onderzocht. Immers, markeringen geven meer bewerkingen doordat uit de gegeven waarden onder andere de plateauhoogten worden bepaald. Voor het bepalen van de parameters van een plateau zal bij de methode van figuur 3.3.5a dan minder bewerkingen behoeven te worden uitgevoerd dan bij de methode van figuur 3.3.5b.

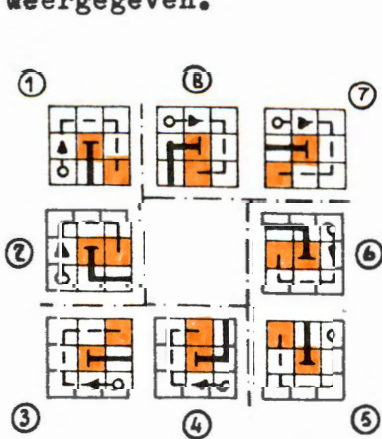
3.4 De structuurdiagrammen

In het voorafgaande werd steeds aangenomen dat de plateaus de randen van de beeldmatrix niet konden raken. De plateaus kunnen dit echter wel. Hierdoor zal, wanneer het laatst gevonden contourpunt zich op een rand bevindt, zowel bij het bepalen van het beginpunt voor het zoeken naar het volgende contourpunt als bij het nemen van een volgend punt voor het onderzoeken of het een contourpunt is, maatregelen moeten worden genomen om binnen de matrix te blijven. Deze maatregelen kunnen achterwege worden gelaten door in plaats van een M bij N matrix een $\emptyset:M+1$ bij $\emptyset:N+1$ matrix te declareren waarbij de randen vóór het bewerken met het nulniveau worden gevuld. De beeldmatrix blijft een M bij N matrix. In figuur 3.4.1 is het weergegeven. Hierbij is voor het aantal rijen van de beeldmatrix PY en voor het aantal kolommen PX genomen.

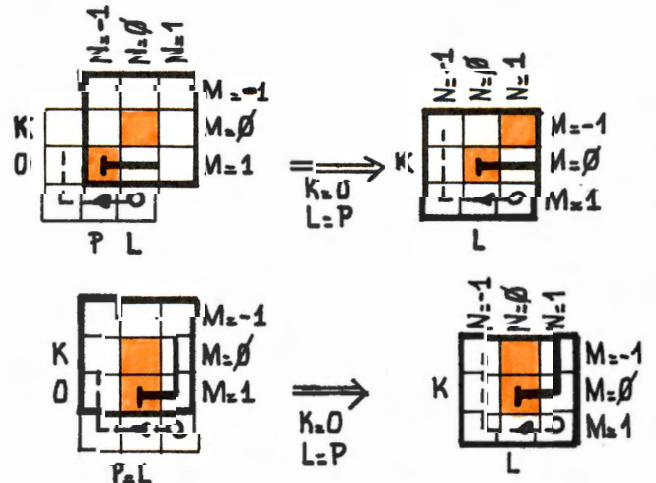


figuur 3.4.1

Voor het verklaren van de te treffen maatregelen bij het bepalen van het beginpunt nemen we de figuren 3 en 4 die in figuur 3.4.2 zijn weergegeven.

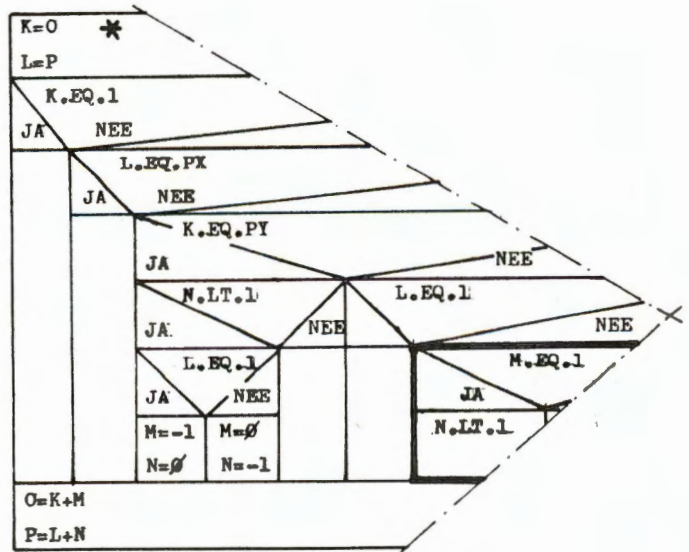
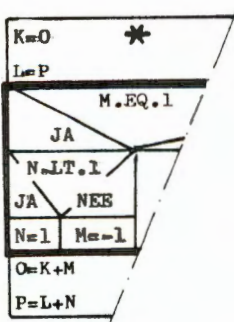
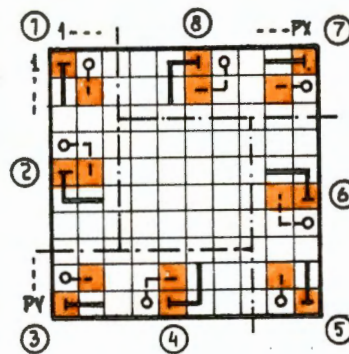
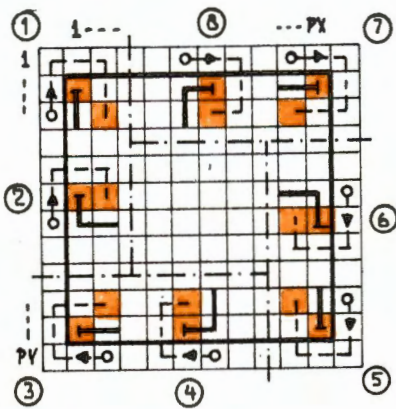


figuur 3.4.2



figuur 3.4.3

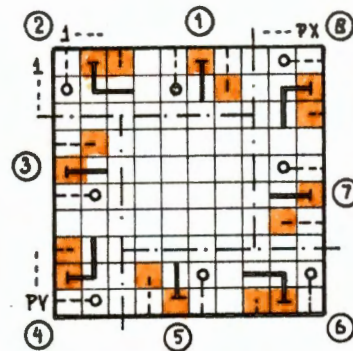
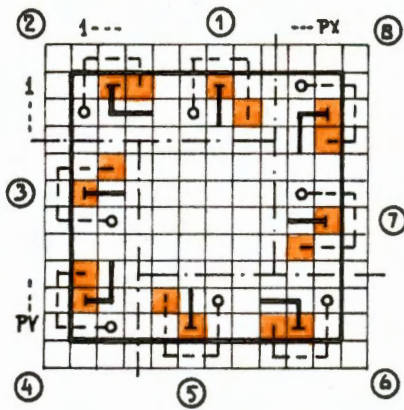
Het voorlaatst gevonden contourpunt bevindt zich op K,L (figuur 3.4.3). Het laatst gevonden contourpunt bevindt zich op O,P met $O=K+M$ waarbij $M=1$ is en $P=L+N$ waarbij $N=-1$ of $N=\emptyset$ is. Het laatst gevonden contourpunt wordt het voorlaatst gevonden contourpunt door $K=0$ en $L=P$ te maken. In de structuurdiagrammen van figuur 3.4.4 wordt dit aangegeven met een $*$. Uit K en L en uit M en N wordt het beginpunt bepaald. Het beginpunt is op O,P met $O=K+M$ waarbij $M=1$ blijft en $P=L+N$ waarbij $N=1$ wordt. Wanneer echter het laatst gevonden punt zich op de rij PY bevindt, worden M en N anders. Hierdoor moet bij het bepalen van een beginpunt altijd eerst getest worden of het laatst gevonden punt zich op één van de randen van de matrix bevindt. Gedeelten van de structuurdiagrammen zijn in figuur 3.4.4 weergegeven. In figuur 3.4.4a is een $\emptyset:PY+1$ bij $\emptyset:PX+1$ matrix gedeclareerd, in figuur 3.4.4b een PY bij PX matrix.



figuur 3.4.4a

figuur 3.4.4b

Voor het verklaren van de maatregelen die moeten worden getroffen bij het nemen van een volgend punt voor het onderzoeken of het een contourpunt is, nemen we de figuren 5 en 6 die in figuur 3.4.2 zijn weergegeven. Het laatst gevonden contourpunt bevindt zich op K,L. Nadat het punt op O,P is onderzocht met $O=K+M$ waarbij $M=\emptyset$ is en $P=L+N$ waarbij $N=1$ is, wordt O,P met $O=K+M$ waarbij $M=1$ wordt en $P=L+N$ waarbij $N=1$ blijft, het nieuw te onderzoeken punt. Wanneer echter K rij PY is, worden M en N anders. Hierdoor moet bij het nemen van een volgend te onderzoeken punt ook altijd eerst getest worden of het laatst gevonden contourpunt zich op een rand van de matrix bevindt. In figuur 3.4.5a is een $\emptyset:PY+1$ bij $\emptyset:PX+1$ matrix gedeclareerd, in figuur 3.4.5b een PY bij PX matrix.



JA		M.EQ. 1		NEE	
		N=N-1			
N.EQ.-2		JA		M.EQ.-1	
JA		NEE		N=N+1	
M=∅		N.EQ. 2		M.EQ. 1	
N=-1		JA		NEE	
		M=∅		M=1	
		N=1		M=-1	
O=K+M					
P=L+N					

figuur 3.4.5a

K.EQ.1		JA		L.EQ.PX		JA		K.EQ.PY		M.EQ.∅		L.EQ.1	
JA		NEE		M.EQ.∅		JA		N.EQ.-1		N.EQ.1		M.EQ.1	
N=-1		M=-1		M=∅		N=∅		JA		NEE		N=N-1	
N.EQ.-2		JA		NEE		M=∅		N=-1		JA		NEE	
O=K+M												P=L+N	

figuur 3.4.5b

In het bovenstaande hebben we gezien dat wanneer er een M bij N matrix wordt gedeclareerd, zowel voor het bepalen van het beginpunt als voor het nemen van het volgende te onderzoeken punt, altijd eerst moet worden

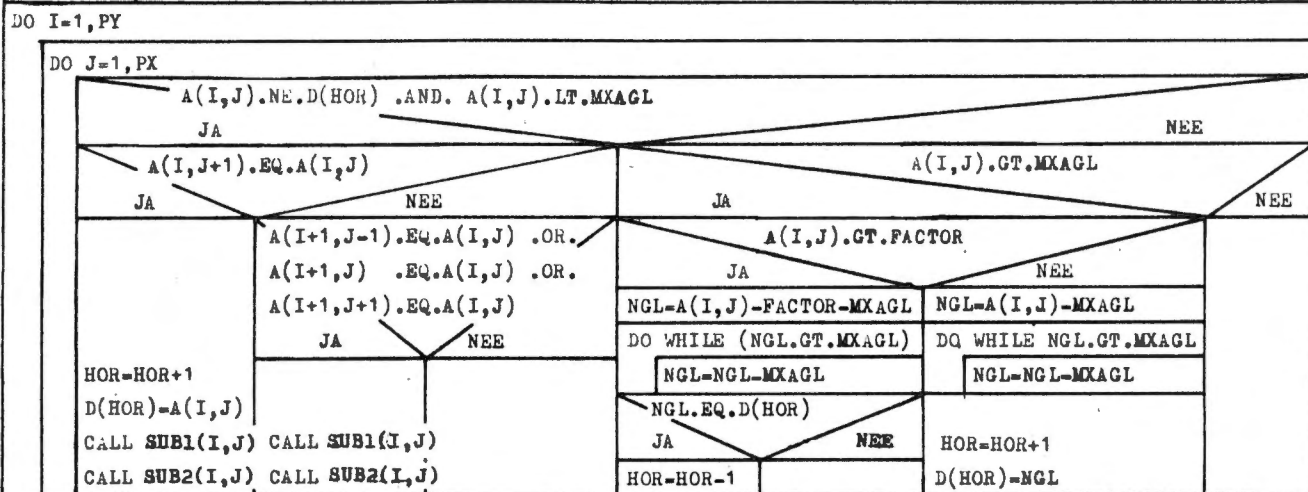
nagegaan of het laatst gevonden punt zich op één van de randen van de matrix bevindt. Dit gaat ten koste van de snelheid van het algoritme. Wanneer in plaats van een 128 bij 64 een $\emptyset:129$ bij $\emptyset:65$ matrix wordt gedeclareerd dan is voor de matrix de voor deze matrix 4,7% meer geheugenruimte nodig. Dit percentage neemt weer af doordat de geheugenruimte die het algoritme vereist kleiner is geworden.

We beginnen nu met het algoritme voor het beschrijven van het beeld. Het hoofdprogramma is 'het vinden van contouren'. Indien in het hoofdprogramma een contourpunt wordt gevonden waarvan de contour nog niet is gevolgd en de oppervlakte van het pas gevonden plateau minimaal twee beeldpunten groot is, dan wordt subroutine SUB1 aangeroepen. Over de grootte van het aantal beeldpunten komen we in hoofdstuk 4 nog nader op terug. Subroutine SUB1 volgt de contour voor de eerste keer. SUB1 heeft twee subroutines die GEHEEL en ZOEK zijn genoemd. GEHEEL bepaalt $\sum m_i x_i$, $\sum m_i y_i$ en $\sum m_i$ van het gehele plateau en ZOEK bepaalt het beginpunt voor het zoeken naar het volgende contourpunt. Wanneer de contour voor de eerste keer is gevolgd, wordt subroutine SUB2 aangeroepen die de contour voor de tweede keer volgt. SUB2 heeft ook twee subroutines. Subroutine DEEL bepaalt $\sum m_i x_i$, $\sum m_i y_i$ en $\sum m_i$ van de deelplateaus, ZOEK doet hetzelfde als bij subroutine SUB1.

Figuur 3.4.6 illustreert het structuurdiagram van het hoofdprogramma, figuur 3.4.7 de subroutines SUB1 en GEHEEL. Figuur 3.4.8 de subroutines SUB2 en DEEL. Tenslotte wordt voor het beschrijven van het beeld in figuur 3.4.10 het structuurdiagram ZOEK weergegeven. Figuur 3.4.11 geeft het structuurdiagram weer dat vanuit deze parameters het beeld weer wordt opgebouwd.

HOR=1
D(HOR)=0
AANTAL=0

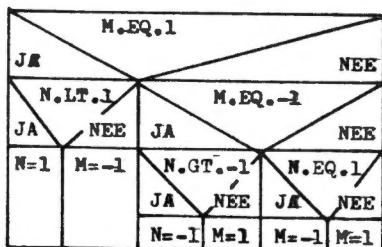
HET VINDEN VAN CONTOUREN
HOOFDPROGRAMMA



Betekeningen van de parameters:

- A = beeldmatrix.
- HOR = pointer
- D = extra matrix
- AANTAL= nummer van het plateau
- MXAGL = aantal waarden die in de oorspronkelijke matrix kunnen voorkomen
- FACTOR= verschil tussen een linker- en een rechtermarkering van hetzelfde plateau

figuur 3.4.6



BEPAAI BEGINPUNT VOOR HET
ZOEKEN NAAR HET VOLGEND
CONTOURPUNT
SUBROUTINE ZOEK [M.N]

figuur 3.4.10

```

AANTAL=AANTAL+1
B(AANTAL,1)=A(I,J)
G(2)=A(I,J)
G(1)=G(2)-MXAGL
A(I,J)=G(1)
E=I
N=1
A(I,J+1).EQ.G(2)
JA
NEE
L=J+1
M=0
L=J
M=1
C(1,1)=L(L+1)
C(1,2)=L.K
C(1,3)=L
O=K+M
E=L+N
ERWIT=0
    
```

HET BEPALEN VAN HET ZWAARTEPUNT
VAN HET GEHELE PLATEAU

SUBROUTINE SUB1(I,J)

figuur 3.4.7

Betekeningen van de parameters:

- G(1),---,G(4) = de waarden die aan de gemarkeerde contourpunten worden gegeven
- B(AANTAL,1) = intensiteitsniveau van het plateau
- B(AANTAL,2) = X-coördinaat van het zwaartepunt van het gehele plateau
- B(AANTAL,3) = Y-coördinaat
- C(1,1) = $\sum_{i=1}^N x_i$ van het gehele plateau
- C(1,2) = $\sum_{i=1}^N y_i$
- C(1,3) = $\sum_{i=1}^N 1$

$$C(1,1)=C(1,1)+P.N$$

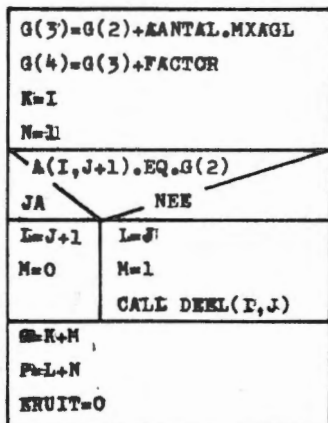
$$C(1,2)=C(1,2)+P.O$$

$$C(1,3)=C(1,3)+P$$

SUBROUTINE GEHEEL(O,P,N)

```

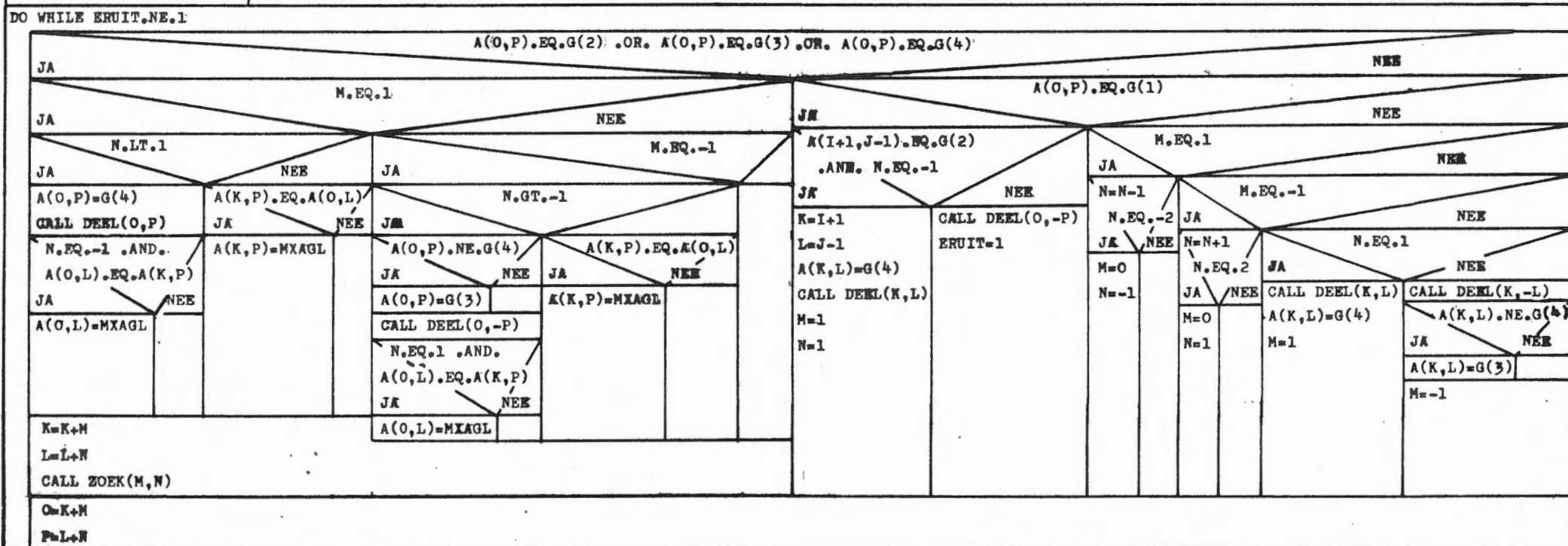
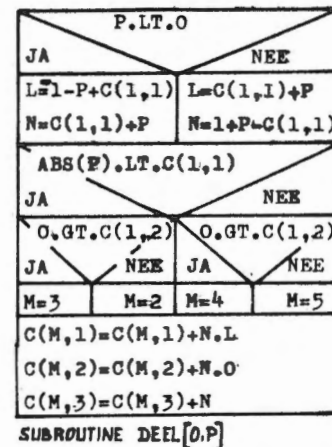
DO WHILE ERWIT.NE.1
  A(O,P).EQ.G(2)
  JA
  NEE
  M.EQ.-1 .AND. N.LT.1
  JA
  NEE
  CALL GEHEEL(O,P,1+P)
  M.EQ.-1 .AND. N.GT.-1
  JA
  NEE
  CALL GEHEEL(O,P,2-P)
  A(I+1,J-1).EQ.G(2) .AND. N.EQ.-1
  JA
  NEE
  CALL GEHEEL(I,J,1-J)
  N=N-1
  N.EQ.-2
  JA
  NEE
  M=0
  N=-1
  M.EQ.-1
  JA
  NEE
  M=N+1
  N.EQ.2
  JA
  NEE
  M=0
  N=1
  M.EQ.-1
  JA
  NEE
  CALL GEHEEL(K,L,1+L)
  CALL GEHEEL(K,L,1-L)
  K=K+M
  L=L+N
  CALL ZOEK(M,N)
  O=K+M
  P=L+N
  B(AANTAL,2)= C(1,1)/(C(1,3).2)
  B(AANTAL,3)= C(1,2)/C(1,3)
    
```



HET BEPALEN VAN DE ZWAARTEPUNTEN
EN OPPERVLAKKEN VAN DE DEELPLATEAUS

SUBROUTINE SUB2[I, J]

figuur 3.4.8

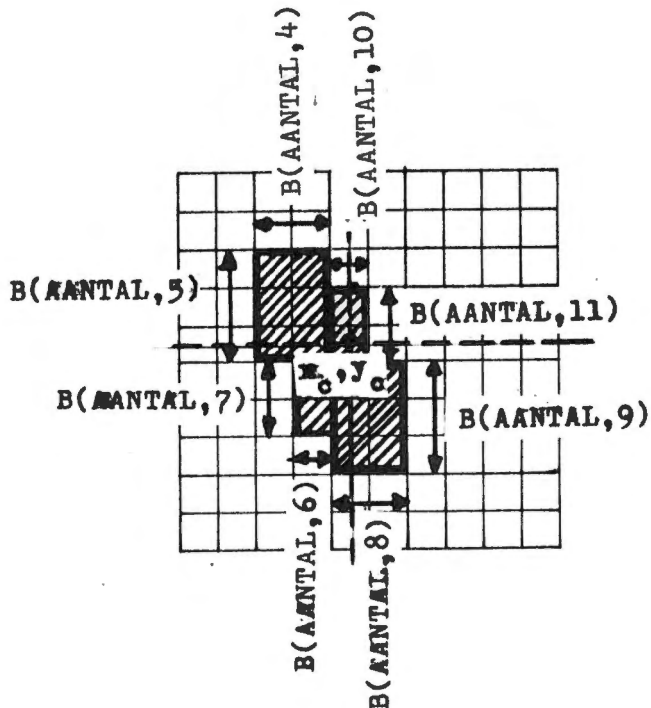


DO M=2,5			
N=2,M			
B(M,3).EQ.0			
JA	NEE		
B(AAN,N)=0	REX=C(M,1)/(C(M,3).2)		
B(AAN,N+1)=0	REY=C(M,2)/C(M,3)		
M.EQ.2			
JAK	NEE		
HOEK=C(1,2)-REY		M.EQ.3	
C(1,1)-1-REX		NEE	
HOEK=REY-C(1,2)+1		M.EQ.4	
C(1,1)-1-REX		NEE	
HOEK=REY-C(1,2)+1		HOEK=C(1,2)-REY	
REX-C(1,1)		REX-C(1,1)	
HOEK.LT.1/PX			
JAK	NEE		
B(AAN,N)=C(M,3)	HOEK.GT.PY		
B(AAN,N+1)=1	JA	NEE	
	B(AAN,N)=1	B(AAN,N+1)=SQRT(C(M,3).HOEK)	
	B(AAN,N+1)=C(M,3)	B(AAN,N)=C(M,3)/B(AAN,N+1)	

Betekeningen van de parameters:

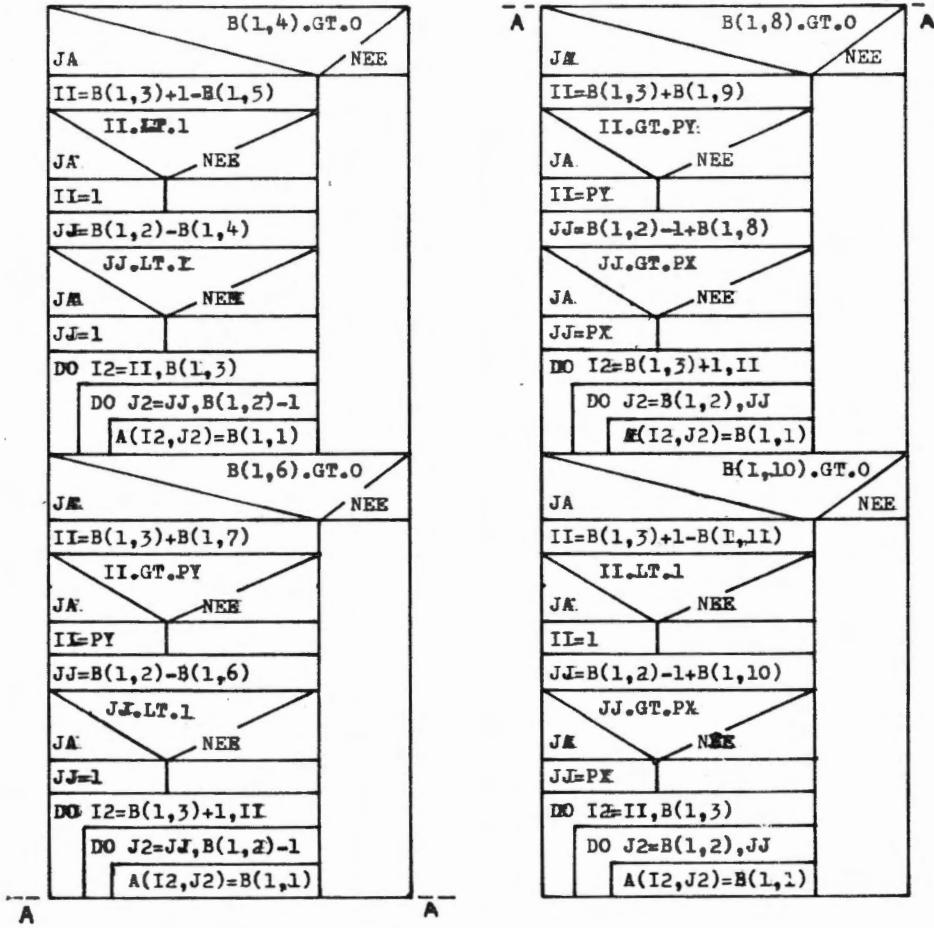
B(AANTAL,4),-----,B(AANTAL,11) = de rechthoekszijden (zie figuur 3.4.9) van een plateau

figuur 3.4.8



figuur 3.4.9

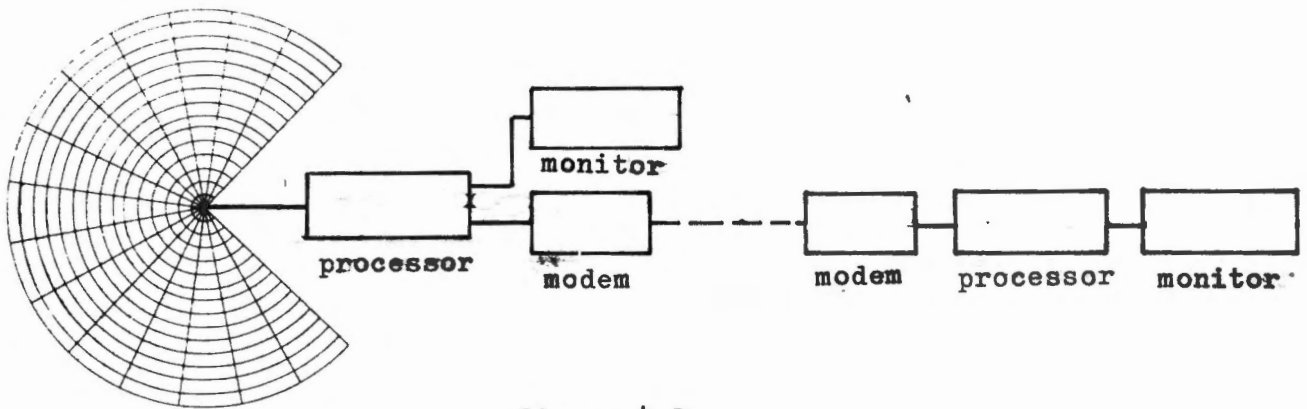
HET BEELD WEER OPBOUWEN



figuur 3.4.11

4. Aspecten betreffende de communicatie tussen de grondstations

Door weersveranderingen kan de reflectivity als functie van de tijd snelle veranderingen ondergaan. Dit gaat op zijn beurt weer gepaard met snelle dempingsveranderingen. Om de weersveranderingen te kunnen volgen zal de rondzoekradar permanent in bedrijf moeten zijn. Het gevolg hiervan is dat slechts een beperkte tijdruimte beschikbaar is om de parameters die het beeld beschrijven over te zenden. Mede door de beperkte bandbreedte van het transmissiekanaal is datareductie noodzakelijk. Hierdoor kan slechts een beperkt aantal plateaus worden overgezonden. Om beneden dit maximum aantal over te zenden plateaus te blijven, worden twee methoden besproken. Als voorbeeld is het schema van het systeem geschetst in figuur 4.1. In dit voorbeeld is van een aantal veronderstellingen uitgegaan, omdat nog niet bekend is met welke apparatuur het systeem zal worden opgebouwd.



figuur 4.1

Deze veronderstellingen zijn:

- De beschikbare verwerkingstijd is 5 s per beeld.
- De grootte van de beeldmatrix is 128 bij 64 beeldpunten.
- De intensiteit van een beeldpunt kan 16 verschillende waarden aannemen.
- Met behulp van de parameters kan het beeld oneindig snel weer worden gereconstrueerd.
- Er is een (telefonie)-kanaal beschikbaar met een capaciteit van 9600 bit/s.
- De invloed van de storingen op het kanaal wordt te verwaarlozen verondersteld zodat geen kanaalcodering nodig is.
- Aangezien de kansen van optreden van de waarden van de parameters nog niet bekend zijn, wordt broncodering niet toegepast.

Het aantal bits per plateau bedraagt:

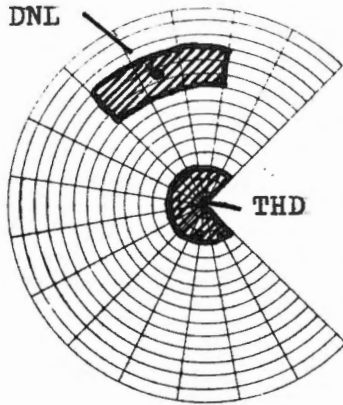
plaatscoördinaten van het zwaartepunt	$7+6 = 13$ bit
de 8 rechthoekszijden	$4 \cdot (7+6) = 52$ bit
de hoogte van het plateau	$= \underline{4}$ bit
	69 bit/plateau

Per beeld staan ter beschikking $5 \times 9600 = 48$ kbit. Van maximaal $48000/69 = 695$ plateaus kunnen de parameters worden overgezonden. Aangezien een beeldpunt 16 verschillende intensiteitsniveaus kan aannemen, kunnen de 8 buurpunten van een beeldpunt andere intensiteitsniveaus hebben dan dat van het beeldpunt. Hierdoor kunnen er maximaal $128 \times 64 = 8192$ plateaus in het beeld aanwezig zijn. De contour van een plateau wordt, zoals in hoofdstuk 3 is beschreven, tweemaal gevolgd om de parameters van het plateau te bepalen. De eerste keer worden van het plateau de grootte van het oppervlak en de positie van het zwaartepunt bepaald en de tweede keer worden van de deelplateaus de grootte van de oppervlakken en de posities van de zwaartepunten bepaald. Door vóór de aanvang van het voor de tweede keer volgen van de contour een drempeltoets ten aanzien van de grootte van het oppervlak aan te leggen, kan worden voorkomen dat het maximum aantal plateaus wordt overschreden. In het voorbeeld is de drempel $8192/695 = 12$ beeldpunten. Boven deze drempel worden de parameters van het plateau bepaald beneden deze niet. Deze methode heeft de volgende voordelen:

- Het ruisachtige karakter van het gereconstrueerde beeld wordt verminderd.
- Indien de te verwachten beelden erbij worden betrokken, dan zal de verwachting zijn dat een bui die in de beeldpuntenmatrix één rij hoog is en 13 kolommen lang niet of nauwelijks zal voorkomen. Hierdoor kan het algoritme sneller zijn bijvoorbeeld door de matrix niet rij voor rij maar om de andere rij te onderzoeken.

Bij de tweede methode wordt ook het zwaartepunt van het gehele plateau erbij betrokken. Wanneer het zwaartepunt zich binnen een bepaalde omgeving van de grondstations bevindt, worden de parameters van het plateau bepaald. Wanneer het zich buiten een bepaalde omgeving van de grondstations bevindt, wordt met een drempeltoets ten aanzien van de grootte van het oppervlak van het plateau beslist of de parameters van dit plateau worden bepaald. Hierdoor

komen voor het diversity onderzoek meer significante gegevens beschikbaar. In figuur 4.2 is de werking van deze methode geschetst. De gearceerde gebieden geven de omgevingen van de grondstations aan.



figuur 4.2

5. Conclusies

Teneinde grote hoeveelheden data afkomstig van de rondzoekradar te vermijden, wordt een beeld geanalyseerd en beschreven met behulp van een aantal parameters, zodat getracht wordt alleen voor het site diversity onderzoek significante informatie beschikbaar te laten komen.

Expliciet kunnen de contourpunten van de plateaus worden opgespoord door het oorspronkelijke beeld door een gradiënt-operator te sturen, zoals in paragraaf 2.4 is beschreven. Het grote nadeel van deze methode, ten opzichte van de methode die in het oorspronkelijke beeld de contourpunten opspoorde, is dat in een beeld met ruis de contourpunten van een plateau niet meer bekend zijn. We kunnen concluderen dat deze methode minder geschikt voor dit doel is.

Bij het volgen van de contour van een plateau heeft procedure A verschillende nadelen ten opzichte van procedure B, die beide in paragraaf 2.6 zijn beschreven. Een nadeel is dat meer plateaus worden gevonden dan in het oorspronkelijke beeld aanwezig zijn. Dit komt doordat niet alle buurpunten van een contourpunt worden onderzocht of het een contourpunt is. Een ander nadeel is dat in het algemeen minder goede parameters worden verkregen dan bij procedure B, dat veroorzaakt wordt doordat zich binnen een contour contouren kunnen bevinden. Ook hier kunnen we de conclusie trekken dat procedure A minder geschikt is. Het algoritme kan sneller worden gemaakt door in plaats van een PY bij PX matrix een $\emptyset:PY+1$ bij $\emptyset:PX+1$ matrix te declareren. Hierdoor hoeft zowel voor het bepalen van een beginpunt als voor het nemen van het volgende te onderzoeken punt niet eerst worden nagegaan of het laatst gevonden contourpunt zich niet op één van de randen van de matrix bevindt. Een tweede mogelijkheid om het algoritme te versnellen is door de te verwachten beelden erbij te betrekken aangezien bijvoorbeeld buien met een hoogte van één rij niet of nauwelijks voorkomen. We kunnen dan de matrix bijvoorbeeld om de andere rij te onderzoeken.

De prijs die voor de datareductie moet worden betaald, is dat er vervorming optreedt. Indien mocht blijken dat de vervorming te groot is, kan vermindering van de vervorming worden verkregen door een deelplateau als een apart plateau te beschouwen. Men kan zich afvragen of met een andere benadering van de vorm van een plateau betere resultaten kunnen worden verkregen.

LITERATUURLIJST

1. F.M.-C.W. Delft atmospheric research radar
L.P. Ligthart, L.R. Nieuwkerk
IEE Proc., Vol. 127, December 1980
2. Voorstel Rondzoekradarproject
Samenwerkingsproject PTT/DNL en THD
L.P. Ligthart, L.R. Nieuwkerk
December 1983
3. Microgolfaspecten van de satellietcommunicatie
L. Krul
Collegedictaat TH Delft
4. Granulatiereductie in beelden van regenbuien, verkregen met
begulp van de F.M.-C.W. radar, door middel van in 'real-time'
werkende beeldbewerkingsalgorithmen
R. Paliwoda
Afstudeerverslag 05-1-565-28-208
5. Onderzoek naar polaire afbeelding van Radardata afkomstig van
de Delft Atmospheric Research Radar
C.J. van Waveren
Afstudeerverslag 565-28-233
6. Informatietheorie II
I.J. Boxma
Collegedictaat TH Delft
7. Coderen
D.E. Boekee
Collegedictaat TH Delft
8. Digital picture processing
A. Rosenfeld, A.C. Kak
Academic Press

9. Inleiding in de digitale beeldverwerking
J.J. Gerbrands
Collegedictaat TH Delft

10. Introduction
Special Issue on Image Bandwith Compression
A. Habibi
IEEE Transactions on Communications, vol. COM-25, N^o 11, November 1977

11. Image Transmission by Two-Dimensional Contour Coding
D.N. Graham
Proceedings of the IEEE, vol. 55, N^o 3, March 1967

12. Course on Pattern Recognition and Image Processing 1978
taught by the Staff of the Pattern Recognition Section of the
Group Signal/SystemTechnique
Provisional edition (reprinted in 1981)

```

00001 FTN7X,L
00002 PROGRAM WIEL
00003 *****
00004 * HET VEREENVOUDIGD WEERGEVEN VAN CONTOUREN
00005 * opbouw van het algoritme:
00006 * Hoofdprogramma: het vinden van contouren
00007 * bevat subroutines SUB1,SUB2
00008 * subroutine SUB1
00009 * bevat subroutines GEHEEL,ZOEK
00010 * subroutine SUB2
00011 * bevat subroutines DEEL en ZOEK
00012 *****
00013 IMPLICIT NONE
00014 INTEGER*2 I !
00015 INTEGER*2 J !
00016 INTEGER*2 K ! K,L coördinaten van het laatst gevonden
00017 INTEGER*2 L ! contourpunt.
00018 INTEGER*2 M ! M=-1,0,+1
00019 INTEGER*2 N ! N=-1,0,+1
00020 INTEGER*2 O ! O=K+M A(O,P) wordt onderzocht of het een
00021 INTEGER*2 P ! P=L+N contourpunt is.
00022 INTEGER*2 HOR ! POINTER
00023 INTEGER*2 NGL !
00024 INTEGER*2 MXAGL ! MAXIMUM aantal waarden die in de oorspron-
00025 C ! kelijke matrix kunnen voorkomen (0-MXAGL)
00026 INTEGER*2 FACTOR ! FACTOR verschil tussen LM- en RM-punt
00027 INTEGER*2 AANTAL ! AANTAL plateaus van een beeld
00028 INTEGER*2 ERUIT ! ERUIT=JA ? uit de DO-loop
00029 INTEGER*2 PY ! aantal rijen beeldmatrix
00030 INTEGER*2 PX ! aantal kolommen beeldmatrix
00031 INTEGER*2 A(0:11,0:25)! beeldmatrix
00032 INTEGER*2 B(100,11) !
00033 INTEGER*2 C(5,3) !
00034 INTEGER*2 D((24/2)+1) ! D(HOR) mag niet gevonden worden
00035 INTEGER*2 G(4) ! G(1) beginpunt; G(2) plateauhoogte
00036 C ! G(3) LM-punt; G(4) RM-punt
00037 REAL REX,REY,HOEK
00038 COMMON/BLOK/PY,PX,AANTAL,MXAGL,FACTOR,A,B,C,G
00039 *
00040 *
00041 PY=10
00042 PX=24
00043 MXAGL=8
00044 FACTOR=100
00045 DO I=0,PY+1
00046 DO J=0,PX+1
00047 A(I,J)=0
00048 END DO
00049 END DO
00050 WRITE(1,('WILT U DE MATRIX [10,24] VULLEN:'))
00051 READ(1,('(24(I))')) ((A(I,J),J=1,24),I=1,10)
00052 WRITE(1,('OORSPRONKELIJKE MATRIX:'))
00053 WRITE(1,('(24(I3))')) ((A(I,J),J=1,24),I=1,10)
00054 HOR=1
00055 D(HOR)=0
00056 AANTAL=0
00057 DO I=1,PY
00058 DO J=1,PX
00059 IF (A(I,J).NE.D(HOR).AND.A(I,J).LT.MXAGL) THEN
00060 IF (A(I,J+1).EQ.A(I,J)) THEN

```

```

00061          HOR=HOR+1
00062          D(HOR)=A(I,J)
00063          CALL SUB1(I,J)
00064          CALL SUB2(I,J)
00065          ELSE IF (A(I+1,J-1).EQ.A(I,J).OR.
00066 +             A(I+1,J).EQ.A(I,J).OR.
00067 +             A(I+1,J+1).EQ.A(I,J) ) THEN
00068              CALL SUB1(I,J)
00069              CALL SUB2(I,J)
00070          END IF
00071          ELSE IF (A(I,J).GT.MXAGL) THEN
00072              IF (A(I,J).GT.FACTOR) THEN
00073                  NGL=A(I,J)-FACTOR-MXAGL
00074                  DO WHILE (NGL.GT.MXAGL)
00075                      NGL=NGL-MXAGL
00076                  END DO
00077                  IF (NGL.EQ.D(HOR)) THEN
00078                      HOR=HOR-1
00079                  END IF
00080              ELSE
00081                  NGL=A(I,J)-MXAGL
00082                  DO WHILE (NGL.GT.MXAGL)
00083                      NGL=NGL-MXAGL
00084                  END DO
00085                  HOR=HOR+1
00086                  D(HOR)=NGL
00087              END IF
00088          END IF
00089      END DO
00090  END DO
00091 *
00092 *
00093 *****
00094 *   HET WEER OPBOUWEN VAN HET BEELD
00095 *****
00096      DO I=1,PY
00097          DO J=1,PX
00098              A(I,J)=0
00099          END DO
00100      END DO
00101      WRITE(1,('PARAMETERS DIE HET BEELD BESCHRIJVEN:'))
00102      WRITE(1,(' INT. X Y M1 M2 M3 M4 M5 M6 M7 M8'))
00103      DO I=1,AANTAL
00104          WRITE(1,('11(I4)')) (B(I,J),J=1,11)
00105      END DO
00106      DO I=1,AANTAL
00107          IF (B(I,4).GT.0) THEN
00108              M=B(I,3)+1-B(I,5)
00109              IF (M.LT.1) THEN
00110                  M=1
00111              END IF
00112              N=B(I,2)-B(I,4)
00113              IF (N.LT.1) THEN
00114                  N=1
00115              END IF
00116              DO O=M,B(I,3)
00117                  DO P=N,B(I,2)-1
00118                      A(O,P)=B(I,1)
00119                  END DO
00120              END DO

```

```

00121      END IF
00122      IF (B(I,6).GT.0) THEN
00123          M=B(I,3)+B(I,7)
00124          IF (M.GT.PY) THEN
00125              M=PY
00126          END IF
00127          N=B(I,2)-B(I,6)
00128          IF (N.LT.1) THEN
00129              N=1
00130          END IF
00131          DO O=B(I,3)+1,M
00132              DO P=N,B(I,2)-1
00133                  A(O,P)=B(I,1)
00134              END DO
00135          END DO
00136      END IF
00137      IF (B(I,8).GT.0) THEN
00138          M=B(I,3)+B(I,9)
00139          IF (M.GT.PY) THEN
00140              M=PY
00141          END IF
00142          N=B(I,2)-1+B(I,8)
00143          IF (N.GT.PX) THEN
00144              N=PX
00145          END IF
00146          DO O=B(I,3)+1,M
00147              DO P=B(I,2),N
00148                  A(O,P)=B(I,1)
00149              END DO
00150          END DO
00151      END IF
00152      IF (B(I,10).GT.0) THEN
00153          M=B(I,3)+1-B(I,11)
00154          IF (M.LT.1) THEN
00155              M=1
00156          END IF
00157          N=B(I,2)-1+B(I,10)
00158          IF (N.GT.PX) THEN
00159              N=PX
00160          END IF
00161          DO O=M,B(I,3)
00162              DO P=B(I,2),N
00163                  A(O,P)=B(I,1)
00164              END DO
00165          END DO
00166      END IF
00167  END DO
00168  WRITE(1,('("VERKREGEN MATRIX:")'))
00169  WRITE(1,('(" 1  2  3  4  5  6  7  8  9 10 11 12 13 14 15 16 17 18
+19 20 21 22 23 24",/)'))
00171  WRITE(1,('24(I3)')) ((A(I,J),J=1,PX),I=1,PY)
00172  END
00173 *
00174 *
00175 *****
00176 *   EERSTE KEER DE CONTOUR VOLGEN VOOR;
00177 *   - HET BEPALEN VAN HET ZWAARTEPUNT VAN DE GEHELE PLATEAU
00178 *****
00179  SUBROUTINE SUBI(I,J)
00180  INTEGER*2 I,J,K,L,M,N,O,P

```

```

00181      INTEGER*2 PY,PX,MXAGL
00182      INTEGER*2 FACTOR,AANTAL,ERUIT
00183      INTEGER*2 A(0:11,0:25),B(100,11),C(5,3),G(4)
00184      COMMON/BLOK/PY,PX,AANTAL,MXAGL,FACTOR,A,B,C,G
00185 *
00186      C(1,1)=0
00187      C(1,2)=0
00188      C(1,3)=0
00189      K=1
00190      N=1
00191      IF (A(I,J+1).EQ.A(I,J)) THEN
00192          L=J+1
00193          M=0
00194      ELSE
00195          L=J
00196          M=1
00197          CALL GEHEEL(I,1+J,J,C)
00198      END IF
00199      AANTAL=AANTAL+1
00200      DO X=1,11
00201          B(AANTAL,X)=0
00202      END DO
00203      B(AANTAL,1)=A(I,J)
00204      G(2)=A(I,J)
00205      G(1)=G(2)-MXAGL
00206      A(I,J)=G(1)
00207      O=K+M
00208      P=L+N
00209      ERUIT=0
00210      DO WHILE (ERUIT.NE.1)
00211          IF (A(O,P).EQ.G(2)) THEN
00212              IF (M.EQ.1.AND.N.LT.1) THEN
00213                  CALL GEHEEL(O,P+1,P,C)
00214              ELSE IF (M.EQ.-1.AND.N.GT.-1) THEN
00215                  CALL GEHEEL(O,P,1-P,C)
00216              END IF
00217              K=O
00218              L=P
00219              CALL ZOEK(M,N)
00220          ELSE IF (A(O,P).EQ.G(1)) THEN
00221              IF (A(I+1,J-1).EQ.G(2).AND.N.EQ.-1) THEN
00222                  K=I+1
00223                  L=J-1
00224                  M=1
00225                  N=1
00226                  CALL GEHEEL(K,1+L,L,C)
00227              ELSE
00228                  CALL GEHEEL(I,J,1-J,C)
00229                  ERUIT=1
00230              END IF
00231          ELSE IF (M.EQ.1) THEN
00232              N=N-1
00233              IF (N.EQ.-2) THEN
00234                  M=0
00235                  N=-1
00236              END IF
00237          ELSE IF (M.EQ.-1) THEN
00238              N=N+1
00239              IF (N.EQ.2) THEN
00240                  M=0

```

```

00241             N=1
00242             END IF
00243             ELSE IF (N.EQ.1) THEN
00244                 CALL GEHEEL(O,P,L,C)
00245                 M=1
00246             ELSE
00247                 CALL GEHEEL(O,L,-P,C)
00248                 M=-1
00249             END IF
00250             O=K+M
00251             P=L+N
00252             END DO
00253             B(AANTAL,2)=NINT(REAL(C(1,1))/REAL(C(1,3)*2))
00254             C(1,1)=B(AANTAL,2)
00255             B(AANTAL,3)=INT((REAL(C(1,2))/REAL(C(1,3))))
00256             C(1,2)=B(AANTAL,3)
00257             RETURN
00258             END
00259 *
00260 *
00261 *****
00262 *     BEPALEN ZWAARTEPUNT GEHELE PLATEAU
00263 *     behorende bij subroutine GEHEEL
00264 *****
00265     SUBROUTINE GEHEEL(O,P,M,C)
00266     IMPLICIT NONE
00267     INTEGER*2 O,P,M
00268     INTEGER*2 C(5,3)
00269 *
00270     C(1,1)=C(1,1)+P*M
00271     C(1,2)=C(1,2)+O*M
00272     C(1,3)=C(1,3)+M
00273     RETURN
00274     END
00275 *
00276 *
00277 *****
00278 *     TWEDE KEER DE CONTOUR VOLGEN VOOR:
00279 *     - HET BEPALEN VAN DE DEELOPPERVLAKKEN VAN EEN PLATEAU
00280 *     - HET BEPALEN VAN DE ZWAARTEPUNTEN VAN DE DEELOPPERVLAKKEN
00281 *     - HET MARKEREN VAN DE CONTOURPUNTEN
00282 *     - HET MARKEREN VAN PUNTEN BUITEN DE CONTOUR
00283 *****
00284     SUBROUTINE SUB2(I,J)
00285     IMPLICIT NONE
00286     INTEGER*2 I,J,K,L,M,N,O,P
00287     INTEGER*2 PY,PX,MXAGL
00288     INTEGER*2 FACTOR,AANTAL,ERUIT
00289     INTEGER*2 A(0:11,0:25),B(100,11),C(5,3),G(4)
00290     REAL      REX,REY,HOEK
00291     COMMON/BLOK/PY,PX,AANTAL,MXAGL,FACTOR,A,B,C,G
00292 *
00293     DO K=2,5
00294         DO L=1,3
00295             C(K,L)=0
00296         END DO
00297     END DO
00298     G(3)=G(2)+AANTAL*MXAGL
00299     G(4)=G(3)+FACTOR
00300     K=I

```



```

00301      N=1
00302      IF (A(I,J+1).EQ.G(2)) THEN
00303          L=J+1
00304          M=0
00305      ELSE
00306          L=J
00307          M=1
00308          CALL DEEL(K,L,C)
00309      END IF
00310      O=K+M
00311      P=L+N
00312      ERUIT=0
00313      DO WHILE (ERUIT.NE.1)
00314          IF (A(O,P).EQ.G(2).OR.
00315      +      A(O,P).EQ.G(3).OR.
00316      +      A(O,P).EQ.G(4) ) THEN
00317              IF (M.EQ.1) THEN
00318                  IF (N.LT.1) THEN
00319                      A(O,P)=G(4)
00320                      CALL DEEL(O,P,C)
00321                      IF (N.EQ.-1.AND.A(O,L).EQ.A(K,P)) THEN
00322                          A(O,L)=MXAGL
00323                      END IF
00324                      ELSE IF (A(K,P).EQ.A(O,L)) THEN
00325                          A(K,P)=MXAGL
00326                      END IF
00327                      ELSE IF (M.EQ.-1) THEN
00328                          IF (N.GT.-1) THEN
00329                              IF (A(O,P).NE.G(4)) THEN
00330                                  A(O,P)=G(3)
00331                              END IF
00332                              CALL DEEL(O,-P,C)
00333                              IF (N.EQ.1.AND.A(O,L).EQ.A(K,P)) THEN
00334                                  A(O,L)=MXAGL
00335                              END IF
00336                              ELSE IF (A(K,P).EQ.A(O,L)) THEN
00337                                  A(K,P)=MXAGL
00338                              END IF
00339                          END IF
00340                          K=O
00341                          L=P
00342                          CALL ZOEK(M,N)
00343                      ELSE IF (A(O,P).EQ.G(1)) THEN
00344                          IF (A(O+1,P-1).EQ.G(2).AND.N.EQ.-1) THEN
00345                              K=O+1
00346                              L=P-1
00347                              A(K,L)=G(4)
00348                              CALL DEEL(K,L,C)
00349                              M=1
00350                              N=1
00351                          ELSE
00352                              CALL DEEL(O,-P,C)
00353                              ERUIT=1
00354                          END IF
00355                      ELSE IF (M.EQ.1) THEN
00356                          N=N-1
00357                      IF (N.EQ.-2) THEN
00358                          M=0
00359                          N=-1
00360                      END IF

```

```

00361     ELSE IF (M.EQ.-1) THEN
00362         N=N+1
00363         IF (N.EQ.2) THEN
00364             M=0
00365             N=1
00366         END IF
00367     ELSE IF (N.EQ.1) THEN
00368         A(K,L)=G(4)
00369         CALL DEEL(K,L,C)
00370         M=1
00371     ELSE
00372         IF (A(K,L).NE.G(4)) THEN
00373             A(K,L)=G(3)
00374         END IF
00375         CALL DEEL(K,-L,C)
00376         M=-1
00377     END IF
00378     O=K+M
00379     P=L+N
00380 END DO
00381 DO M=2,5
00382     N=2*M
00383     IF (C(M,3).EQ.0) THEN
00384         B(AANTAL,N)=0
00385         B(AANTAL,N+1)=0
00386     ELSE
00387         REX=REAL(C(M,1))/(REAL(C(M,3)*2))
00388         REY=REAL(C(M,2))/REAL(C(M,3))
00389         IF (M.EQ.2) THEN
00390             HOEK=(REAL(C(1,2))-REY)/(REAL(C(1,1)-1)-REX)
00391         ELSE IF (M.EQ.3) THEN
00392             HOEK=(REY-REAL(C(1,2)+1))/(REAL(C(1,1)-1)-REX)
00393         ELSE IF (M.EQ.4) THEN
00394             HOEK=(REY-REAL(C(1,2)+1))/(REX-REAL(C(1,1)))
00395         ELSE
00396             HOEK=(REAL(C(1,2))-REY)/(REX-REAL(C(1,1)))
00397         END IF
00398         IF (HOEK.GT.REAL(PY)) THEN
00399             B(AANTAL,N+1)=C(M,3)
00400             B(AANTAL,N)=1
00401         ELSE IF (HOEK.LT.1./REAL(PX)) THEN
00402             B(AANTAL,N)=C(M,3)
00403             B(AANTAL,N+1)=1
00404         ELSE IF (HOEK.GT.REAL(1)) THEN
00405             B(AANTAL,N+1)=NINT(SQRT(REAL(C(M,3))*HOEK))
00406             B(AANTAL,N)=NINT(REAL(C(M,3))/REAL(B(AANTAL,N+1)))
00407         ELSE
00408             B(AANTAL,N)=NINT(SQRT(REAL(C(M,3))/HOEK))
00409             B(AANTAL,N+1)=NINT(REAL(C(M,3))/REAL(B(AANTAL,N)))
00410         END IF
00411     END IF
00412 END DO
00413 RETURN
00414 END
00415 *
00416 *
00417 *****
00418 *   HET BEPALEN VAN DE DEELOPPERVLAKKEN EN DE ZWAARTEPUNTEN ERVAN
00419 *   behorende bij subroutine DEEL
00420 *****

```

```

00421      SUBROUTINE DEEL(O,P,C)
00422      IMPLICIT NONE
00423      INTEGER*2 M,N,L,O,P
00424      INTEGER*2 C(5,3)
00425 *
00426      IF (P.LT.0) THEN
00427          L=C(1,1)-P-1
00428          N=C(1,1)+P
00429      ELSE
00430          L=C(1,1)+P
00431          N=1+P-C(1,1)
00432      END IF
00433      IF (ABS(P).LT.C(1,1)) THEN
00434          IF (O.GT.C(1,2)) THEN
00435              M=3
00436          ELSE
00437              M=2
00438          END IF
00439      ELSE IF (O.GT.C(1,2)) THEN
00440          M=4
00441      ELSE
00442          M=5
00443      END IF
00444      C(M,1)=C(M,1)+N*L
00445      C(M,2)=C(M,2)+N*O
00446      C(M,3)=C(M,3)+N
00447      RETURN
00448      END
00449 *
00450 *
00451 *****
00452 *      EEN CONTOURPUNT IS GEVONDEN:
00453 *      BEPAAL BEGINPUNT VOOR HET ZOEKEN NAAR HET VOLGEND CONTOURPUNT
00454 *      behorende bij de subroutines GEHEEL en DEEL
00455 *****
00456      SUBROUTINE ZOEK(M,N)
00457      IMPLICIT NONE
00458      INTEGER*2 M,N
00459 *
00460      IF (M.EQ.1) THEN
00461          IF (N.LT.1) THEN
00462              N=1
00463          ELSE
00464              M=-1
00465          END IF
00466      ELSE IF (M.EQ.-1) THEN
00467          IF (N.GT.-1) THEN
00468              N=-1
00469          ELSE
00470              M=1
00471          END IF
00472      ELSE IF (N.EQ.1) THEN
00473          M=-1
00474      ELSE
00475          M=1
00476      END IF
00477      RETURN
00478      END

```