Eindhoven University of Technology

MASTER

Berekening van binaire amplitude hologrammen met een optimale diffractie-efficientie

Kuipers, P.G.M.

Award date:
1992

Link to publication

Disclaimer
This document contains a student thesis (bachelor's or master's), as authored by a student at Eindhoven University of Technology. Student theses are made available in the TU/e repository upon obtaining the required degree. The grade received is not published on the document as presented in the repository. The required complexity or quality of research of student theses may vary by program, and the required minimum study period may vary in duration.

General rights
Copyright and moral rights for the publications made accessible in the public portal are retained by the authors and/or other copyright owners and it is a condition of accessing publications that users recognise and abide by the legal requirements associated with these rights.

- Users may download and print one copy of any publication from the public portal for the purpose of private study or research.
- You may not further distribute the material or use it for any profit-making activity or commercial gain

Take down policy
If you believe that this document breaches copyright please contact us providing details, and we will remove access to the work immediately and investigate your claim.

Download date: 20. Jul. 2025

TECHNISCHE UNIVERSITEIT EINDHOVEN

FACULTEIT ELEKTROTECHNIEK

VAKGROEP Theoretische Elektrotechniek

Berekening van binaire amplitude hologrammen met een optimale diffractie-efficiëntie

door

P.G.M. Kuipers

ET-4-92

Verslag van een afstudeeronderzoek, verricht in de vakgroep ET, onder leiding van ir. P.M.T. van den Bulck en prof.dr.-ing. H.J. Butterweck, in de periode juli 1991 - februari 1992.

Eindhoven, 4 maart 1992.

De faculteit Elektrotechniek van de Technische Universiteit Eindhoven aanvaardt geen aansprakelijkheid voor de inhoud van stage- en afstudeerverslagen.

Samenvatting

Het doel van computer gegenereerde hologrammen is om een bepaalde lichtverdeling in het Fourier-vlak te realiseren. De kwaliteit van het gegenereerde hologram wordt bepaald door zowel de intensiteit van de lichtverdeling als de mate waarin de lichtverdeling overeenkomt met de gewenste lichtverdeling.

De intensiteit van de lichtverdeling wordt weergegeven door de diffractie-efficiëntie van het hologram. In dit verslag wordt gekeken hoe een aantal aanwezige vrijheden in het Fourier-vlak benut moeten worden om deze diffractie-efficiëntie te optimaliseren.

Om het hologram met behulp van een laserwriter te kunnen fabriceren moet deze uit een binaire pixelverdeling bestaan. Hiervoor wordt gebruik gemaakt van een iteratieve kwantisatie-methode. Deze kwantisatie-methode zal zodanig uitgewerkt worden dat de kwantisatie-ruis op het object zo klein mogelijk wordt.

Het totaal-algoritme voor de berekening van een binair amplitude-hologram wordt dan opgebouwd uit de volgende drie stappen.

1. Het berekenen van het initiële hologram.
2. Het iteratief verhogen van het lichtvermogen dat door het hologram doorgelaten wordt.
3. Het iteratief kwantiseren van het hologram.

Al deze stappen worden zodanig geoptimaliseerd dat het uiteindelijke hologram een zo groot mogelijke diffractie-efficiëntie en signaal-ruis-verhouding heeft.

Lijst van symbolen en afkortingen

C	(complexe) constante
f	brandpuntafstand
K_g	golfgetal
$N \cdot N$	grootte hologram
$\Delta u \cdot \Delta v$	grootte object
(u1,v1)	hoekpunt van object in de reconstructie
$\phi_i(x,y)$	lichtverdeling net na het hologram
$\phi_u(u,v)$	lichtverdeling in het reconstructie-vlak
F_2	twee-dimensionale Fourier-transformatie
f(x,y)	Grijswaarde-verdeling
FP(u,v)	Fourier-verdeling
X,Y	bemonsterings-afstanden in het hologramvlak
R,S	afmetingen van rechthoekige hologram-pixels
U,V	bemonsterings-afstanden in het reconstructievlak
λ	schaalfactor
I(u,v)	intensiteitsverdeling in reconstructievlak
SF	Schaalfactor-vrijheid
PF	Fase-vrijheid
AF-1	Amplitude-vrijheid buiten het object
AF-2	Amplitude-vrijheid op het object
2*AF	Amplitude-vrijheid zowel op het object als er buiten
E_{quant}	Kwantisatie-ruis
E_{clip}	Kwantisator-ruis
α	Foutverspreidingsfactor voor error-diffusie

Inhoudsopgave

Samenvatting 1

Lijst van symbolen en afkortingen 2

Inhoudsopgave 3

1. Inleiding. 5

2. Het naar diffractie-efficiëntie geoptimaliseerde kwantisatie-algoritme. 7

2.1 Belangrijke parameters voor het Fourier-patroon. 7
2.2 De vrijheden van het Fourier-patroon. 13
2.3 Het kwantisatie-algoritme. 14
2.4 Karakteristieke verloop van de procesparameters. 16

3. Invloed van het initiële hologram op het iteratieproces. 19

3.1 Het verloop van het iteratie-proces bij het met verschillende vrijheden gegenereerde initiële hologram. 19
3.2 De invloed van de plaats van het object op het iteratie-proces. 25
3.3 Invloed van de verhouding van het aantal pixels van het object t.o.v het aantal pixels van het hologram. 26
3.4 Conclusies en aanbevelingen met betrekking tot het initiële hologram. . . . 29

4. Beïnvloeding van het iteratief verhogen van de transparantie-efficiëntie. 31

4.1 De invloed van de verschillende vrijheden bij het terugzetten van het object. 35
4.2 De invloed van de snelheid waarmee de transparantie-efficiëntie verhoogd wordt. 41
4.3 Verschillende begrenzers voor het iteratief verhogen van de transparantie-efficiëntie. 43

4.3.1 De lineaire begrenzer. 43
4.3.2 Niet lineaire begrenzers. 46

4.4 Conclusies en aanbevelingen met betrekking tot het iteratief verhogen van de transparantie-efficiëntie. 47

5. Het kwantisatie-proces 51

5.1 De invloed van de verschillende reconstructievrijheden tijdens het kwantisatieproces. 53
5.2 De invloed van verschillende kwantisatoren tijdens het kwantisatie-proces. 54

5.2.1 De stepwise begrenzer met dalende grenswaarde. 54
5.2.2 De stepwise begrenzer met stijgende grenswaarde. 57
5.2.3 Toepassing van het output-output algoritme voor de kwantisatie. . . 59
5.2.4 Toepassing van error-diffusie voor de kwantisatie. 62

5.3 Conclusies en aanbevelingen met betrekking tot het kwantisatie-proces. . . 68

6. Simulated Annealing 71

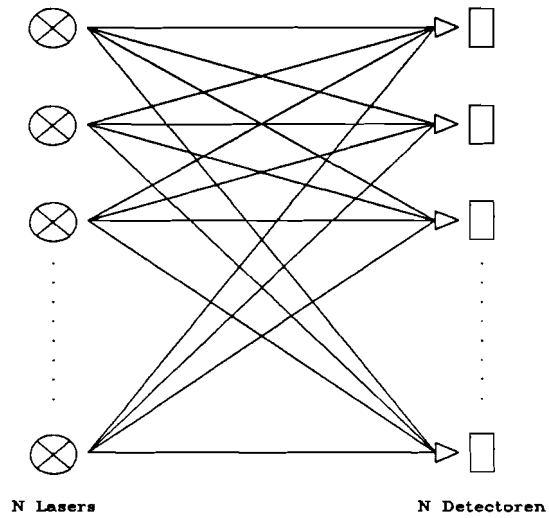
7. Uitwerking van een 2-dimensionale Fast-Fourier-transformatie. 75

8. Conclusies en aanbevelingen 79

Literatuurlijst 80

1. Inleiding.

Het ontwerpen van complexere systemen gaat gepaard met het realiseren van complexere interconnectie-structuren. Het realiseren van de interconnecties met printsporen wordt moeilijker. Met licht kan een complexe interconnectie-structuur echter veel eenvoudiger gerealiseerd worden omdat de interconnecties met licht elkaar ongestoord kunnen kruisen. In de volgende figuur is een voorbeeld van een complexe interconnectie-structuur weergegeven.

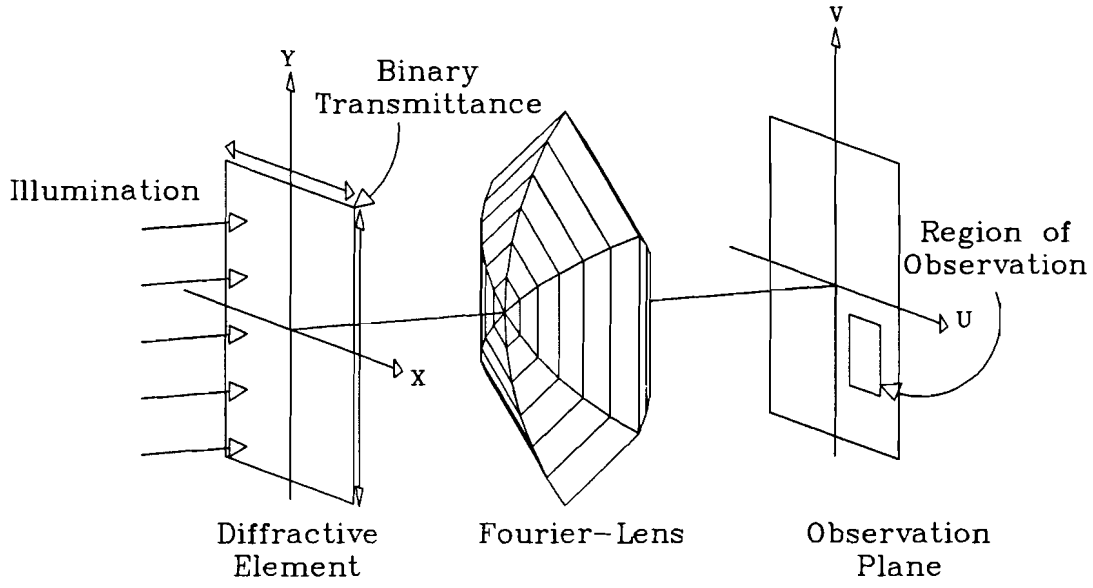


Figuur 1.1 : Een voorbeeld van een complexe interconnectie-structuur.

Achter de verschillende lasers, waarvan de lichtintensiteit ook nog variabel is, staan hologrammen die de verspreiding van het licht over de verschillende detectoren verwezenlijken.

Deze verspreiding van het licht over de detectoren hoeft echter niet uniform te zijn. Het is dus niet alleen mogelijk om complexe interconnectie-patronen te realiseren maar het interconnectie-circuit kan verschillende rekenfuncties uitvoeren. Zo zou het mogelijk zijn om met behulp van deze opstelling een Fourier-transformator te realiseren.

Omdat het in de vakgroep niet mogelijk is om fase-hologrammen te fabriceren is ervoor gekozen om te rekenen aan een opstelling waarbij een Fourier-lens de fase-verdeling over het hologram vervangt. De opstelling waaraan gerekend wordt is in de volgende figuur weergegeven.



Figuur 1.2 : De opstelling waaraan gerekend wordt.

Achter elke laser uit figuur 1.1 zal, in het uiteindelijke ontwerp van een bepaalde rekenfunctie, zo'n opstelling komen te staan. Het 'Observation plane' komt dan overeen met de plaats waar de detectoren opgesteld komen te staan.

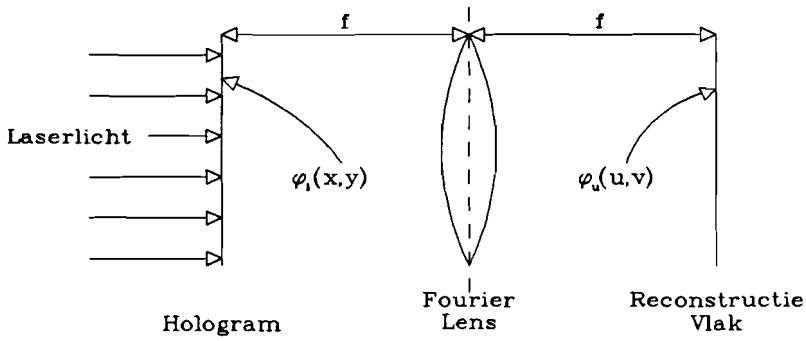
Het is echter mogelijk om de functie van de lens in de opstelling in het hologram te integreren zodat de realisatie van de rekenfunctie een stuk eenvoudiger wordt.

2. Het naar diffractie-efficiëntie geoptimaliseerde kwantisatie-algoritme.

In dit hoofdstuk wordt het uiteindelijke kwantisatie-algoritme nader uitgewerkt. Dit zal beginnen met een bespreking van het gewenste resultaat en enkele variabelen waarmee de kwaliteit van het verkregen resultaat gemeten kan worden.

2.1 Belangrijke parameters voor het Fourier-patroon.

In figuur 2.1 is de opstelling waarin de hologrammen toegepast worden weergegeven. De lens wordt hier gebruikt als Fourier-transformator.



Figuur 2.1 : De lens als Fourier-transformator.

Voor het veld in het reconstructievlak geldt [12]

$$\phi_u(u,v)=C\Phi_i(k_g\frac{x}{f}\ ,\ k_g\frac{y}{f}) \tag{2.1}$$

waarbij C een complexe constante is, k_g het golfgetal en f de brandpuntsafstand.

Omdat het veld in het uitgangsvlak evenredig is met de Fourier-getransformeerde van het veld achter het hologram is het noodzakelijk de grijswaarde-verdeling

$$f_h(x,y)=F_2^{-1}[FP(u,v)] \tag{2.2}$$

als een transparantie te genereren. Hierin is $FP(u,v)$ het gewenste Fourier-patroon en F_2 een tweedimensionale Fourier-transformatie.

Over het algemeen zal $f_h(x,y)$ een complexe verdeling zijn, die in deze vorm niet als transparantie te realiseren is. Om een verdeling als transparantie te realiseren moet $f_h(x,y)$ aan de volgende voorwaarden voldoen

1. $f_h(x,y)$ moet voor alle (x,y) reëel zijn, omdat complexe grijswaarden niet te realiseren zijn. Hieraan zal voldaan worden door het reele deel van $f_h(x,y)$ te realiseren.
2. $f_h(x,y)$ moet voor alle (x,y) in amplitude begrensd en positief zijn, omdat de grijswaarden, waar $f_h(x,y)$ op afgebeeld wordt, begrensd zijn door zwart (geen doorlating) en wit (volledige doorlating).

De eis van begrenzing wordt omgezet in de eis dat de amplitude van het hologram tussen -1 en 1 ligt. Dit is anders dan in de meeste literatuur, waar geëist wordt dat de amplitude van het hologram tussen 0 en 1 ligt, omdat er in de vakgroep veel onderzoek gedaan wordt naar binarisatie van hologrammen met behulp van neurale netwerken. Hierin wordt gewerkt met pixels die de waarde -1 en 1 aan kunnen nemen. Om de voor dit onderzoek ontwikkelde programmatuur ook voor neurale netwerken te bruikbaar te maken is ervoor gekozen om hier ook op pixelwaarden van -1 en 1 uit te komen.

Dit betekent voor de distributie dat deze overgaat in de transparantie

$$f_t(x,y)=2\frac{Re(f_h(x,y))-\min(Re(f_h(x,y))}{\max(Re(f_h(x,y))-\min(Re(f_h(x,y))} - 1 \tag{2.3}$$

oftwel

$$f_t(x,y)=\frac{2Re(f_h(x,y))-\min(Re(f_h(x,y))-\max(Re(f_h(x,y))}{\max(Re(f_h(x,y))-\min(Re(f_h(x,y))}=\alpha Re(f_h(x,y))-\beta \tag{2.4}$$

met

$$\alpha=\frac{2}{\max(Re(f_h(x,y)))-\min(Re(f_h(x,y)))} \tag{2.5}$$

en

$$\beta=\frac{\max(Re(f_h(x,y)))+\min(Re(f_h(x,y)))}{\max(Re(f_h(x,y)))-\min(Re(f_h(x,y)))}. \tag{2.6}$$

Door deze transformatie zal het Fourier-patroon overgaan in

$$FP_t(u,v)=0.5\alpha\, FP(u,v)+0.5\alpha\, FP^*(-u,-v)-\beta\,\delta(u,v). \tag{2.7}$$

De in (u,v) begrensde $FP_t(u,v)$ wordt bemonsterd met bemonsteringsafstanden U en V. De inverse Fourier-transformatie wordt benaderd door de diskrete Fourier-transformatie met lengte N. Het hologram wordt dan

$$f_d(x,y)=\frac{1}{(2\pi)^2}\sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1}\sum_{l=-\frac{N}{2}}^{\frac{N}{2}-1} FP_t(kU,lV)e^{jkUx+jlVy}. \tag{2.8}$$

Door het bemonsteren in het (u,v)-domein is $f_d(x,y)$ periodiek met periode $2\pi/U$ in de x-richting en $2\pi/V$ in de y-richting.

In de praktijk is het niet mogelijk om een continu verlopende en oneindig grote transparantiefunctie te realiseren. Het hologram zal bestaan uit een NxN array van rechthoekige pixels met de afmetingen RxS. De transparantie-distributie wordt dan benaderd door

$$f_p(x,y)=\sum_{m=-\frac{N}{2}}^{\frac{N}{2}-1}\sum_{n=-\frac{N}{2}}^{\frac{N}{2}-1} f_d(mR,nS)\,rect\left(\frac{x-mR}{R},\frac{y-nS}{S}\right) \tag{2.9}$$

met

$$rect(a,b)=[U(a+\tfrac{1}{2})-U(a-\tfrac{1}{2})][U(b+\tfrac{1}{2})-U(b-\tfrac{1}{2})] \tag{2.10}$$

waarin $U(t)$ de stapfunctie is.

Hierdoor gaat het Fourier-patroon over in

$$FP_p(u,v)=RS\,sinc(Ru,Sv)\sum_{m=-\frac{N}{2}}^{\frac{N}{2}-1}\sum_{n=-\frac{N}{2}}^{\frac{N}{2}-1} f_d(mR,nS)e^{-j(mRu+nSv)} \tag{2.11}$$

met

$$sinc(u,v)=\frac{\sin(\frac{u}{2})\sin(\frac{v}{2})}{\frac{uv}{4}}. \tag{2.12}$$

$FP_p(u,v)$ is door het bemonsteren in het hologramvlak periodiek met periode $2\pi/R$ in de u-richting en periode $2\pi/S$ in de v-richting. De grootte van het Fouriervlak is bij bemonstering $NUxNV$ gekozen. Deze afmetingen NU en NV worden nu gelijk genomen aan één periode $2\pi/R$ respektievelijk $2\pi/S$.

Combinatie van (2.8) en (2.11) geeft

$$FP_p(u,v)=RS\ Sinc(Ru,Sv)\sum_{m=-\frac{N}{2}}^{\frac{N}{2}-1}\sum_{n=-\frac{N}{2}}^{\frac{N}{2}-1}\frac{1}{(2\pi)^2}\sum_{k=-\frac{N}{2}}^{\frac{N}{2}-1}\sum_{l=-\frac{N}{2}}^{\frac{N}{2}-1}FP_t(kU,lV)e^{jmR(kU-u)+jnS(lV-v)}. \tag{2.13}$$

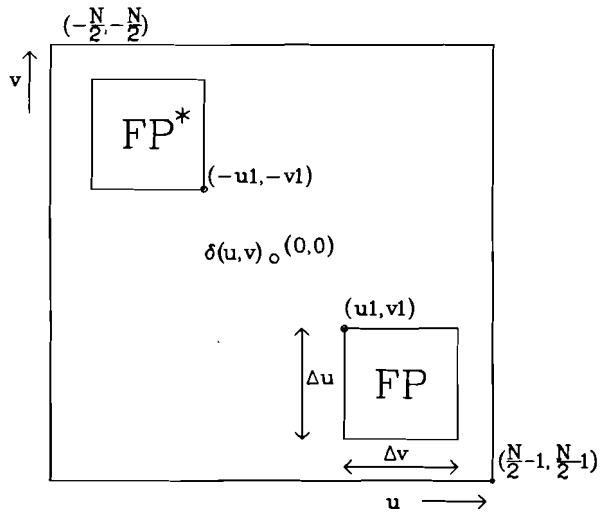
Voor de punten $(u,v)=(fU,gV)$ geldt dan

$$FP_p(fU,gV)=\frac{1}{UV}\ sinc(RfU,SgV)FP_t(fU,gV). \tag{2.14}$$

Volgens (2.7) geldt dan

$$FP_p(fU,gV)=\frac{1}{UV}\ sinc(RfU,SgV)[0.5\alpha FP(fU,gV)+0.5\alpha FP^*(-fU,-gV)-\beta\partial(fU,gV)]. \tag{2.15}$$

Het Fourier-patroon zal dus altijd bestaan uit het gewenste object, een stip in het midden (de gelijkspannings-component) en een puntgespiegelde versie van het object, het zogenaamde 'twin signal'. Dit is in figuur 2.2 weergegeven.



Figuur 2.2 : Het Fourier-patroon van een reëel en begrensd hologram.

Door de eindige afmetingen van de pixels, wordt het beeld vervormd. Deze vervorming is gegeven door de tweede factor uit het rechterlid van (2.15). De eerste nuldoorgang van deze factor ligt voor de u-richting op $u=2\pi/R=NU$. Voor de v-richting ligt de eerste nuldoorgang op $v=2\pi/S=NV$. Als de plaats van het object in de reconstructie dicht bij de oorsprong gekozen wordt, mag deze vervorming verwaarloosd worden.

Omdat voor dit onderzoek alleen de intensiteit van het Fourier-patroon van belang is zal dit patroon op zijn modulus gewaardeerd worden. De volgende waarden geven een indicatie over

de kwaliteit van het verkregen Fourier-patroon.

De transparantie-efficiëntie (η_t) is het deel van de lichtenergie dat door de hologram-distributie doorgelaten wordt. In formulevorm geldt dan

$$\eta_t = \frac{\int_{x=-\frac{N}{2}R}^{(\frac{N}{2}-1)R} \int_{y=-\frac{N}{2}S}^{(\frac{N}{2}-1)S} |f_p(x,y)|^2 dx dy}{N^2 \int_{x=-\frac{R}{2}}^{\frac{R}{2}} \int_{y=-\frac{S}{2}}^{\frac{S}{2}} |rect(\frac{x}{R},\frac{y}{S})|^2 dx dy}. \tag{2.16}$$

Hierin geeft N de hologram-grootte aan. Met (2.9) gaat deze vergelijking over in

$$\eta_t = \frac{\int_{x=-\frac{N}{2}R}^{(\frac{N}{2}-1)R} \int_{y=-\frac{N}{2}S}^{(\frac{N}{2}-1)S} |\sum_{m=-\frac{N}{2}}^{\frac{N}{2}-1} \sum_{n=-\frac{N}{2}}^{\frac{N}{2}-1} f_d(mR,nS) rect(\frac{x-mR}{R},\frac{y-nS}{S})|^2}{N^2 \int_{-\frac{R}{2}}^{\frac{R}{2}} \int_{-\frac{S}{2}}^{\frac{S}{2}} |rect(\frac{x}{R},\frac{y}{S})|^2}. \tag{2.17}$$

Nadere uitwerking hiervan levert

$$\eta_t = \frac{\sum_{m=-\frac{N}{2}}^{\frac{N}{2}-1} \sum_{n=-\frac{N}{2}}^{\frac{N}{2}-1} |f_d(mR,nS)|^2}{N^2}. \tag{2.18}$$

Voor de verdere berekeningen in dit verslag gaan we over op diskrete coördinaten. Hiervoor passen we de volgende coördinatentransformatie toe.

$$(x',y') = (\frac{x}{R},\frac{y}{S}) \quad en \quad (u',v') = (\frac{u}{U},\frac{v}{V}). \tag{2.19}$$

Voor het gemak zullen de accenten weggelaten worden.

De distributie-efficiëntie (η_d) is het deel van de doorgelaten lichtenergie wat op het gewenste plaatje terecht komt. Hiervoor geldt

$$\eta_d = \frac{\sum_{u=u1}^{u1+\Delta u} \sum_{v=v1}^{v1+\Delta v} |FP_t(u,v)|^2}{\sum_{u=-\frac{N}{2}}^{\frac{N}{2}-1} \sum_{v=-\frac{N}{2}}^{\frac{N}{2}-1} |FP_t(u,v)|^2}. \tag{2.20}$$

Hierin zijn $(u1,v1)$ het hoekpunt van het object en $(\Delta u,\Delta v)$ de afmetingen (zie fig. 2.2). Het produkt van deze twee efficiënties is de diffractie-efficiëntie, de verhouding van lichtenergie op het object tot de energie van de op het hologram vallende lichtbundel. Hiervoor geldt volgens Parseval

$$\eta = \eta_t \eta_d = \frac{\sum_{u=u1}^{u1+\Delta u} \sum_{v=v1}^{v1+\Delta v} |FP_t(u,v)|^2}{N^2}. \tag{2.21}$$

Bij het optimaliseren van de voorgaande parameters zullen er handelingen verricht worden die een stoorsignaal over het object genereren. Dit stoorsignaal geeft aanleiding tot het introduceren van een vierde belangrijke procesparameter namelijk de signaal-ruis-verhouding van het object. Dit is de verhouding van het vermogen van het onvervuilde object tot het vermogen van het stoorsignaal. Hiervoor geldt

$$\frac{S}{N} = \frac{\sum_{u=0}^{\Delta u} \sum_{v=0}^{\Delta v} [\lambda FP(u,v)]^2}{\sum_{u=u1}^{u1+\Delta u} \sum_{v=v1}^{v1+\Delta v} [FP_t(u,v) - \lambda FP(u-u1,v-v1)]^2}. \tag{2.22}$$

Hierin is λ de schaalfactor waarmee het originele object vermenigvuldigd wordt om de reconstructie te verkrijgen. Dit is schaalfactor-vrijheid en zal in de volgende paragraaf aan de orde komen.

Over het algemeen zal de applicatie waarin het hologram gebruikt wordt de eisen voor de signaal-ruisverhouding en de diffractie-efficiëntie (η) bepalen.

In de volgende paragraaf zullen de verschillende reconstructie-vrijheden besproken worden.

2.2 De vrijheden van het Fourier-patroon.

In een applicatie waarbij de reconstructie verschillende detectoren aanstuurt zal alleen de intensiteit van het uiteindelijke signaal van belang zijn. Voor deze intensiteit geldt

$$I(u,v) = |FP(u,v)|^2. \tag{2.23}$$

In deze applicatie heeft het diffractie-patroon de volgende vrijheden:

1. Schaalfactor-vrijheid (SF).

   Als niet $FP(u,v)$ gerealiseerd wordt maar $\lambda FP(u,v)$ dan zal dit alleen een contrastverhoging ($\lambda > 1$) tot gevolg hebben. Dit is dus een verbetering van het resultaat.

2. Fase-vrijheid (PF).

   Omdat alleen de modulus van $FP(u,v)$ van invloed is op de uiteindelijke intensiteitsverdeling is de fase van het gerealiseerde diffractie-patroon vrij te kiezen. Ook aan het deel van de reconstructie wat buiten het gebied van de detectoren valt zullen geen eisen m.b.t. de fase gesteld worden. De fase over de gehele reconstructie is dus vrij te kiezen.

3. Amplitude-vrijheid buiten het object (AF-1).

   Omdat we alleen geïnteresseerd zijn in het diffractie-patroon binnen het gebied waar de detectoren opgesteld staan mag de reconstructie buiten dit gebied alle waarden aannemen.

4. Amplitude-vrijheid binnen het object (AF-2).

   Omdat afhankelijk van de applicatie een bepaalde signaal-ruis-verhouding opgelegd wordt mag de amplitude binnen het object afwijken van de gewenste amplitude. Het maximum van deze afwijking wordt bepaald door de gewenste signaal-ruis-verhouding en de eigenschappen van de detectoren.

Het uiteindelijk Fourier-patroon zal er dan, voor alle (u,v) binnen de grenzen van het object, als volgt uitzien

$$FP_g(u,v) = [\lambda |FP(u,v)| + n(u,v)]e^{j\phi(u,v)}. \tag{2.24}$$

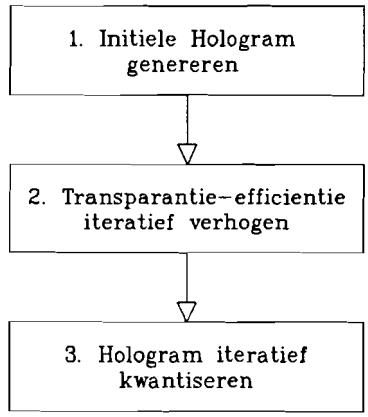
Buiten het object mag het Fourier-patroon een random-signaal zijn. Het vermogen van het

signaal buiten het object moet echter wel zo klein mogelijk zijn om een zo optimaal mogelijke distributie-efficiëntie te verkrijgen.

In de volgende paragraaf zal het algoritme, dat in drie stappen een binair amplitude hologram met een zo groot mogelijke diffractie-efficiëntie en signaal-ruis-verhouding berekent, besproken worden.

2.3 Het kwantisatie-algoritme.

In figuur 2.3 is het kwantisatie-algoritme schematisch weergegeven.

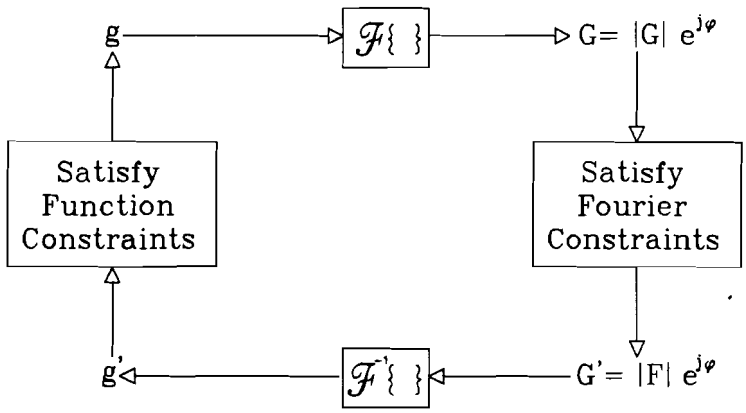


Figuur 2.3 : Het totale kwantisatiealgoritme.

Het algoritme begint met het genereren van het initiële hologram. Dit houdt in dat het originele object, samen met zijn twin-signal, in een lege matrix geplaatst wordt en dat de hierbij behorende pixelverdeling bepaald wordt. Het is echter mogelijk om in deze fase van het algoritme de vrijheden, die in paragraaf 2.2 besproken zijn, te benutten.

Het benutten van deze vrijheden tijdens de generatie van het initiële hologram zal van invloed zijn op het uiteindelijke binaire hologram. Hoe dit van invloed is op het verloop van het totale algoritme zal in hoofdstuk 3 verder onderzocht worden.

In het tweede blok van het algoritme zal de transparantie-efficiëntie iteratief verhoogd worden. Hiervoor wordt een aangepaste versie van het error-reduction algoritme van Gerchberg-Saxton [11] gebruikt. Dit algoritme is in de volgende figuur weergegeven.



Figuur 2.4 : Blokdiagram van het error-reduction (Gerchberg-Saxton) algoritme.

Dit algoritme was in principe uitgevonden in verband met het probleem van het reconstrueren van de fase uit twee intensiteits-metingen in plaats en frekwentiedomein. Het voor hologrammen aangepaste algoritme bestaat uit de volgende vier stappen.

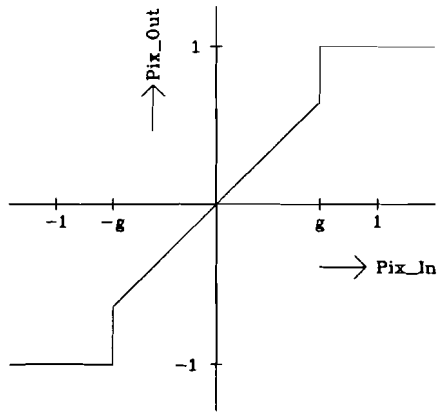
1. Het berekenen van de bij een bepaalde pixelverdeling behorende reconstructie door middel van een twee-dimensionale Fourier-transformatie.
2. Het object en zijn 'twin' in deze reconstructie terugzetten. Hierbij kunnen de verschillende vrijheden benut worden. Het verschil hierin met het error-reduction algoritme is dat alleen voor een deel van de reconstructie er Fourier-constraints zijn, namelijk voor het object en zijn 'twin'.
3. De bij de nieuwe reconstructie behorende pixelverdeling bepalen d.m.v. een inverse Fourier-transformatie.
4. Het in amplitude begrenzen van de berekende pixelverdeling. Dit is namelijk de enige Function-constraint waaraan voldaan hoeft te worden.

Het stopcriterium van dit algoritme is in deze figuur niet weergegeven omdat deze voor de diverse toepassingen verschillend is. Er kan gestopt worden als de reconstructie-fout een gewenste waarde behaald heeft of als het proces om de een of andere reden stagneert.

Het verhogen van de transparantie-efficiëntie per iteratie gebeurt door het object met een grotere amplitude in de reconstructie terug te zetten (stap 2). Dit zal in hoofdstuk 4 verder uitgewerkt worden.

In het derde blok van het hoofdalgoritme zal het hologram iteratief gekwantiseerd worden. Ook hiervoor wordt het aangepaste algoritme uit figuur 2.4 gebruikt. Tijdens het begrenzen van de pixelverdeling (stap 4) zal ook de kwantisatie uitgevoerd worden. Dit kan bijvoorbeeld gedaan worden door een begrenzer die een deel van de pixels op 1 en -1 afbeeldt en een deel van de pixels onveranderd doorlaat.

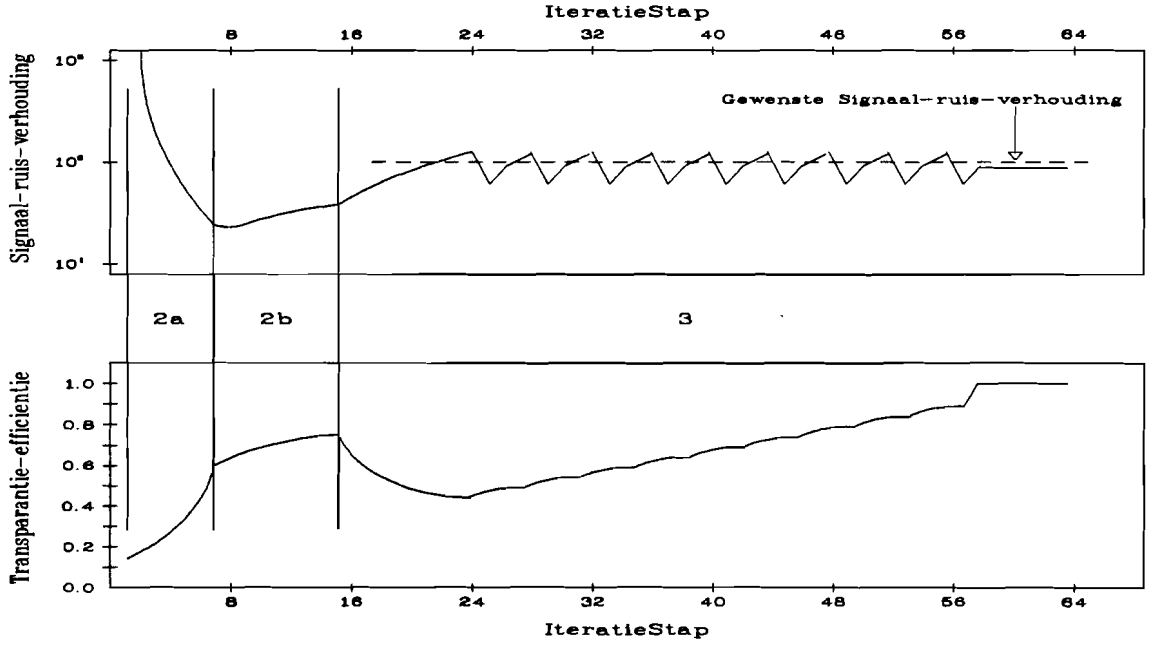
In beginsel zal er uitgegaan worden van een begrenzer die alle pixels met een modulus groter dan een grenswaarde g volgens hun teken op 1 of -1 afbeeldt. Deze begrenzer is in figuur 2.5 weergegeven.
Het kwantisatie-proces begint met een grenswaarde, g, van 1 en zodra de signaal-ruisverhouding de gewenste waarde bereikt of vastloopt wordt deze grenswaarde verlaagd. Uiteindelijk zal de begrenzer alle pixels op 1 of -1 afbeelden en is het hologram een binair amplitude hologram. In hoofdstuk 5 wordt dit kwantisatie-proces nader bekeken en ook andere begrenzers zullen hier aan de orde komen.



Figuur 2.5 : De kwantisator.

2.4 Karakteristieke verloop van de procesparameters.

De berekeningen die verderop in dit verslag volgen leveren zeer karakteristieke figuren op waarin de verschillende stadia van het hoofdalgoritme goed te herkennen zijn. In de volgende figuur is het karakteristieke verloop van de transparantie-efficiëntie en signaal-ruis-verhouding weergegeven.



Figuur 2.6 : Het karakteristieke verloop van de transparantie-efficiëntie en signaal-ruis-verhouding tijdens het totale proces.

De eerste stap van het algoritme kan in deze figuur niet aangegeven worden omdat het initiële

hologram alleen van invloed is op de plaats waar de figuren bij de eerste iteratiestap beginnen.

Het tweede deel van het hoofdalgoritme wordt in twee delen gesplits. Tijdens 2a wordt de transparantie-efficiëntie iedere iteratiestap met een constante factor vermenigvuldigd door het object met een hogere amplitude in de reconstructie terug te zetten. Dit is te zien aan het exponentiële verloop van de transparantie-efficiëntie aan het begin van dit iteratieproces.

Dit terugzetten van het object in de reconstructie wordt echter begrensd zodat de efficiëntie van het object nooit boven de theoretisch maximale waarde van 0.5 kan stijgen. Op een gegeven ogenblik zal de transparantie-efficiëntie dus minder toe gaan nemen. Dit is in de figuur weergegeven als deel 2b.

Tijdens dit iteratief verhogen van de transparantie-efficiëntie is de signaal-ruis-verhouding niet van belang. Deze zal tijdens het verhogen van de transparantie-efficiëntie dan ook dalen. Zodra echter het object met maximale diffractie-efficiëntie in de reconstructie terug wordt gezet wordt de vervorming van het object door de begrenzer minder. De signaal-ruis-verhouding zal dan iets gunstiger worden.

Het derde deel is het uiteindelijke kwantisatieproces. Dit deel komt in de figuren duidelijk naar voren door een trapsgewijs verloop voor zowel de transparantie-efficiëntie als de signaal-ruis-verhouding. Als de grenswaarde van de kwantisator, die in figuur 2.5 weergegeven is, verlaagd wordt zullen er meer pixels op 1 en -1 afgebeeld worden. Hierdoor zal zowel de transparantie-efficiëntie als de kwantisatie-ruis op het object stijgen. De transparantie-efficiëntie zal met een sprong stijgen en de signaal-ruis-verhouding zal met een sprong dalen. Tijdens de volgende iteraties wordt de grenswaarde van de kwantisator constant gehouden totdat de signaal-ruisverhouding weer de gewenste waarde aanneemt of stagneert.

Als de grenswaarde van de kwantisator nul wordt gaat deze over in een hard limiter. Alle pixels worden dan op 1 en -1 afgebeeld. De transparantie-efficiëntie heeft dan zijn maximum van 1.0 bereikt en het hologram is binair.

3. Invloed van het initiële hologram op het iteratieproces.

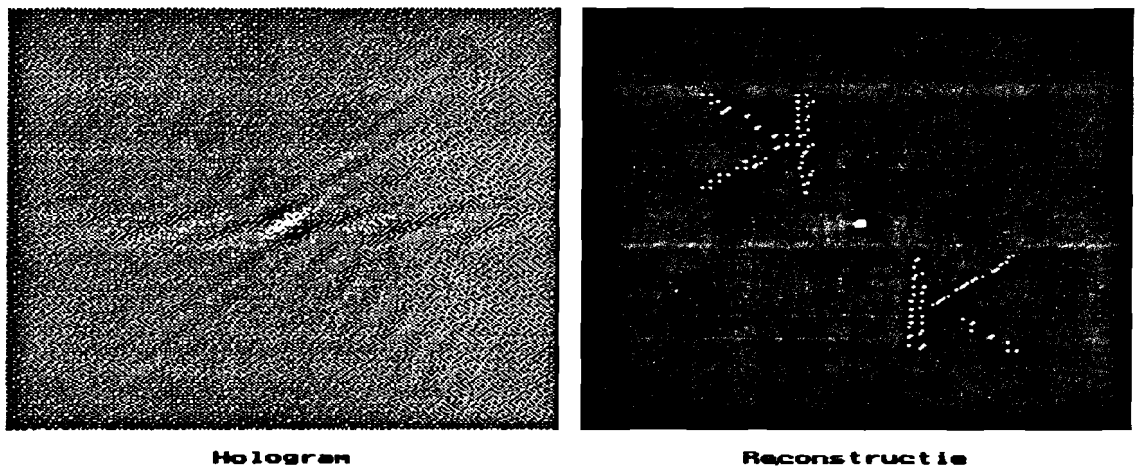
Om te kijken hoe het iteratie-proces reageert op verschillende begin-hologrammen zijn er vier verschillende variaties van dit begin-hologram bekeken. In paragraaf 3.1 zullen de verschillende vrijheden, die ook voor het initiële hologram gelden, bekeken worden. De invloed van de plaats van het object in de reconstructie zal in paragraaf 3.2 besproken worden en in paragraaf 3.3 zal de verhouding van het aantal pixels voor het object tot het totaal aantal pixels aan bod komen.

3.1 Het verloop van het iteratie-proces bij het met verschillende vrijheden gegenereerde initiële hologram.

Voor vier combinaties van vrijheden hebben we het iteratie-proces bekeken. Deze combinaties zijn

Alleen schaalfactor-vrijheid (SF).

Hierbij wordt het beginhologram gegenereerd door het object en zijn 'twin' in een nulmatrix, een matrix bestaande uit nullen, te plaatsen en de bijbehorende pixel-verdeling te berekenen. Deze pixelverdeling wordt dan geschaald zodat deze, wat amplitude betreft, precies in het interval [-1,1] ligt. Deze geschaalde pixelverdeling wordt het initiële hologram. In de volgende figuur is zo'n hologram en de bijbehorende reconstructie weergegeven.



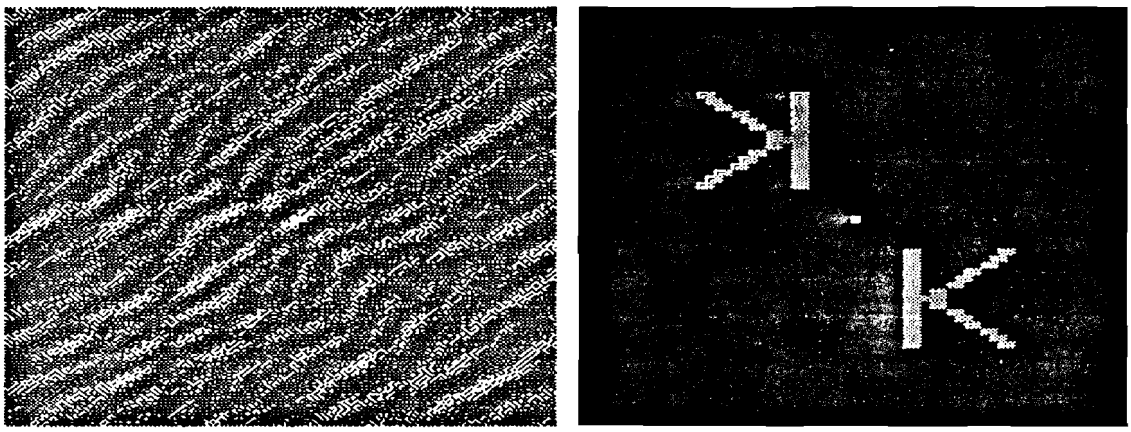
Figuur 3.1 : Initiële hologram met bijbehorende reconstructie gegenereerd met alleen schaalfactor-vrijheid (η=0.0059, η_t=0.051).

Dit hologram en de bijbehorende reconstructie zijn uitgeplot met het error-diffusie-algoritme. De K in de reconstructie lijkt erg slecht maar dit komt omdat er bij verschillende amplitudes verschillende grijswaarden horen. Deze grijswaarden worden door error-diffusie in verschillende punt-verdelingen omgezet. De waarden van de amplitude van het object zijn dus wel constant. Er ligt geen ruis op het object.

Deze K is ook het object dat voor de rest van dit onderzoek als testobject gebruikt wordt.

Schaalfactor- en Fase-vrijheid (SF+PF).

Hier wordt het object met zijn 'twin' ook in een nul-matrix geplaatst maar daarna wordt deze matrix met een random-fase-functie vermenigvuldigd. De bijbehorende pixelverdeling wordt eveneens naar [-1,1] geschaald. Deze geschaalde pixelverdeling wordt het initiële hologram. In de volgende figuur is dit hologram met zijn reconstructie weergeven.



Hologram Reconstructie

Figuur 3.2 : Initiële hologram en de bijbehorende reconstructie gegenereerd met schaalfactor- en fase-vrijheid (η=0.017, η_t=0.034).

Schaalfactor-, Fase- en Amplitude-vrijheid (SF+PF+AF).

Hier wordt eerst een random-matrix gegenereerd. Voor deze matrix geldt dat de amplitudes uniform op het interval [0,1] verdeeld zijn. In deze matrix wordt het met een factor vermenigvuldigde object en zijn 'twin' geplaatst. Deze factor wordt zodanig bepaald dat het lichtvermogen wat op het object valt gelijk is aan het vermogen van de ruis buiten het object en zijn 'twin'. Hiervoor is gekozen omdat aan het begin van het onderzoek bleek dat tijdens het iteratie-proces de maximale signaal-ruis-verhouding bereikt wordt als het lichtvermogen op deze manier over de reconstructie verspreid

wordt. Aan deze vermogengelijkheid wordt voldaan als

$$factor^2 \sum_{u=0}^{\Delta u} \sum_{v=0}^{\Delta v} |FP(u,v)|^2 = \sum_{u=0}^{N} \sum_{v=0}^{N} |RND(u,v)|^2 - 2 \sum_{u=0}^{\Delta u} \sum_{v=0}^{\Delta v} |RND(u,v)|^2. \tag{3.1}$$

Omdat de amplitude buiten het object uniform verdeeld is mag in het rechterlid van (3,1) de integrand vervangen worden door de gemiddelde waarde van deze integrand over het interval [0,1].

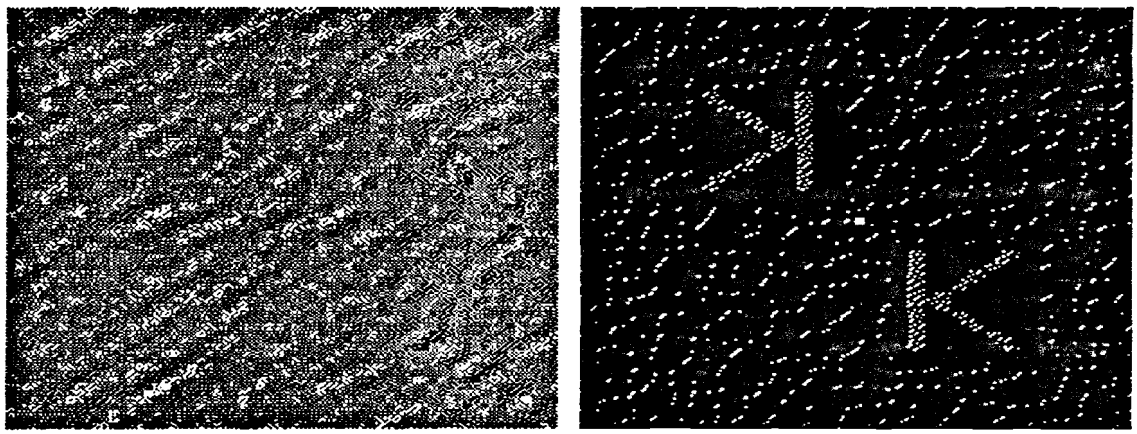
Hiervoor geldt

$$Mean(|RND(u,v)|^2) = \int_{a=0}^{1} f_a(a)|a|^2 da = \int_{a=0}^{1} |a|^2 da = \frac{1}{3}. \tag{3.2}$$

Voor de factor waarmee het object vermenigvuldigd moet worden om deze vermogengelijkheid te garanderen geldt dan

$$factor = \sqrt{\frac{\frac{1}{3}(N^2 - 2\Delta u \Delta v)}{\sum_{u=0}^{\Delta u} \sum_{v=0}^{\Delta v} |FP(u,v)|^2}}. \tag{3.3}$$

In de volgende figuur is zo'n hologram met zijn reconstructie weergegeven.



Hologram Reconstructie

Figuur 3.3 : Hologram met bijbehorende reconstructie gegenereerd met schaalfactor-, fase- en amplitude-vrijheid (η=0.033, η_i=0.097).

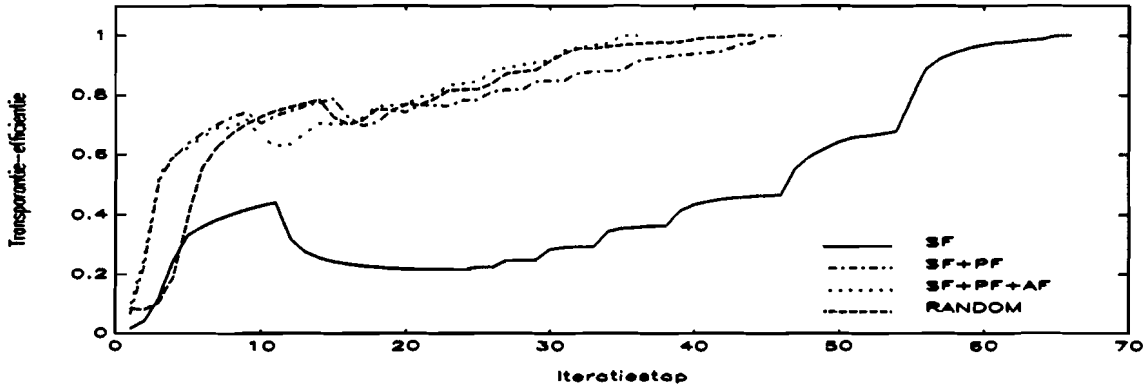
In de drie voorgaande figuren is duidelijk te zien dat het hologram steeds lichter wordt. Dit betekent dat het hologram meer licht doorlaat en een hogere transparantie-efficiëntie heeft. Vooral het hologram waarin alle drie de vrijheden benut zijn geeft goed weer hoe het probleem het best opgelost kan worden. Op het eerste gezicht lijkt

het namelijk slechter om licht buiten het object te laten vallen omdat dit ten koste gaat van het lichtvermogen dat op het object valt. Dit zou een correcte gedachte zijn ware het niet dat het toevoegen van deze componenten buiten het object het vermogen van het licht wat door het hologram doorgelaten wordt groter maakt. Het blijkt zelfs, uit de verschillende efficiëntiewaarden, die bij de figuren weergegeven zijn, dat door het toevoegen van deze componenten het lichtvermogen op het object stijgt.

Volledige vrijheid (RANDOM).

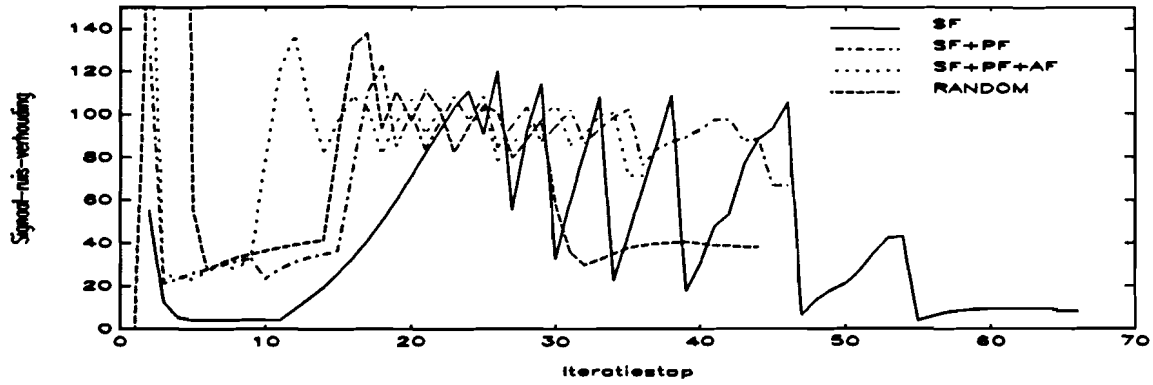
Hier wordt het initiële hologram een random hologram waarvan alle pixels uniform over het interval [-1,1] verdeeld zijn.

In de volgende figuur is het verloop van de transparantie-efficiëntie voor de vier verschillende start-hologrammen weergegeven. Let wel tijdens het iteratie-proces worden, bij het terugzetten van het object, alle vrijheden gebruikt.



Figuur 3.4 : Het verloop van de transparantie-efficiëntie voor initiële hologrammen met verschillende vrijheden.

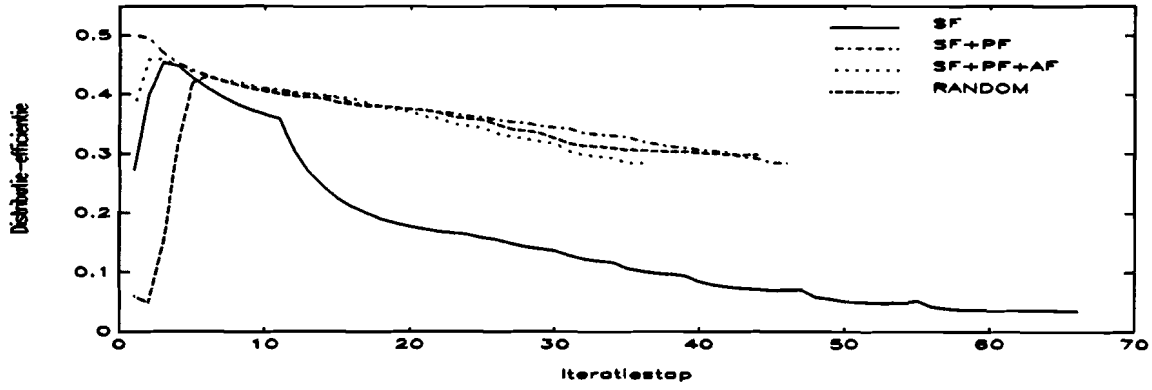
Te zien is dat als in het initiële hologram alleen schaalfactor-vrijheid gebruikt wordt het gehele iteratie-proces veel langer duurt. Dit komt omdat er veel meer iteraties nodig zijn om de signaal-ruis-verhouding weer op de gewenste waarde te krijgen. Dit is in de volgende figuur beter te zien.



Figuur 3.5 : Verloop van de signaal-ruis-verhouding voor de verschillende beginhologrammen.

In deze figuur is te zien dat na het iteratief verhogen van de transparantie-efficiëntie de signaal-ruis-verhouding bij alleen schaalfactor-vrijheid veel langzamer stijgt dan bij de andere drie initiële hologrammen. Ook valt het op dat bij het verlagen van de grenswaarde van de 'stepwise' begrenzer, zoals deze in het vorige hoofdstuk besproken is, de signaal-ruis-verhouding veel meer daalt dan bij de overige drie vrijheden-combinaties.

Blijkbaar kan het algoritme alleen fase, die aanwezig is, zodanig omvormen dat de signaal-ruisverhouding stijgt. Het algoritme is echter niet in staat om deze fase zelf te genereren. Bij alleen schaalfactor-vrijheid is het algoritme dus aangewezen op de amplitudevrijheid om de signaal-ruis-verhouding te verbeteren. Dit is ook duidelijk te zien aan het verloop van de distributie-efficiëntie dat in de volgende figuur weergegeven is.



Figuur 3.6 : Het verloop van de distributie-efficiëntie voor de verschillende beginhologrammen.

Deze distributie-efficiëntie is bij alleen schaalfactor-vrijheid veel lager dan bij de andere drie vrijheden-combinaties. Dit betekent dat er in verhouding veel meer vermogen buiten het object valt. Blijkbaar is er tijdens het kwantisatie-proces geen fase die het algoritme kan shapen om de signaal-ruis-verhouding te laten stijgen. Dit gebrek moet opgevangen worden

door het benutten van de amplitude-vrijheid wat betekent dat er veel meer amplitude buiten het object gegenereerd wordt. Dit leidt tot een veel lagere distributie-efficiëntie.

Opvallend in deze figuur is dat bij SF de distributie-efficiëntie niet bij 0.5 begint. Dit is op het eerste gezicht vreemd omdat er gestart wordt met een lege matrix met daarin het object en zijn 'twin'. De distributie-efficiëntie hiervan is 0.5. De bij deze reconstructie behorende pixelverdeling wordt berekend en in amplitude tussen -1 en 1 geschaald. Dit schalen introduceert blijkbaar een gelijkspannings-component die verantwoordelijk is voor een lagere distributie-efficiëntie. Zodra het object van een random-fase voorzien wordt (SF+PF) genereert het schalen van de bijbehorende pixelverdeling een veel lagere gelijkspannings-component.

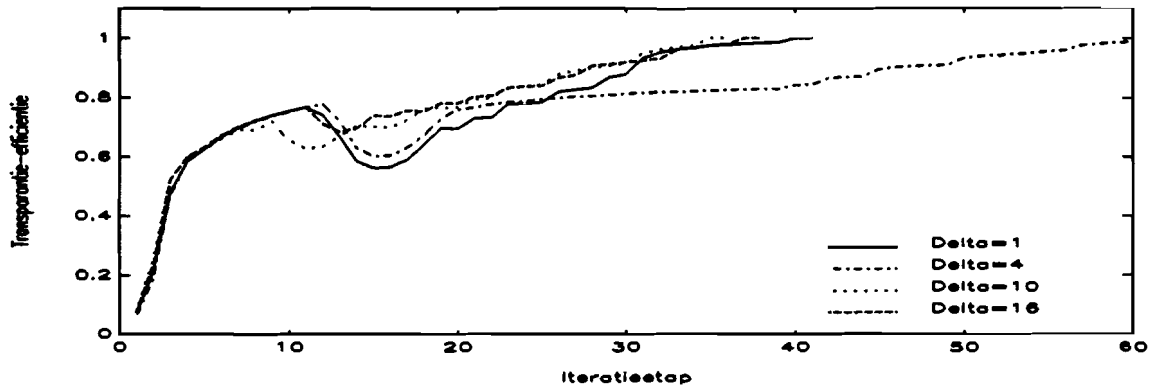
Te zien is dat bij SF+PF, SF+PF+AF en RANDOM de distributie-efficiëntie naar een derde toe gaat. Dit betekent dat er net zoveel vermogen buiten het object, en zijn twin, als op het object valt. Dit is de reden waarom ervoor gekozen is om te bekijken wat er gebeurt bij een begin-hologram waar dit al het geval is (SF+PF+AF).

Ook is in figuur 3.5 te zien dat bij alle vrijheden-combinaties de signaal-ruis-verhouding iedere keer als de grenswaarde verlaagd wordt iets meer inzakt. Bij de laatste verlaging van deze grenswaarde gaat de begrenzer over in een 'hard limiter' die alle pixels van hologram clipt. De signaal-ruis-verhouding blijft twee iteraties hetzelfde en het algoritme stopt. Als er echter met een random hologram begonnen wordt, dan wordt het punt waarop de signaal-ruis-verhouding nagenoeg vastzit al veel eerder bereikt.

Omdat het begin-hologram met de drie vrijheden, SF+PF+AF, na het iteratief verhogen van de transparantie-efficiëntie (stap 2 van het hoofdalgoritme) met een betere signaal-ruis-verhouding uitkomt zijn er minder iteratiestappen voor het gehele proces nodig. Verder is het verschil tussen wel of geen random-amplitude buiten het object erg gering. Het iteratie-proces zal aan de hand van dit initiële hologram verder bekeken worden.

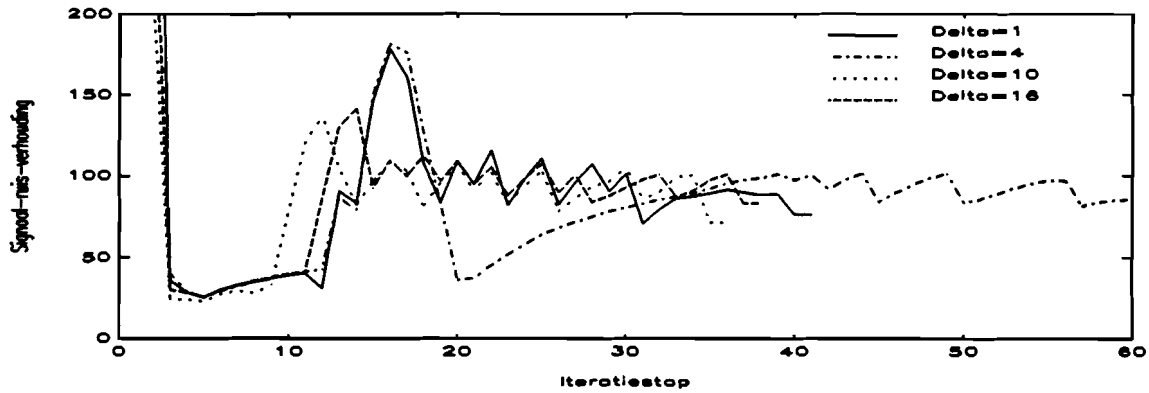
3.2 De invloed van de plaats van het object op het iteratie-proces.

Het iteratie-proces is voor verschillende plaatsen van het object in de totale reconstructie bekeken. Dit onderzoek is echter wel beperkt tot verschillende plaatsen van het object op de hoofd-diagonaal van de reconstructie. Dit betekent dat het algoritme voor verschillende hoekpunten *(ul=delta, vl=delta)*, zie figuur 2.2, bekeken is. In de volgende figuur is het verloop van de transparantie-efficiëntie voor verschillende delta's weergegeven.



Figuur 3.7 : Verloop van de transparantie-efficiëntie voor objecten die op verschillende plaatsen op de hoofd-diagonaal van de reconstructie staan.

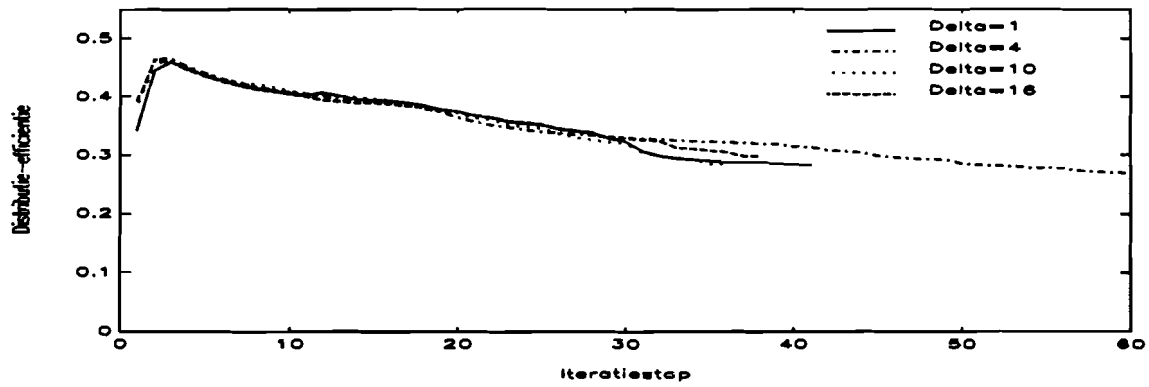
De signaal-ruisverhouding heeft het volgende verloop.



Figuur 3.8 : Het verloop van de signaal-ruis-verhouding voor objecten die op verschillende plaatsen op de hoofd-diagonaal van de reconstructie staan.

Deze twee figuren leveren niets structureels op. De vorm van het object en de plaats waar het in de uiteindelijke reconstructie gezet wordt bepalen samen de verhouding waarmee verschillende frekwentie-componenten voorkomen. Bij dit object en de plaats *delta=4* levert dit blijkbaar een slechte combinatie op waardoor er veel meer iteratiestappen nodig zijn om een pixel-verdeling te genereren die aan de gewenste eisen voldoet.

In de volgende figuur is het verloop van de distributie-efficiëntie weergegeven.



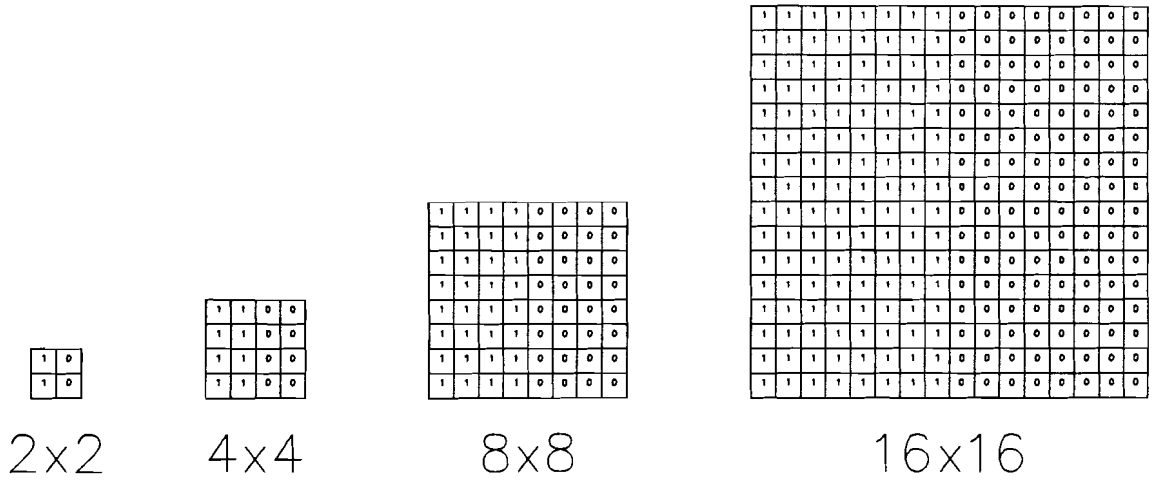
Figuur 3.9 : Verloop van de distributie-efficiëntie voor objecten die op verschillende plaatsen op de hoofd-diagonaal van de reconstructie staan.

Zoals in deze figuur te zien is maakt het voor de distributie-efficiëntie niet erg veel uit waar het object in de totale reconstructie staat.

3.3 Invloed van de verhouding van het aantal pixels van het object t.o.v het aantal pixels van het hologram.

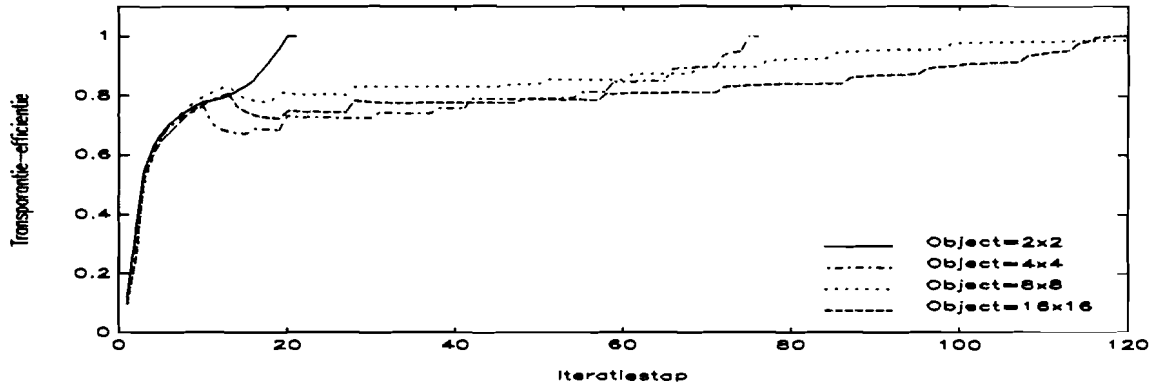
De verhouding tussen het aantal pixels van het object, de zogenaamde 'gebonden' pixels, en de rest van de pixels, de 'vrije' pixels, is van groot belang voor zowel de uiteindelijke diffractie-efficiëntie als de signaal-ruis-verhouding. Het verloop van het algoritme is voor objecten van verschillende afmetingen in een 64x64 pixels grote reconstructie bekeken. Om vooral de invloed van de pixel-verhouding op de te realiseren signaal-ruis-verhouding naar voren te laten komen is ervoor gekozen om deze berekeningen bij een zeer zware signaal-ruis-verhoudingseis van 10000 uit te voeren.

De verschillende objecten zijn in de volgende figuur weergegeven.



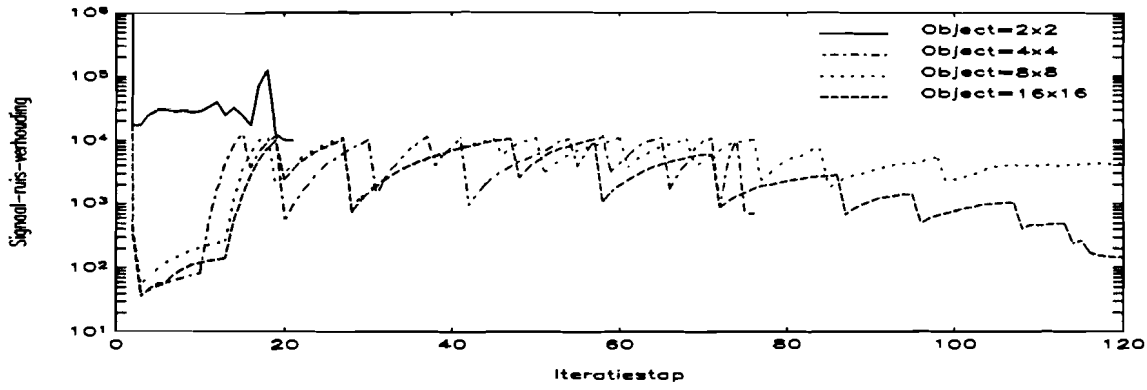
Figuur 3.10 : *Objecten van verschillende groottes om de invloed van de afmetingen van het object op het iteratie-algoritme te testen.*

In de volgende figuur is het verloop van de transparantie-efficiëntie voor deze objecten weergegeven.



Figuur 3.11 : *Verloop van de transparantie-efficiëntie voor objecten met verschillende afmetingen.*

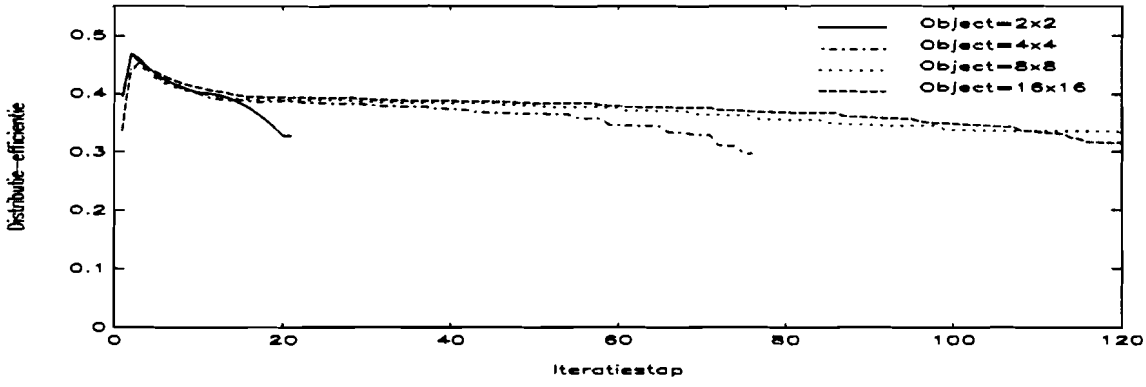
Te zien is dat de transparantie-efficiëntie voor een 2x2 object alleen maar stijgt. Ook als het kwantisatie-proces gestart wordt treedt er geen daling op. Dit komt omdat de signaal-ruis-verhouding bij deze object-grootte zeer hoog blijft. Dit is in de volgende figuur duidelijk te zien.



Figuur 3.12 : Verloop van de signaal-ruis-verhouding voor objecten met verschillende afmetingen.

Met het 16x16 object kan de reconstructie al vrij snel tijdens het kwantisatie-proces niet meer voldoen aan de eis van een signaal-ruis-verhouding van 10000. Deze eis is bewust zo zwaar gemaakt om te laten zien dat het aantal vrije pixels verantwoordelijk is voor de maximaal haalbare signaal-ruis-verhouding. Zodra een applicatie om een zeer grote signaal-ruis-verhouding gaat vragen dan zal de grootte van het hologram hierdoor bepaald worden.

In de volgende figuur is het verloop van de distributie-efficiëntie weergegeven.



Figuur 3.13 : Verloop van de distributie-efficiëntie voor objecten met verschillende afmetingen.

Bij deze figuur moet er wel opgemerkt worden dat als de efficiëntie van het object zelf vier keer zo klein wordt en de distributie-efficiëntie hetzelfde blijft dit betekent dat de efficiëntie-winst vier keer zo groot wordt. Dit betekent dat de intensiteit van het 2x2 object vier keer zo groot is dan de intensiteit van het 4x4 object.

Te zien is dat de verhouding tussen object-grootte en hologram-grootte voor de uiteindelijke distributie-efficiëntie niet veel uitmaakt. Het verloop van de distributie-efficiëntie lijkt alleen anders omdat de lengte van het proces anders is.

Er is ook gekeken naar het verloop van het algoritme als het object hetzelfde bleef maar de grootte van het hologram gewijzigd werd. Hier kwamen dezelfde eigenschappen naar voren als bij de zojuist besproken berekeningen.

Er mag dan ook geconcludeerd worden dat het aantal pixels waaruit het hologram moet bestaan bepaald wordt door de grootte van het object en de gewenste signaal-ruis-verhouding.

3.4 Conclusies en aanbevelingen met betrekking tot het initiële hologram.

De keuze van het initiële hologram is van groot belang op zowel het verloop van het algoritme als op de kwaliteit van de uiteindelijke reconstructie. Voor het verkrijgen van een goede reconstructie moet het initiële hologram aan de volgende eisen voldoen

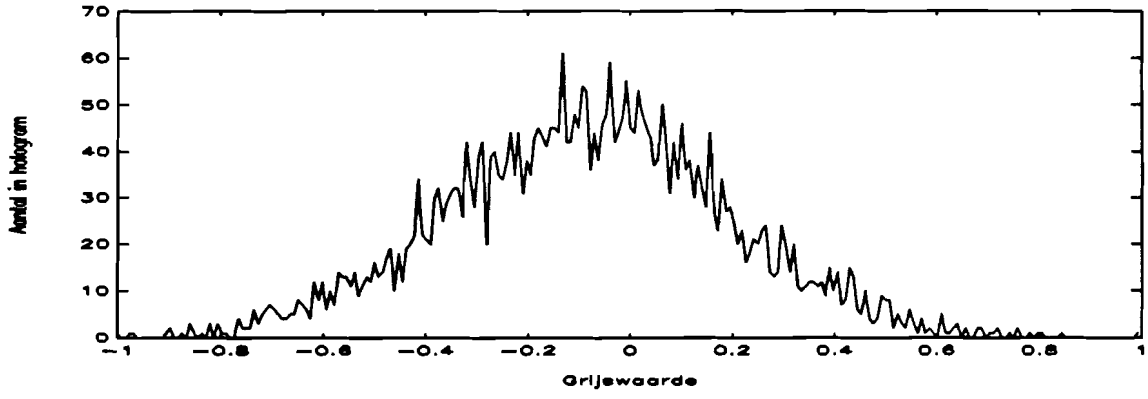
- De fase van de reconstructie die bij het initiële hologram hoort moet een random signaal zijn.

- De amplitude van de reconstructie moet bestaan uit het object en zijn 'twin' en additieve witte ruis buiten deze twee componenten. Het vermogen van deze drie componenten moet identiek zijn, maar zodanig dat het hologram in amplitude begrensd is.

- De grootte van het hologram wordt alleen bepaald door de grootte van het object en de gewenste signaal-ruis-verhouding. Als er uit het verloop van het algoritme blijkt dat de gewenste signaal-ruis-verhouding eenvoudig gehaald wordt dan kan er tijd gewonnen worden door een kleiner hologram te berekenen en enkele van deze kleinere hologrammen naast elkaar te leggen [7].

- De plaats van het object in de reconstructie is niet van te voren te bepalen. Duurt het algoritme gevoelsmatig veel te lang dan kan het een oplossing zijn om hier een wijziging in aan te brengen. Een optimum hierin zal echter door trial en error gevonden moeten worden.

In de volgende figuur is de grijswaarde-verdeling, van een hologram dat met de bovenstaande aanbevelingen berekend is, weergegeven.



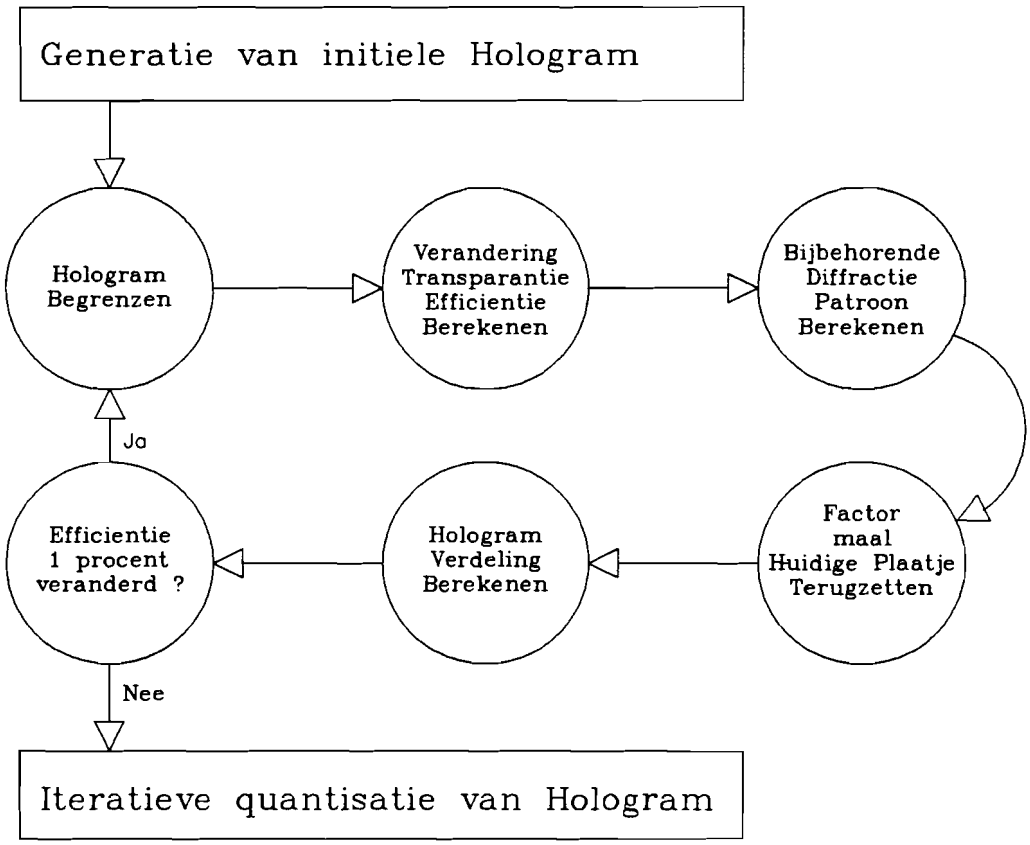
Figuur 3.14 : Grijswaarde-verdeling van een optimaal berekend initiëel hologram.

Te zien is dat de grijswaardes nagenoeg symmetrisch rond 0 verdeeld liggen. Hierdoor is de gelijkspanningscomponent van het hologram zeer klein wat gunstig is voor de distributie-efficiëntie.

In het volgende hoofdstuk zal er gekeken worden naar het iteratief verhogen van de transparantie-efficiëntie van een hologram wat met deze reconstructie-vrijheden (SF+PF+AF-1) berekend is.

4. Beïnvloeding van het iteratief verhogen van de transparantie-efficientie.

Na het genereren van het initiële hologram wordt de tweede stap van het algoritme ingegaan. Deze stap houdt het iteratief verhogen van de transparantie-efficiëntie in. Het doel hiervan is alleen het verhogen van het lichtvermogen wat door het hologram komt. Wat dit voor gevolgen voor de signaal-ruis-verhouding heeft is niet van belang, hoe hoger de transparantie-efficiëntie hoe beter. In de volgende figuur is het algoritme nog eens grafisch weergegeven met deze stap nader uitgewerkt.



Figuur 4.1 : Grafische weergave van het hoofdalgoritme met het iteratief verhogen van de transparantie-efficiëntie (stap 2) verder uitgewerkt.

Het iteratief verhogen van de transparantie-efficiëntie gebeurt in de volgende vijf stappen.

1. Het in amplitude begrenzen van de hologramverdeling met een begrenzer. De vorm van deze begrenzer is op dit moment nog niet van belang. Het is in ieder geval een lokale operator.

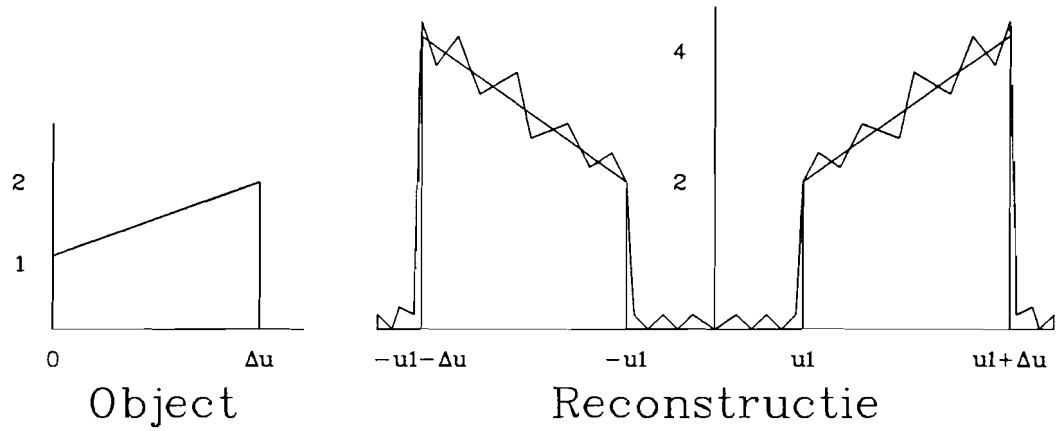
2. Door dit begrenzen zal de transparantie-efficiëntie veranderen. Voor de winst in transparantie-efficiëntie geldt

$$\eta_{t\text{-}winst} = \frac{\eta_t \ \ na \ \ begrenzing}{\eta_t \ \ voor \ \ begrenzing}. \tag{4.1}$$

3. Het berekenen van de reconstructie die bij dit nieuwe hologram hoort. Voor deze reconstructie geldt

$$FP_g(u,v) = [\lambda\,|FP(u,v)| + n(u,v)]e^{j\phi(u,v)}. \tag{4.2}$$

λ is hierin de schaalfactor waarmee het object vergroot is, $n(u,v)$ is de ruis die door de voorgaande bewerkingen over het object terecht gekomen is.
Dit is in de volgende afbeelding voor het één-dimensionale geval grafisch weergegeven.



Figuur 4.2 : Grafische weergave van het begrip schaalfactor voor een één-dimensionaal geval.

Duidelijk is te zien dat de reconstructie opgebouwd is volgens (4.2). In dit geval is de schaalfactor λ gelijk aan 2.

De schaalfactor λ wordt zodanig berekend dat het ruisvermogen minimaal is. Voor dit ruisvermogen geldt

$$N = \sum_{u=ul}^{ul+\Delta u} \sum_{v=vl}^{vl+\Delta v} |n(u,v)|^2 = \sum_{u=ul}^{ul+\Delta u} \sum_{v=vl}^{vl+\Delta v} \big|\,|FP_g(u,v)| - \lambda\,|FP(u,v)|\,\big|^2. \tag{4.3}$$

Dit ruisvermogen is minimaal als

$$\frac{\partial N}{\partial \lambda}=2\sum_{u=u1}^{u1+\Delta u}\sum_{v=v1}^{v1+\Delta v}[\,|FP_g(u,v)|-\lambda\,|FP(u,v)|\,]\,|FP(u,v)|=0. \tag{4.4}$$

Hieraan is voldaan als voor λ geldt

$$\lambda=\frac{\sum_{u=u1}^{u1+\Delta u}\sum_{v=v1}^{v1+\Delta v}|FP_g(u,v)|\,|FP(u,v)|}{\sum_{u=u1}^{u1+\Delta u}\sum_{v=v1}^{v1+\Delta v}|FP(u,v)|^2}. \tag{4.5}$$

4. Om te garanderen dat de transparantie-efficiëntie iedere iteratie met een bepaalde waarde toeneemt zal het huidige object en zijn 'twin', afhankelijk van de transparantie-winst die in stap 2 behaald is, groter of kleiner teruggezet worden. Het huidige object betekent dat niet *Factor* maal het object maar *Factor* maal λ maal het object zal in de berekende reconstructie teruggezet wordt. Voor deze factor geldt

$$Factor=\sqrt{\frac{\Delta\eta_t}{\eta_{t-winst}}}. \tag{4.6}$$

Hierin is $\Delta\eta_t$ de winst in transparantie-efficiëntie die iedere iteratie nagestreefd wordt. Zolang in de reconstructie de amplitude buiten het object en het 'twin-signal' erg klein is zal de transparantie-efficiëntie ook daadwerkelijk iedere iteratie met $\Delta\eta_t$ toenemen. Zodra de amplitude van de frekwentie-componenten buiten het object en zijn 'twin' echter toenemen zal de vermogensverandering, die tijdens het begrenzen van het hologram optreedt, ook voor een deel door verandering van deze componenten teweeg gebracht worden. De vermogensverandering van deze componenten wordt echter niet gecompenseerd door het terugzetten van *Factor* maal het huidige object en de transparantie-efficiëntie zal niet meer met $\Delta\eta_t$ toenemen.

Op den duur zou het hologram een maximale transparantie-efficiëntie van 1.0 hebben en volledig binair zijn. De praktijk heeft echter uitgewezen dat de factor waarmee het object teruggezet wordt zeer groot wordt. Bij het terugzetten van het object met zo'n grote factor treden er bij het begrenzen verminderingen van de transparantie-efficiëntie in de orde grootte van 1000 op. Dit heeft als gevolg dat het object met een factor 100 of groter in de reconstructie teruggezet wordt. Dit levert de volgende iteratie-slag een nog groter verlies in de transparantie-efficiëntie op. Omdat het object met zo'n enorm grote factor teruggezet wordt is het wel of niet waarderen van de amplitude-vrijheid

buiten het object van ondergeschikt belang. De verhouding tussen de amplitude buiten het object en de amplitude van het object zal dan altijd zeer klein zijn.

Om dit te voorkomen wordt de factor waarmee het object teruggezet wordt begrensd. Dit gebeurt zodanig dat altijd geldt dat de efficiëntie van het object na het terugzetten maximaal gelijk is aan het theoretische maximum van 0.5.

5. De bij de nieuwe reconstructie behorende pixelverdeling berekenen.

Het zesde bolletje in de figuur is een beslissingsbol. Hier wordt gekeken of het proces wat betreft de diffractie-efficiëntie ($\eta = \eta_t \eta_d$) stagneert. Het iteratieproces stopt als de diffractie-efficiëntie door één volledige iteratieslag minder dan één procent verandert.

Bij dit transparantie-efficiëntie-verhogende proces zijn de volgende factoren variabel en van invloed op het verloop van dit proces.

1. De vrijheden die bij het terugzetten van het object in acht genomen worden. De invloed hiervan zal in paragraaf 4.1 bekeken worden.

2. De snelheid waarmee de transparantie-efficiëntie per iteratie groeit zal van invloed zijn op het gehele proces. Dit wordt in paragraaf 4.2 verder bekeken.

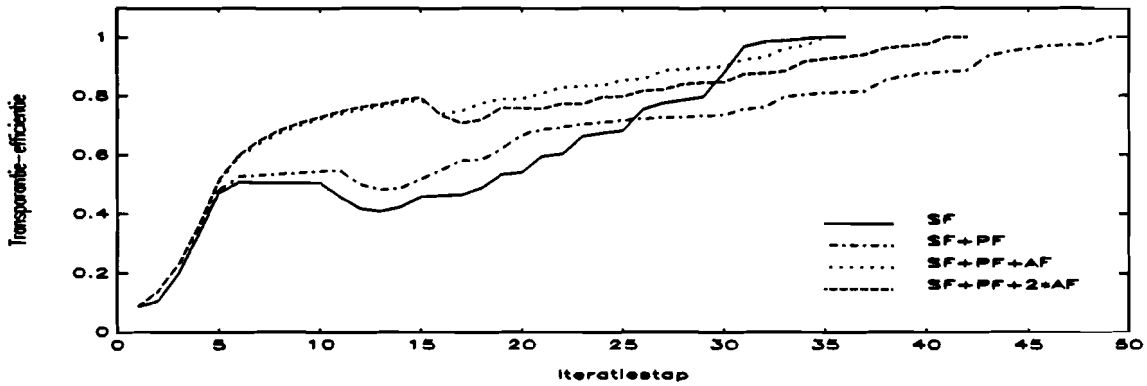
3. De manier waarop het hologram begrensd wordt is ook van invloed op het proces. Dit zal in paragraaf 4.3 bekeken worden.

Bij het bekijken van de invloed van deze drie factoren op het totale iteratie-proces zal dit ingepast worden in het gehele algoritme. Voor het initiële hologram is gekozen voor het hologram zoals dat in het vorige hoofdstuk bepaald is, namelijk het initiële hologram dat met schaalfactor-, fase- en amplitude-vrijheid buiten het object berekend is.

4.1 De invloed van de verschillende vrijheden bij het terugzetten van het object.

Het totale proces is bekeken voor het met verschillende vrijheden terugzetten van het object in de reconstructie tijdens het iteratief verhogen van de transparantie-efficiëntie (stap 2 van het hoofdalgoritme). Deze vrijheden zijn in hoofdstuk 2 besproken.

In de volgende figuur is het verloop van de transparantie-efficiëntie voor deze verschillende vrijheden uitgezet.



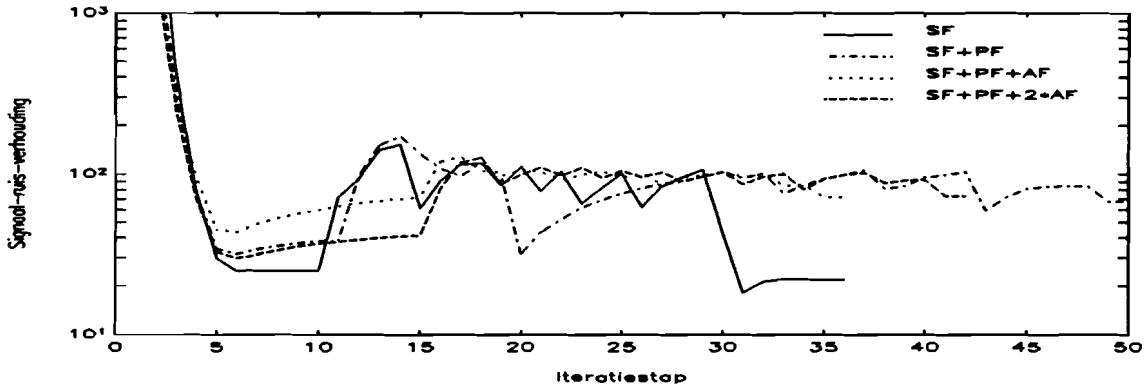
Figuur 4.3 : Verloop van de transparantie-efficiëntie voor het terugzetten van het originele object met verschillende vrijheden.

De afkortingen betekenen het volgende.

SF :	Schaalfactor-vrijheid.
SF+PF :	Schaalfactor- en fase-vrijheid.
SF+PF+AF :	Schaalfactor-, fase- en amplitude-vrijheid buiten het object.
SF+PF+2*AF:	Schaalfactor-, fase- en amplitude-vrijheid binnen en buiten het object.

Te zien is dat het maximum van de transparantie-efficiëntie veel hoger ligt als er amplitude-vrijheid bij komt. Dit komt omdat het iteratie-proces door deze extra vrijheid (of vrijheden) niet zo snel vast zit. Het verschil tussen alleen amplitude-vrijheid buiten het object of ook op het object is niet erg groot.

Opvallend is dat het voor de transparantie-efficiëntie niet erg veel uitmaakt of er wel of geen fase-vrijheid op de reconstructie gewaarborgd wordt. Het verloop lijkt alleen anders omdat de grafiek erg uitgerekt is. Dit komt omdat het proces voor (SF+PF) veel trager varieert omdat de signaal-ruis-verhouding erg langzaam verandert. Het verloop van deze signaal-ruis-verhouding is in de volgende figuur weergegeven.

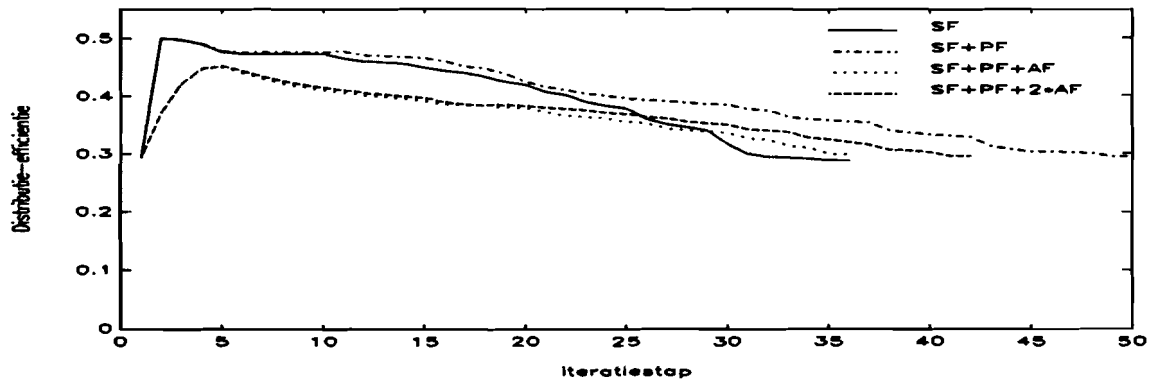


Figuur 4.4 : Het verloop van de signaal-ruis-verhouding bij het terugzetten van het originele object met verschillende vrijheden.

Hieruit blijkt dat het voor de signaal-ruis-verhouding wel van belang is dat de fase-vrijheid op de reconstructie gewaarborgd wordt. Alleen schaalfactor-vrijheid (SF) levert op het einde van het iteratief verhogen van de transparantie-efficiëntie (Iteratiestap=10) de laagste signaal-ruis-verhouding op. Zodra het algoritme overgaat op het terugzetten van het object met een maximale efficiëntie stijgt de signaal-ruis-verhouding bij alle vrijhedencombinaties behalve bij alleen schaalfactor-vrijheid. Blijkbaar is fase-vrijheid noodzakelijk om de signaal-ruis-verhouding te doen stijgen. Zodra het kwantisatie-proces start, waarbij alle vrijheden gewaarborgd worden, stijgt deze signaal-ruis-verhouding dus wel.

Duidelijk is ook te zien dat de signaal-ruis-verhouding zonder amplitude-vrijheid op het object, op het moment dat het iteratief verhogen van de transparantie-efficiëntie vastloopt (iteratiestap=15), veel groter is dan met amplitude-vrijheid op het object.

In de volgende afbeelding is het verloop van de distributie-efficiëntie weergegeven.



Figuur 4.5 : Verloop van de distributie-efficiëntie voor het met verschillende vrijheden terugzetten van het originele object.

Hierin valt het volgende op

- Amplitude-vrijheid buiten het object veroorzaakt een slechtere distributie-efficiëntie tijdens de iteratieve verhoging van de transparantie-efficiëntie. Dit is logisch omdat hier de amplitude buiten het object gehandhaaft blijft tijdens het terugzetten van het object.
- Het algehele proces duurt bij SF+PF en SF+PF+2*AF veel langer dan bij de andere twee vrijheden-combinaties.
- De distributie-efficiëntie komt altijd op ongeveer 0.3 terecht. Blijkbaar is dit noodzakelijk voor het behalen van een bepaalde signaal-ruis-verhouding.

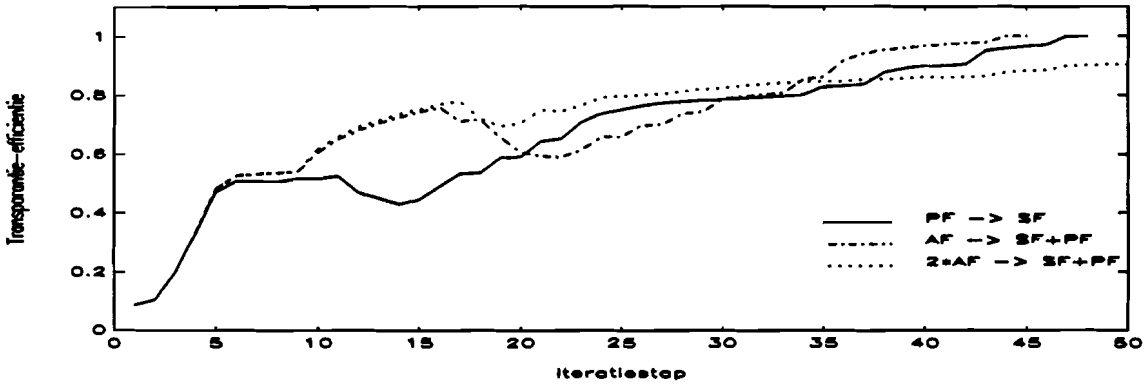
Ook is er geëxperimenteerd met het achtereenvolgens toe laten nemen van het aantal vrijheden die gewaarborgd worden. Zo zijn in de onderstaande figuren de resultaten van de volgende vrijheden-combinaties weergegeven.

PF -> SF : Eerst verhoging met alleen schaalfactor-vrijheid. Als het proces vastloopt wordt fase-vrijheid toegevoegd.

AF1 -> SF+PF : Eerst iteratie met SF+PF, na stagnatie wordt de amplitude-vrijheid buiten het object toegevoegd.

2*AF -> SF+PF : Idem als vorige maar nu met toevoeging van amplitude-vrijheid zowel buiten als op het object.

In de volgende figuur is het verloop van de transparantie-efficiëntie voor deze combinaties weergegeven.

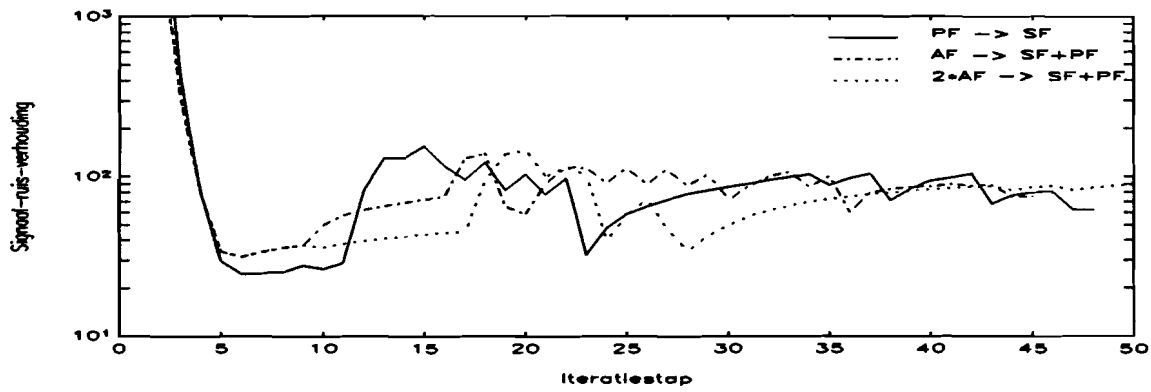


Figuur 4.6 : Verloop van de transparantie-efficiëntie bij de verschillende vrijheden-volgordes.

Hierin valt het volgende op

- Het toevoegen van fase-vrijheid nadat het proces met alleen schaalfactor-vrijheid is vastgelopen (Iteratiestap=11) heeft geen duidelijk zichtbare veranderingen in het verloop van de transparantie-efficiëntie tot gevolg. Blijkbaar is het proces zodanig vastgelopen dat het vrijgeven van de fase op dat moment geen gevolg meer heeft.
- Het toevoegen van de amplitude-vrijheid nadat het proces met schaalfactor- en fase-vrijheid is vastgelopen (Iteratiestap=9) levert wel een duidelijk zichtbare verandering in het verloop van de transparantie-efficiëntie op. Of hierbij nu ook amplitude-vrijheid op het object toegevoegd wordt of niet levert niet zoveel verschil op.

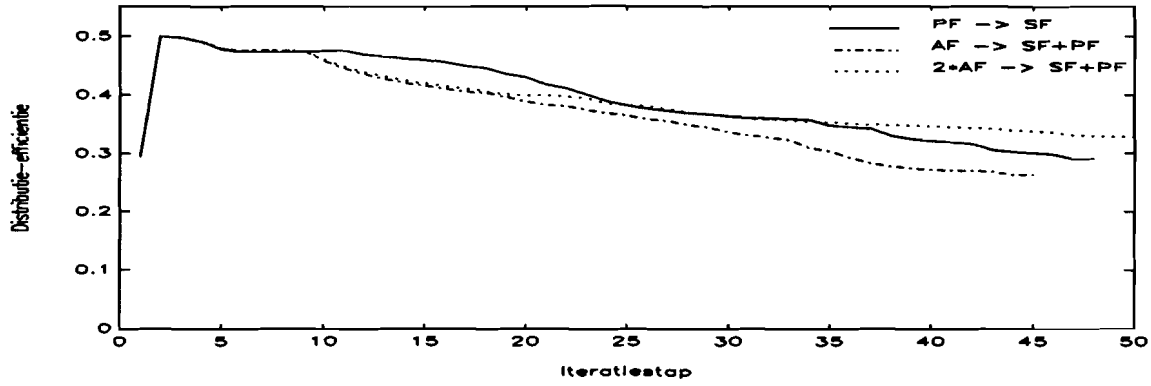
In de volgende figuur is het verloop van de signaal-ruisverhouding voor deze combinaties weergegeven.



Figuur 4.7 : Verloop van de signaal-ruis-verhouding bij verschillende vrijheden-volgordes.

Hierin is te zien dat het toelaten van amplitude-vrijheid op het object, nadat het proces met SF+PF stagneert, voor de signaal-ruis-verhouding wel een verbetering oplevert. Toevoeging van amplitude-vrijheid buiten het object levert alleen een trager verloop van de signaal-ruis-verhouding op.

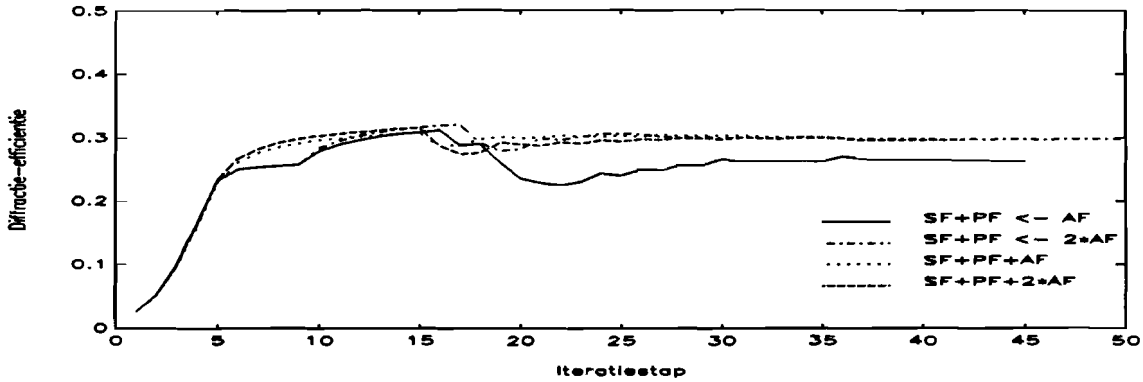
In de volgende figuur is voor deze combinaties het verloop van de distributie-efficiëntie weergegeven.



Figuur 4.8 : Verloop van de distributie-efficiëntie bij de verschillende vrijheden-volgordes.

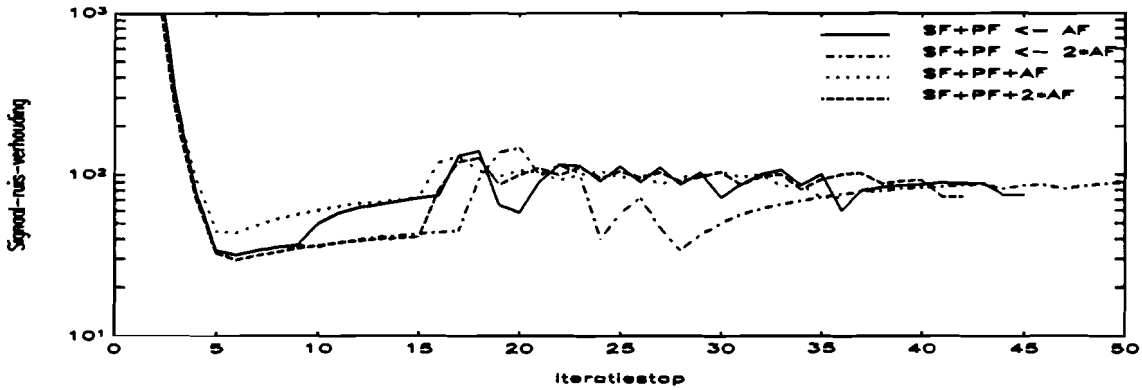
Hierin is duidelijk te zien dat het toevoegen van amplitude-vrijheid buiten het object ten koste van de distributie-efficiëntie gaat. Zodra het proces met schaalfactor- en fase-vrijheid stagneert (Iteratiestap=9) en er amplitude-vrijheid toegevoegd wordt, daalt de distributie-efficientie in de volgende iteratie-slagen veel sneller.

Om het achtereenvolgens toe laten nemen van het aantal vrijheden te vergelijken met het verloop als de vrijheden van begin af aan gewaarborgd worden, is voor de belangrijkste vrijhedencombinaties het verloop van de diffractie-efficiëntie ($\eta = \eta_i \eta_d$) weergegeven.



Figuur 4.9 : Het verloop van de diffractie-efficiëntie voor de belangrijkste vrijheden-combinaties en vrijheden-volgordes.

In de volgende figuur is voor deze combinaties de signaal-ruis-verhouding nog eens weergegeven.



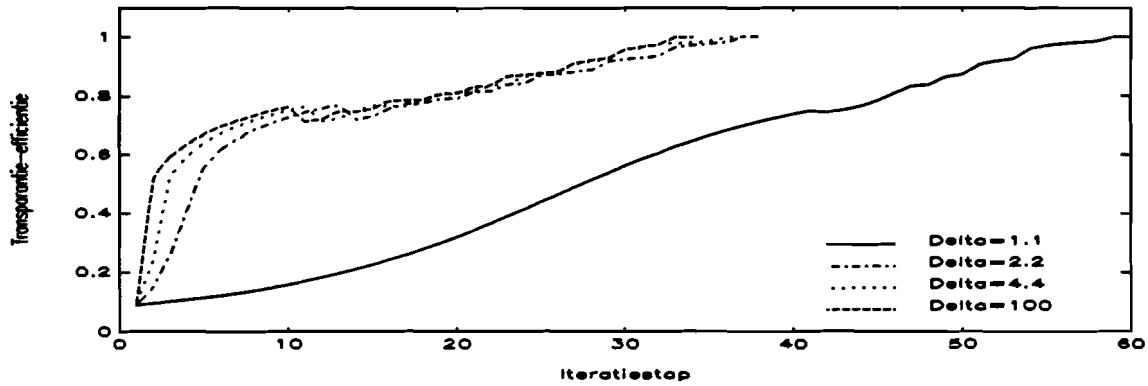
Figuur 4.10 : Verloop van de signaal-ruis-verhouding voor de verschillende combinaties en opeenvolgingen van vrijheden.

Uit deze twee voorgaande figuren blijkt dat het het beste is om tijdens het verhogen van de transparantie-efficiëntie schaalfactor-, fase- en amplitude-vrijheid buiten het object van begin af aan te waarborgen.

Uit figuur 4.10 blijkt ook dat het toelaten van amplitude-vrijheid op het object een slechter uitgangspunt voor het kwantisatie-proces geeft.

4.2 De invloed van de snelheid waarmee de transparantie-efficiëntie verhoogd wordt.

Als de factor waarmee de transparantie-efficiëntie per iteratie vermenigvuldigd wordt, $\eta_{t\text{-}winst}$ uit formule (4.1), varieert dan zal het verloop van deze transparantie-efficiëntie voor verschillende factoren er als volgt uit zien.

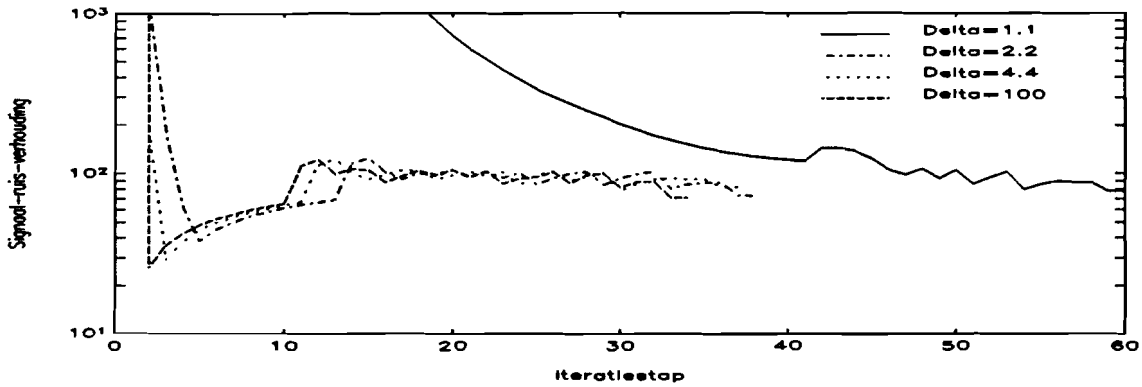


Figuur 4.11 : Het verloop van de transparantie-efficiëntie bij verschillende verhogings-factoren.

De eerste factor is zodanig gekozen dat er gekeken wordt wat er gebeurt als de transparantie-efficiëntie iedere iteratie met een kleine factor (namelijk 1.1) verhoogd wordt. De volgende berekeningen worden uitgevoerd zodat de transparantie-efficiëntie twee keer zo snel groeit als bij de vorige berekening. Een iteratieve verhoging met een delta van 100 is bekeken om te kijken wat er gebeurt als het object iedere iteratie met de theoretisch maximale efficiëntie in de reconstructie terug gezet wordt.

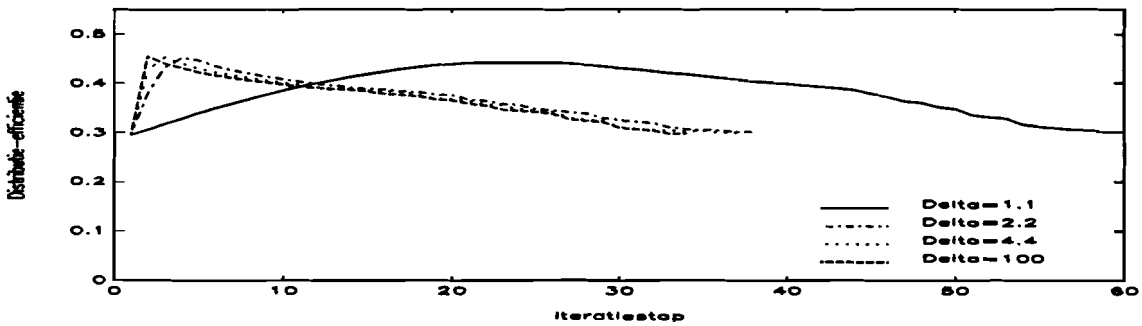
Duidelijk is het verschil in snelheid waarmee de transparantie-efficiëntie iedere iteratie groeit. Het verschil in maximale transparantie-efficiëntie aan het einde van het verhogingsproces is echter erg klein. Wel is te zien dat het langzaam verhogen van de transparantie-efficiëntie veel meer iteratie-stappen (40) nodig heeft.

Het verschil in signaal-ruisverhouding op het einde van het iteratief verhogen van de transparantie-efficiëntie is wel verschillend. Het verloop hiervan is in de volgende figuur weergegeven.



Figuur 4.12 : Verloop van de signaal-ruis-verhouding voor de verschillende verhogings-factoren.

Het verloop voor een delta van 2.2, 4.4 en 100 is niet erg verschillend. Alleen blijkt dat voor een grotere delta de signaal-ruis-verhouding in de eerste iteratie-slagen kleiner is. Ook geldt voor deze waarden van delta, dat tijdens het met maximale efficiëntie terugzetten van het object de signaal-ruis-verhouding stijgt. Dit is voor delta=1.1 niet het geval.
Te zien is dat de invloed van de snelheid waarmee de transparantie-efficiëntie groeit op de signaal-ruis-verhouding van het uiteindelijke hologram niet erg groot is. Dit komt omdat er tijdens het kwantisatie-proces amplitude-vrijheid op het object is en er gestopt wordt als de signaal-ruis-verhouding 100 is. Op dit moment is er weinig te zeggen over de invloed van de snelheid waarmee de transparantie-efficiëntie verhoogd wordt op de signaal-ruis-verhouding van het uiteindelijke hologram. Misschien dat deze snelheid een grotere invloed heeft bij een ander kwantisatie-proces. Bij dit proces is de invloed beperkt tot het aantal iteraties wat er in totaal nodig is. Voor de volledigheid is in de volgende figuur is het verloop van de distributie-efficiëntie weergegeven.



Figuur 4.13 : Verloop van de distributie-efficiëntie voor de verschillende verhogingsfactoren.

4.3 Verschillende begrenzers voor het iteratief verhogen van de transparantie-efficiëntie.

Het in amplitude begrenzen van de pixel-verdeling kan met verschillende begrenzers gebeuren. Het iteratie-proces is voor verschillende begrenzers bekeken. De invloed van deze verschillende begrenzers op het totale proces zal in deze paragraaf verder uitgewerkt worden.

4.3.1 De lineaire begrenzer.

Voor de lineaire begrenzer geldt

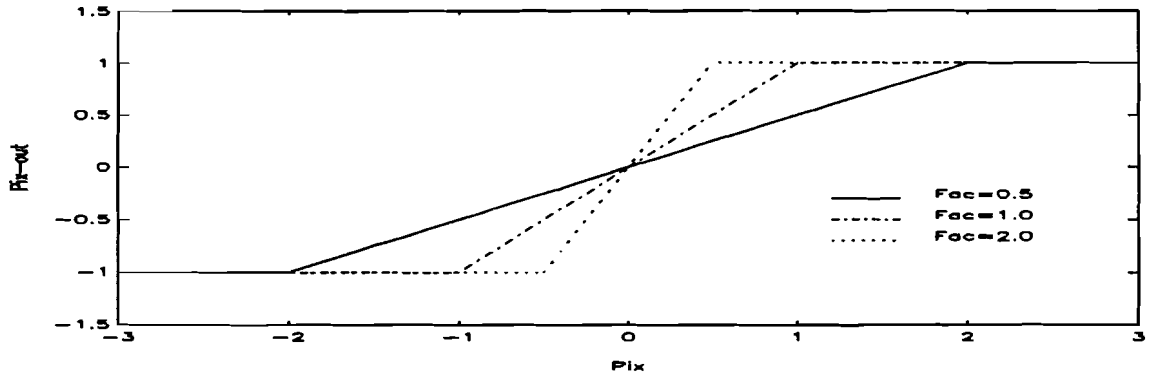
$$Pix_{out} = Saturation(Fac \cdot Pix). \tag{4.7}$$

Waarbij *Pix* de waarde van het pixel voor en Pix_{out} de waarde van het pixel na het clippen voorstelt. *Saturation(x)* is de saturatie-functie waarvoor geldt

$x < -1$	Saturation(x) = 1,
$-1 \leq x < 1$	Saturation(x) = x,
$x \geq 1$	Saturation(x) = 1.

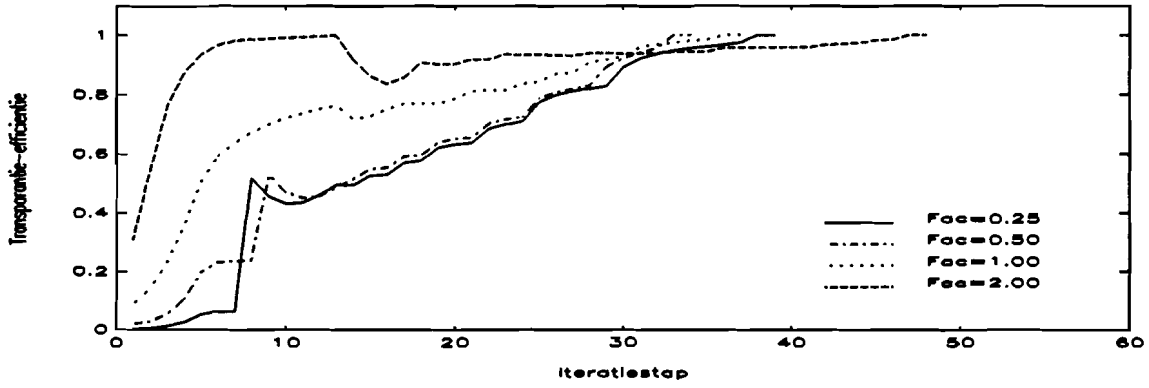
Fac is de variabele stijlheid van de begrenzer waarmee de transparantiewinst per begrenzing gevarieerd kan worden. De bedoeling van deze factor is het varieren van de waarde waar de transparantie-efficiëntie tijdens het begrenzen mee verandert ($\eta_{t\text{-}winst}$). Als *Fac* klein is zal het verlies aan transparantie-efficiëntie tijdens het begrenzen veel groter zijn. Hierdoor zal het algoritme het object veel groter in de reconstructie terugzetten om uiteindelijk de winst in transparantie-efficiëntie gelijk te maken aan de gewenste waarde. Dit zal van invloed zijn op het verloop van de distributie-efficiëntie.

In de volgende figuur is voor verschillende waarde van *Fac* de vorm van de lineaire begrenzer weergegeven.



Figuur 4.14 : Pix_{out} versus Pix voor een lineaire begrenzer met verschillende stijlheden.

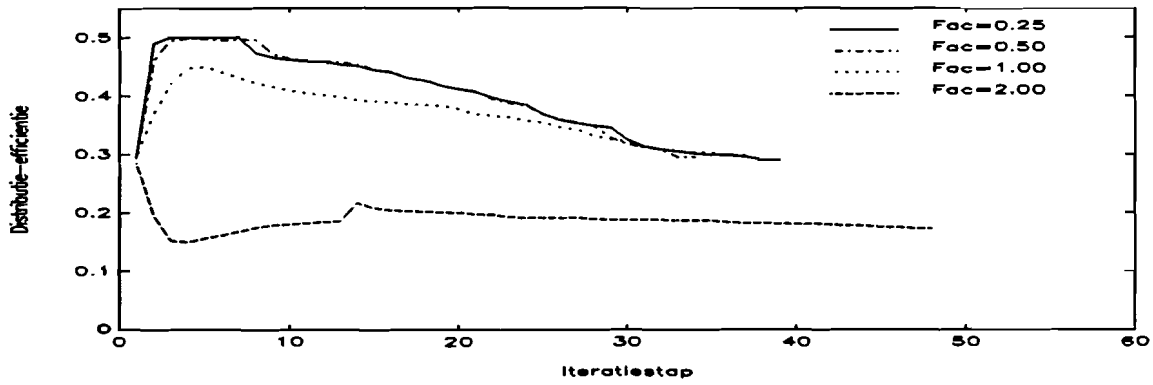
Het iteratie-proces is voor deze begrenzer met verschillende stijlheden bekeken. In de volgende figuur is het verloop van de transparantie-efficiëntie voor deze verschillende stijlheden uitgezet.



Figuur 4.15 : Verloop van de transparantie-efficiëntie bij het begrenzen met een lineaire begrenzer met verschillende stijlheden.

In deze figuur is te zien dat hoe groter de stijlheid van de begrenzer is hoe hoger de transparantie-efficiëntie bij het iteratieve verhogings-proces komt. De winst in transparantie-efficiëntie wordt echter teniet gedaan door het verlies aan distributie-efficiëntie.

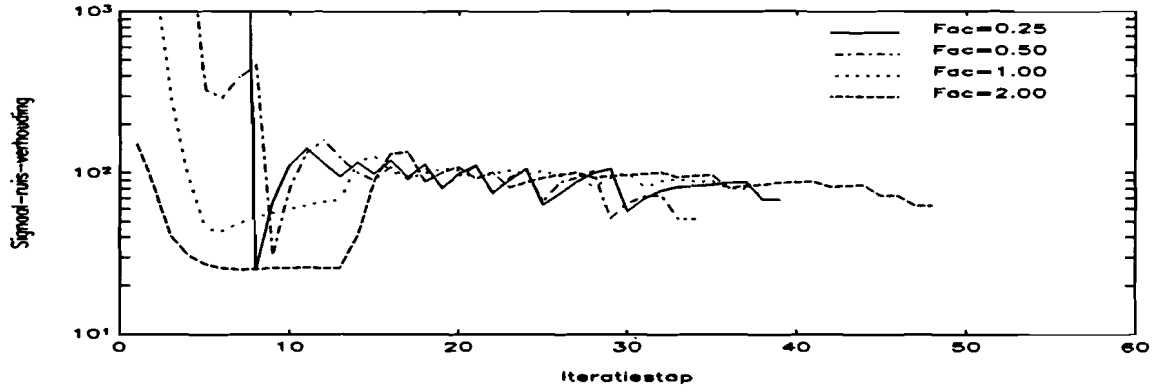
Dit is in de volgende figuur duidelijk te zien.



Figuur 4.16 : Verloop van de distributie-efficiëntie bij het verhogen van de transparantie-efficiëntie met een lineaire begrenzer met verschillende stijlheden.

De oorzaak hiervoor ligt in het feit dat bij een stijlheidsfactor 2.0 aan het begin van het iteratie-proces het object kleiner in de reconstructie teruggezet wordt. Hierdoor wordt de distributie-efficiëntie erg slecht. Het algoritme kan deze slechte distributie-efficiëntie in een latere fase niet meer corrigeren.

In de volgende figuur is het verloop van de signaal-ruisverhouding uitgezet.



Figuur 4.17 : Verloop van de signaal-ruis-verhouding bij het begrenzen van de pixelverdeling met een lineaire begrenzer met verschillende stijlheden.

Te zien is dat het voor de uiteindelijke signaal-ruis-verhouding niet erg veel uitmaakt met welke stijlheidsfactor er begrensd wordt. Over het verloop van de signaal-ruis-verhouding valt alleen te zeggen dat een grotere transparantie-efficiëntie gepaard gaat met een kleinere signaal-ruis-verhouding.

4.3.2 Niet lineaire begrenzers.

Het algoritme is ook bekeken voor niet-lineaire begrenzers. De volgende begrenzers zijn aan de orde gekomen.

- De arctangens-vormige begrenzer waarvoor geldt

$$Pix_{out} = \frac{2}{\pi} atan(Fac \cdot Pix). \tag{4.8}$$

- De exponentiële begrenzer waarvoor geldt

$$Pix_{out} = Saturation((Fac\,|Pix|)^{|Pix|})sign(Pix). \tag{4.9}$$

- De stapvormige begrenzer waarvoor geldt

$$Pix_{out} = Saturation[Fac + (1 - Fac)\,|Pix|]sign(Pix). \tag{4.10}$$

- De tangens-hyperbolicus-vormige begrenzer waarvoor geldt

$$Pix_{out} = Tanh(Fac\ Pix). \tag{4.11}$$

Geen van deze begrenzers leverde betere resultaten op dan de lineaire begrenzer. Dit is ook de reden waarom de resultaten van deze berekeningen hier niet verder uitgewerkt zijn.

4.4 Conclusies en aanbevelingen met betrekking tot het iteratief verhogen van de transparantie-efficiëntie.

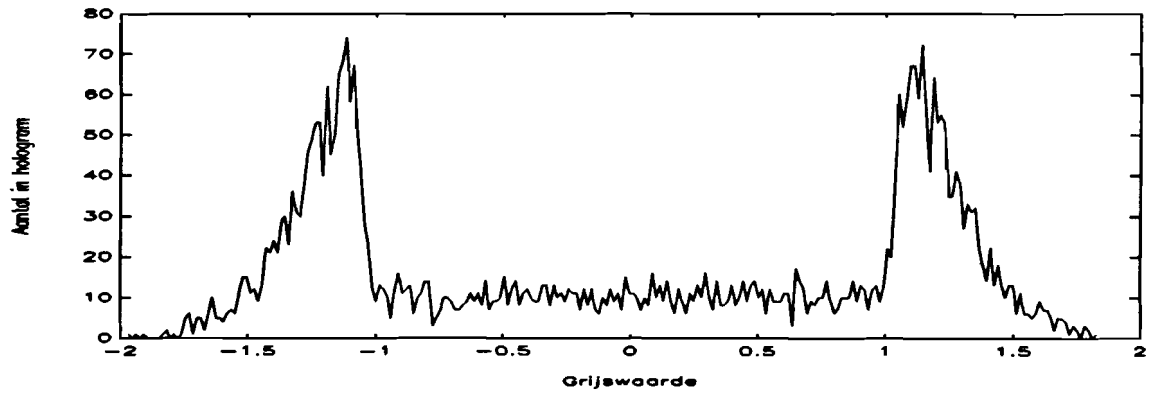
Het aangepaste Gerchberg en Saxton algoritme voor het iteratief verhogen van de transparantie-efficiëntie werkt zeer goed. Voor een optimale werking moeten de volgende regels in acht genomen worden,

- Van begin af aan moet er voldaan worden aan schaalfactor-, fase- en amplitude-vrijheid buiten het object. Amplitude-vrijheid op het object levert geen hogere transparantie-efficiëntie op, maar wel een lagere signaal-ruis-verhouding. Dit levert een slechter uitgangspunt om het kwantisatieproces te starten.

- De snelheid waarmee de transparantie-efficiëntie iedere iteratie groeit heeft heel weinig invloed op het verloop van het totale proces. Alleen een hele lage snelheid heeft merkbare gevolgen voor het verloop van het proces maar dit levert een zeer traag proces op en het uiteindelijke hologram wordt er kwalitatief niet beter van.

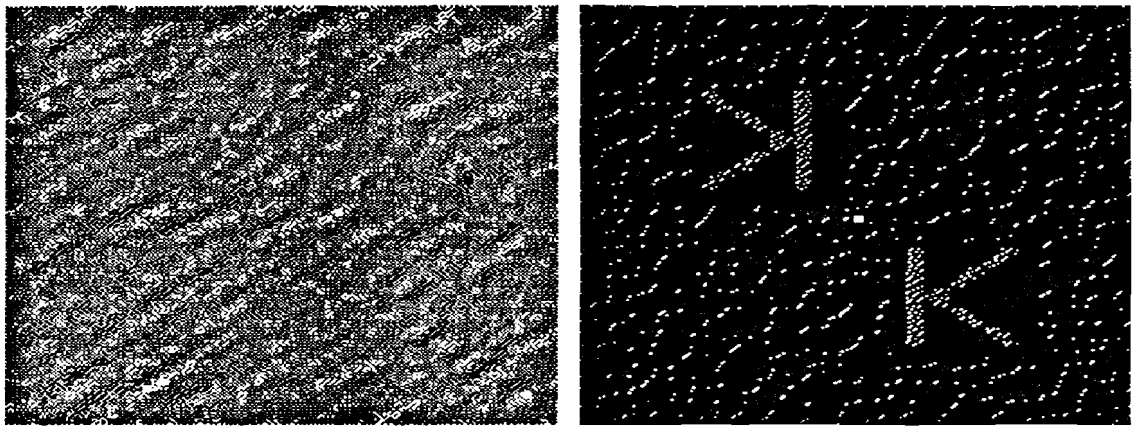
- Voor de begrenzer, die tijdens het proces zorgt voor het in amplitude begrensd blijven van het hologram is, levert een lineaire begrenzer de beste prestaties.

In de volgende figuur is de grijswaardeverdeling van een hologram, waarvan de transparantie-effiëntie iteratief volgens de besproken regels verhoogd is, weergegeven. Niet alle grijswaarden van dit hologram liggen in het interval [-1,1] omdat het hier een hologramverdeling betreft dat, na het terugzetten van het object in de reconstructie, berekend is.



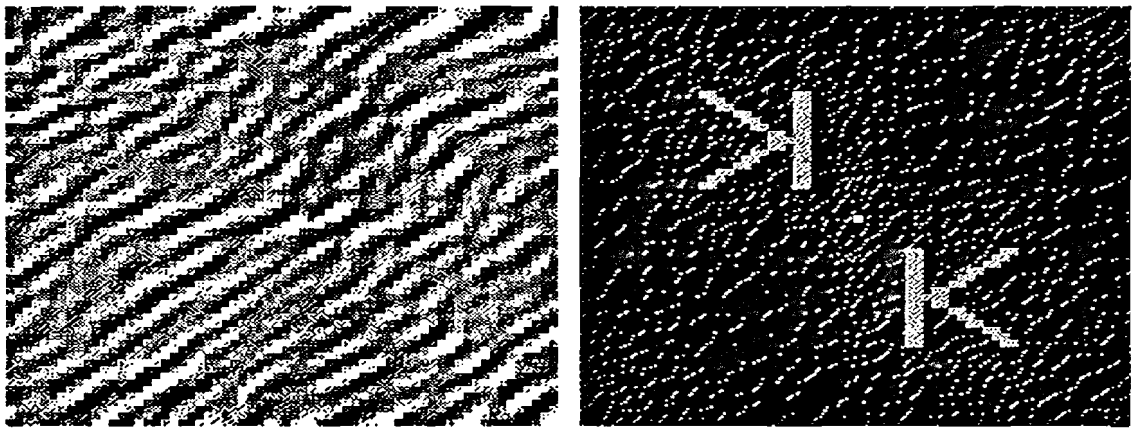
Figuur 4.18 : Grijswaardeverdeling van een hologram waarvan de transparantie-efficiëntie iteratief verhoogd is.

Ter illustratie zijn in de volgende figuren zijn het hologram en de bijbehorende reconstructie op bepaalde momenten van het iteratief verhogen van de transparantie-efficiëntie weergegeven.



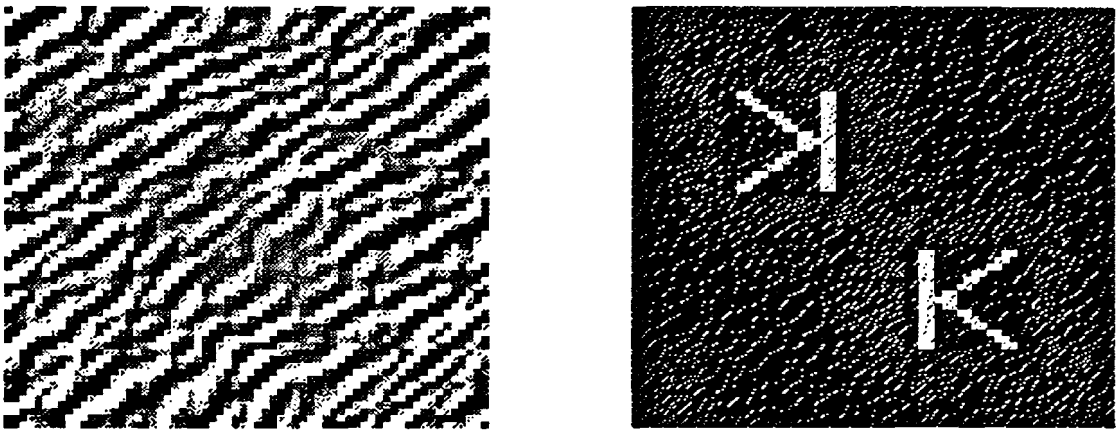
Hologram Reconstructie

Figuur 4.19 : Initiële hologram (iteratiestap=1) en de bijbehorende reconstructie (η=0.033, η_t=0.097).



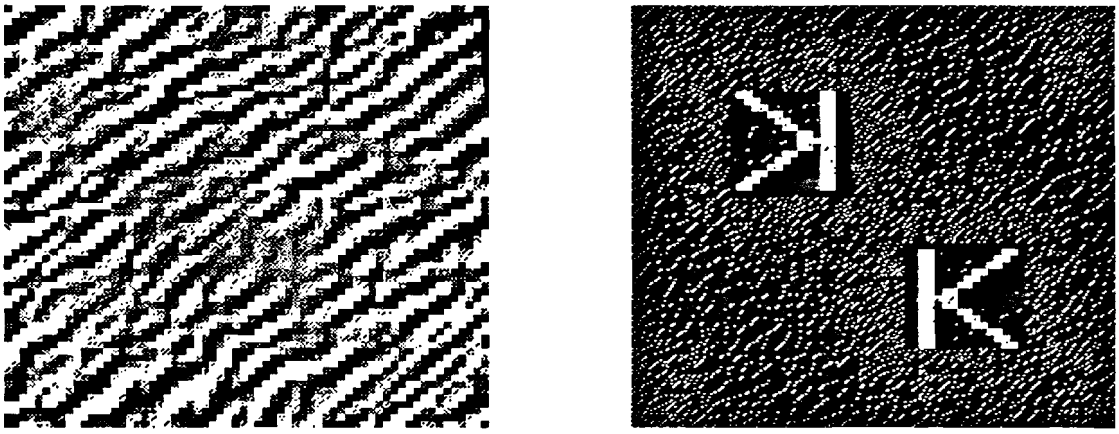
Hologram Reconstructie

Figuur 4.20 : Hologram en de bijbehorende reconstructie na vijf iteraties (η=0.250, η_t=0.567).



Hologram Reconstructie

Figuur 4.21 : Hologram en de bijbehorende reconstructie na 10 iteraties (η=0.292, η_t=0.718).



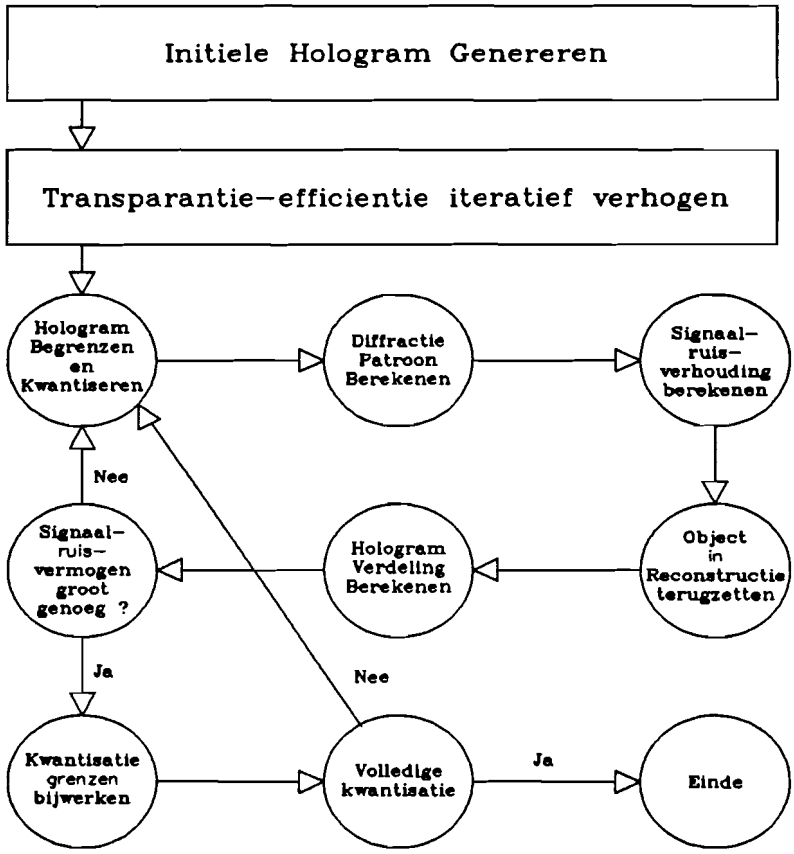
Hologram Reconstructie

Figuur 4.22 : Hologram en de bijbehorende reconstructie aan het einde van het iteratief verhogen van de transparantie-efficiëntie (η=0.304, η_t=0.777).

In het volgende hoofdstuk zal er gekeken worden hoe dit laatste hologram het beste gekwantiseerd kan worden.

5. Het kwantisatie-proces

In figuur 5.1 is het kwantisatie-proces nog eens gedetailleerd in het volledige algoritme weergegeven. In dit hoofdstuk zal deze laatste bewerking op het hologram nader uitgewerkt worden.



Figuur 5.1 : Totale kwantisatie-algoritme.

Het kwantiseren van het hologram gebeurt in de volgende vijf stappen.

1. Het in amplitude begrenzen en gedeeltelijk kwantiseren van de hologramverdeling. De vorm van deze kwantisator/begrenzer en het gedeelte van de pixelverdeling wat gekwantiseerd wordt is op dit moment nog niet van belang.

2. Het berekenen van de reconstructie die bij dit nieuwe hologram hoort.

3. De signaal-ruis-verhouding voor deze nieuwe reconstructie berekenen. Voor deze signaal-ruis-verhouding geldt

$$\frac{S}{N}=\frac{\sum_{u=0}^{\Delta u}\sum_{v=0}^{\Delta v}[\lambda FP(u,v)]^2}{\sum_{u=u1}^{u1+\Delta u}\sum_{v=v1}^{v1+\Delta v}[FP_t(u,v)-\lambda FP(u-u1,v-v1)]^2}. \tag{5.1}$$

Hierin is λ de schaalfactor zoals die in hoofdstuk 4 besproken is.

4. Het huidige object en zijn 'twin' in de reconstructie terugzetten. Er wordt hier niet naar gestreefd om de transparantie-efficiëntie te verhogen dus wordt het object met zijn huidige amplitude teruggezet, λ maal het originele object dus.

5. De bij de nieuwe reconstructie behorende pixel-verdeling berekenen.

De overige bolletjes uit figuur 5.1 zorgen dat het proces gestuurd wordt door de signaal-ruis-verhouding. Dit is namelijk de factor waar in dit deel van het totaal-algoritme het meeste belang aan gehecht wordt.

Na iedere iteratieslag wordt er gekeken of de berekende signaal-ruis-verhouding groot genoeg is. Het algoritme biedt namelijk aan de gebruiker de ruimte om een gewenste signaal-ruis-verhouding mee te geven. Als de berekende signaal-ruis-verhouding groter is dan de gewenste signaal-ruis-verhouding dan is hij groot genoeg. Dit is voor het algoritme het teken om de kwantisator zodanig bij te werken dat het zeker is dat een groter percentage van de hologram-verdeling gekwantiseerd zal worden tijdens stap 1. Om het algoritme tegen vastlopen te beveiligen is ook het stagneren van de signaal-ruis-verhouding een reden om de kwantisator bij te werken. Dit kan bijvoorbeeld gebeuren als de gebruiker een niet te realiseren signaal-ruis-verhouding opgeeft.

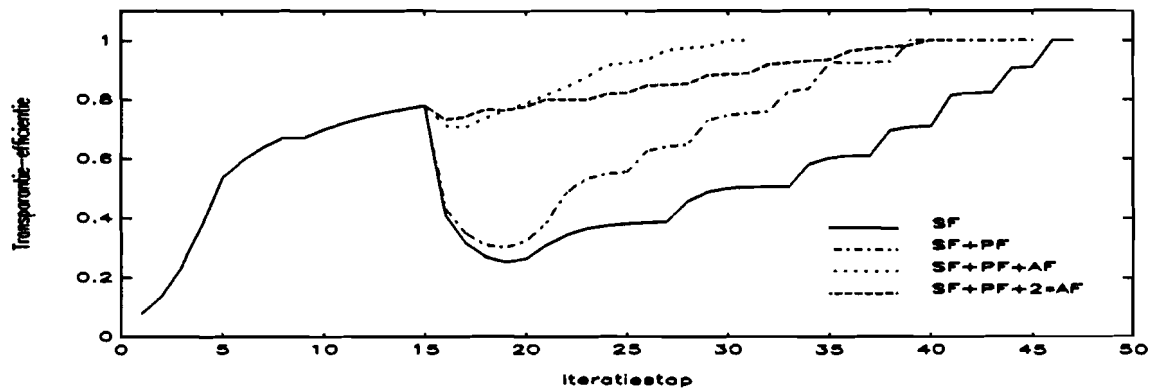
Als uiteindelijk alles gekwantiseerd wordt is het hologram een binair amplitude hologram en is het algoritme ten einde.

In paragraaf 5.1 zal gekeken worden wat de invloed van het benutten van de verschillende vrijheden tijdens dit kwantisatieproces is. In paragraaf 5.2 zal de invloed van het gebruik van verschillende kwantisatoren bekeken worden.

5.1 De invloed van de verschillende reconstructievrijheden tijdens het kwantisatieproces.

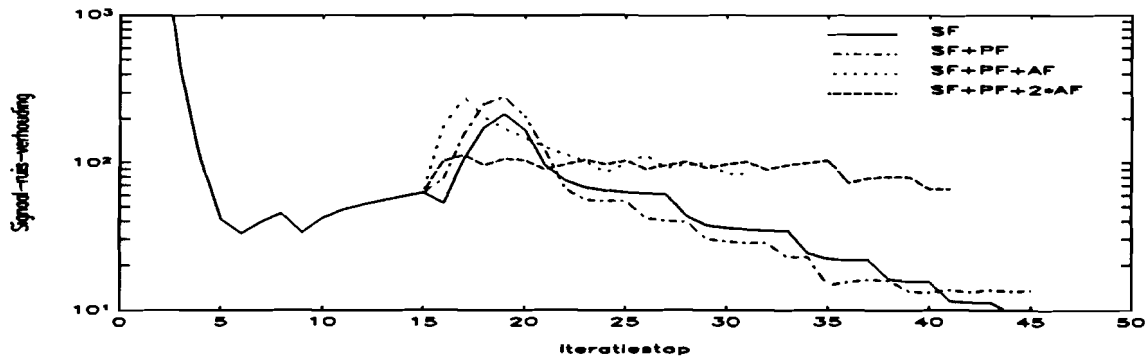
In deze paragraaf wordt de invloed van het benutten van de verschillende vrijheden bij het terugzetten van het object in de reconstructie bekeken. Hiervoor is wederom gebruik gemaakt van de stepwise begrenzer die in hoofdstuk 2 aan de orde is gekomen.

In de volgende figuur is het verloop van de transparantie-efficiëntie voor het terugzetten van het object met verschillende vrijhedencombinaties weergegeven.



Figuur 5.2 : Het verloop van de transparantie-efficiëntie voor het terugzetten van het object met verschillende vrijheden.

In deze figuur is te zien dat het algoritme bij het benutten van SF+PF+AF het snelst is. Fase-vrijheid en amplitude-vrijheid buiten het object zijn nodig om aan de gewenste signaal-ruis-verhouding te voldoen. Dit is beter te zien in de volgende figuur waarin het verloop van de signaal-ruis-verhouding uitgezet is.



Figuur 5.3 : Verloop van de signaal-ruis-verhouding voor het terugzetten van het object met verschillende vrijheden.

In deze figuur is goed te zien waarom amplitude-vrijheid op het object veel meer iteratiestappen nodig heeft. Dit komt omdat de signaal-ruis-verhouding zonder amplitude-vrijheid op het object veel sneller naar de gewenste waarde toe convergeert. Eigenlijk is er bij SF+PF+AF ook sprake van amplitude-vrijheid op het object, omdat het proces gestopt wordt als er aan een bepaalde signaal-ruis-verhouding voldaan wordt. Een bepaalde ruis-amplitude op het object wordt dan als goed genoeg gezien.

Het verloop van de distributie-efficiëntie is achterwege gelaten omdat dit geen nieuwe informatie oplevert. Wel is het zo dat wel of geen amplitude-vrijheid op het object voor de uiteindelijke distributie-efficiëntie niet van belang is.

In de volgende paragraaf worden verschillende kwantisatoren bekeken. Hierbij zal het terugzetten van het object geschieden met schaalfactor-, fase- en amplitude-vrijheid buiten het object. Fasevrijheid en amplitude-vrijheid buiten het object zijn noodzakelijk om een bepaalde signaal-ruis-verhouding te behalen. Amplitude-vrijheid op het object vertraagd het totale proces zonder dat hier een andere kwaliteitseis stijgt. Dit is alleen nadelig. Schaalfactor-vrijheid is altijd gewenst omdat dit een contrast-vergroting oplevert.

5.2 De invloed van verschillende kwantisatoren tijdens het kwantisatie-proces.

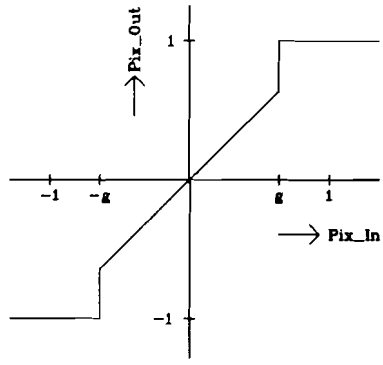
In deze paragraaf zal het gebruik van verschillende kwantisatoren en hun invloed op het uiteindelijke resultaat aan de orde komen.

5.2.1 De stepwise begrenzer met dalende grenswaarde.

Deze begrenzer is tot nu toe steeds gebruikt tijdens het kwantiseren van het hologram. De karakterisiek van deze begrenzer is in figuur 5.4 weergegeven.

Deze begrenzer wordt hier een begrenzer met dalende grenswaarde g genoemd, omdat de grens waarboven het hologram gekwantiseerd wordt tijdens het iteratie-proces daalt.

Een nadeel van deze begrenzer is, dat de laatste keer dat de grenswaarde verlaagd wordt, de pixels die tot dan toe nog niet gekwantiseerd zijn een hele grote sprong maken. De begrenzer gaat dan over in een hard limiter en het iteratie-

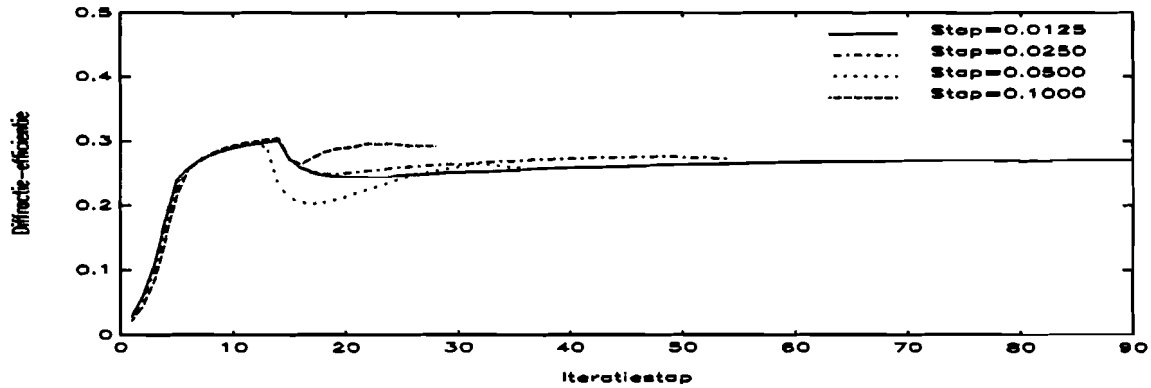


Figuur 5.4 : Overdrachtskarakteristiek van de stepwise begrenzer met dalende grenswaarde.

proces stagneert direct.

Om deze laatste sprong voor zo min mogelijk pixels noodzakelijk te maken moet de stapgrootte van de grenswaarde tijdens het iteratie-proces zo klein mogelijk gemaakt worden. Om te kijken hoe deze stapgrootte van invloed is op het kwantisatieproces is de volledige berekening voor verschillende stapgroottes uitgevoerd.

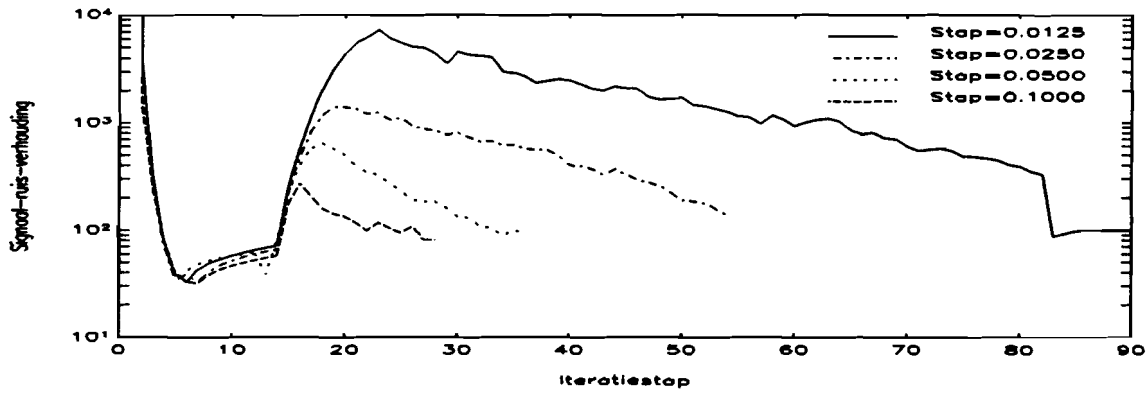
In de volgende figuur is het verloop van de diffractie-efficiëntie voor verschillende stapgroottes weergegeven.



Figuur 5.5 : Verloop van de diffractie-efficiëntie voor verschillende stapgroottes.

Het verschil in de uiteindelijk diffractie-efficiëntie is bij deze stapgroottes niet zeer groot. De lengte van het totale iteratie-proces is echter wel ingrijpend van de stapgrootte afhankelijk.

In de volgende figuur is het verloop van de signaal-ruis-verhouding weergegeven.

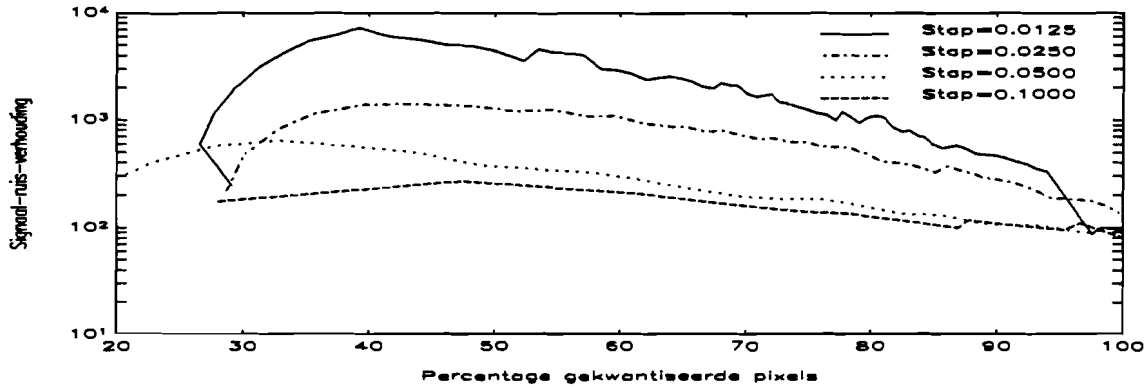


Figuur 5.6 : Verloop van de signaal-ruis-verhouding voor verschillende stapgroottes.

Voor de uiteindelijke signaal-ruis-verhouding is de stapgrootte wel van belang. Deze ligt namelijk tussen 90 voor een stapgrootte van 0.1 en 150 voor een stapgrootte van 0.025.

Het vreemde is echter dat voor een hele kleine stapgrootte van 0.0125 de signaal-ruis-verhouding op het laatste moment een grote sprong maakt en slechter uitkomt dan voor een grotere stapgrootte.

Om te kijken hoe het komt dat de stapgrootte van invloed is op de uiteindelijke signaal-ruis-verhouding is in figuur 5.7 de signaal-ruis-verhouding tegen het percentage gekwantiseerde pixels uitgezet.

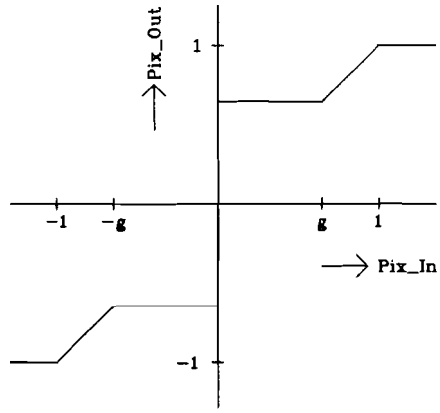


Figuur 5.7 : Signaal-ruis-verhouding versus percentage gekwantiseerde pixels voor verschillende stapgroottes.

Uit deze figuur blijkt dat bij een kleinere stapgrootte op het moment dat bijvoorbeeld de helft van de pixels gekwantiseerd wordt de signaal-ruis-verhouding veel beter is dan bij een grotere stapgrootte. Blijkbaar is het trajekt wat door het hologram doorlopen wordt veel beter bij een kleinere stapgrootte. Het kan echter gebeuren dat er een snelle terugval voor de signaal-ruis-verhouding optreedt zoals hier bij een stapgrootte van 0.0125 het geval is.

5.2.2 De stepwise begrenzer met stijgende grenswaarde.

Het nadeel van de stepwise begrenser met dalende grenswaarde is dat de laatste verlaging van de grenswaarde de grootste pixelverschuiving oplevert. Dit kan opgevangen worden door een stepwise begrenzer met stijgende grenswaarde voor de kwantisatie te gebruiken. Een overdrachtskarakteristiek voor deze kwantisator is in figuur 5.8 weergegeven.



Figuur 5.8 : De overdrachtskarakteristiek van de stepwise begrenzer met stijgende grenswaarde.

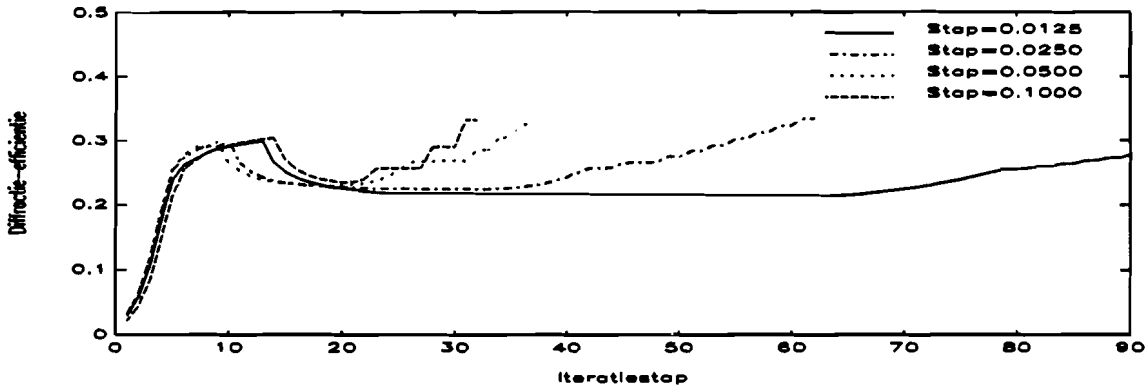
Deze kwantisator clipt alle pixels waarvan de modulus kleiner is dan een grenswaarde, afhankelijk van hun teken, naar plus of min deze grenswaarde. Alle pixels met een modulus groter dan 1 worden, afhankelijk van hun teken, naar +1 of -1 geclipt.

Het kwantisatie-proces begint hierbij met een grenswaarde van 0 en eindigt met een grenswaarde van 1. De karakteristiek van deze begrenzer is zowel bij het begin als bij het einde van het kwantisatieproces gelijk aan de karakteristiek van de stepwise begrenzer met dalende grenswaarde.

Deze begrenzer heeft echter ook een nadeel en wel het volgende. Stel dat de grens de waarde 0.5 bereikt heeft. Dit betekent dat na het clippen met deze kwantisator alle pixels een modulus groter of gelijk aan 0.5 zullen hebben. Wil het proces niet stagneren dan zullen er, door het terugzetten van het object in de reconstructie, pixels moeten zijn die in absolute waarde meer dan 0.5 veranderen. Alle pixels die minder dan 0.5 in absolute waarde veranderen worden namelijk door de kwantisator weer op de grenswaarde met hetzelfde teken afgebeeld. Hoe hoger de grenswaarde wordt hoe moeilijker het hologram verandert.

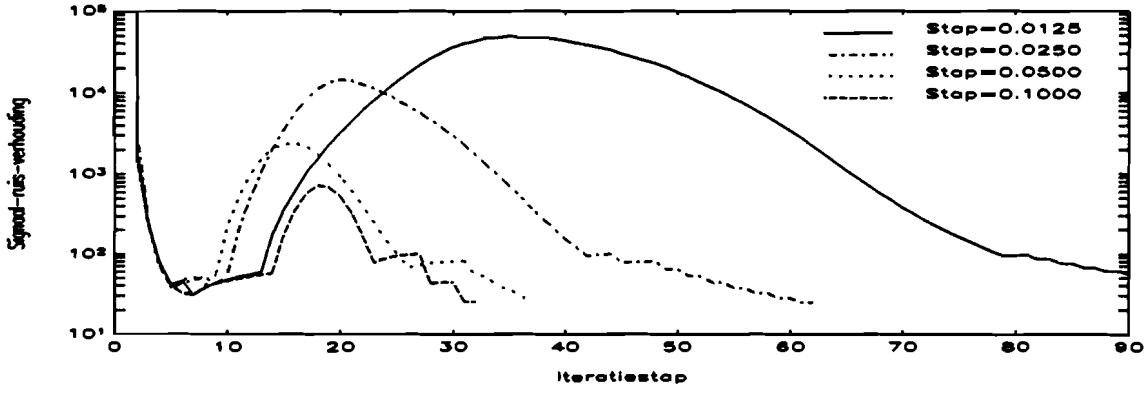
Om te kijken hoe de grootte van de stap, waarmee de grenswaarde van deze kwantisator verhoogd wordt van invloed is op het kwantisatie-proces is deze berekening voor verschillende stapgroottes uitgevoerd.

In de volgende figuur is het verloop van de diffractie-efficiëntie voor verschillende stapgroottes weergegeven.



Figuur 5.9 : Verloop van de diffractie-efficiëntie voor verschillende stapgroottes.

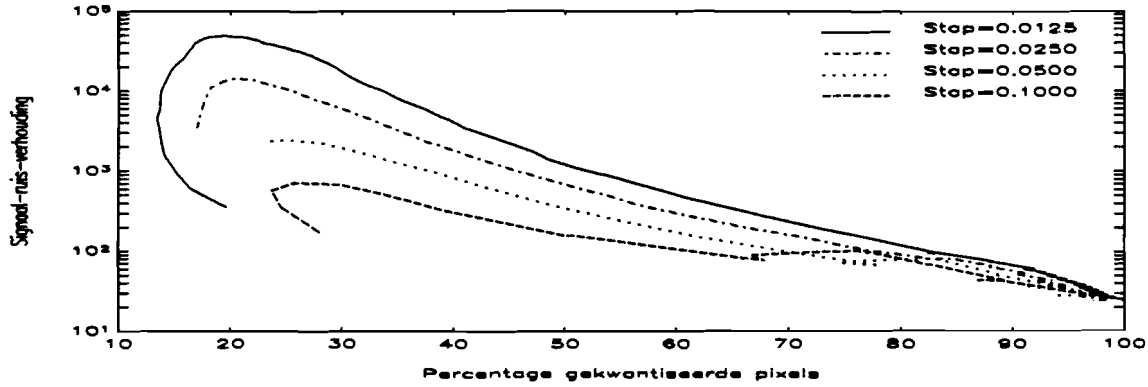
Blijkbaar is de stapgrootte niet van groot belang voor de uiteindelijke diffractie-efficiëntie van het hologram.
In de volgende figuur is het verloop van de signaal-ruis-verhouding weergegeven.



Figuur 5.10 : Verloop van de signaal-ruis-verhouding voor verschillende stapgroottes.

In deze figuur is te zien dat tijdens het eerste stuk van het kwantisatie-algoritme, dat bij iteratiestap 13 begint, de signaal-ruis-verhouding stijgt. Dit stijgen duurt totdat de grens een bepaalde waarde aangenomen heeft. Daarna begint de signaal-ruis-verhouding te dalen. Zolang de signaal-ruis-verhouding groter dan 100 is wordt de grenswaarde iedere iteratieslag verhoogd. Pas als de signaal-ruis-verhouding onder de 100 is gekomen begint het verloop trapvormig te worden. Dit komt omdat vanaf dit moment het stagneren van de signaal-ruis-verhouding de grenswaarde doet stijgen. Zelfs bij een zeer kleine stapgrootte voldoet de signaal-ruis-verhouding op het einde van het proces niet aan de gestelde eis.

In de volgende figuur is de signaal-ruis-verhouding tegen het aantal gekwantiseerde pixels uitgezet.

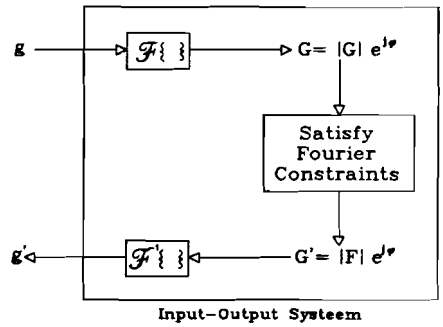


Figuur 5.11 : Signaal-ruis-verhouding versus percentage gekwantiseerde pixels.

Te zien is dat, zodra er 80 procent van de pixels geclipt worden, de signaal-ruis-verhouding onder de gewenste waarde van 100 zakt. Vanaf dat moment is het verschil tussen de vier krommes erg klein. Blijkbaar is het verloop van de signaal-ruis-verhouding afhankelijk van de stapgrootte zolang de signaal-ruis-verhouding boven de gewenste waarde blijft. Zodra deze onder de gewenste waarde komt gaat het stagneren van het proces de belangrijkste rol vervullen.

5.2.3 Toepassing van het output-output algoritme voor de kwantisatie.

In de literatuur [10] wordt het output-output algoritme gebruikt om het probleem van de intensiteit-metingen op te lossen. Het output-output algoritme wordt afgeleid van het input-output algoritme. Dit input-output algoritme ziet het bereken van de reconstructie, het terugzetten van het object in de reconstructie en het berekenen van de nieuwe hologram-verdeling als één systeem. Dit systeem is in figuur 5.12 weergegeven.



Figuur 5.12 : Schematische weergave van het input-output systeem.

Als er een verandering $\Delta g(x)$ in de input van dit systeem optreedt dan zal dit in een eerste orde benadering een verandering $\alpha\Delta g(x)$ in de output opleveren. De gewenste output-waarde is plus of min een. Dit geeft een gewenste outputverandering $\alpha\Delta g(x)$ waarvoor geldt

$$\alpha\Delta g(x)=[1-g'(x)]sign(g'(x)). \tag{5.2}$$

Hierdoor wordt de nieuwe input van het systeem

$$g_{k+1}(x)=g_k(x)+\frac{1}{\alpha}[1-g'_k(x)]sign(g'_k(x)). \tag{5.3}$$

Dit is de formule voor het input-output algoritme. Het output-output algoritme gaat echter uit van het feit dat als de input gelijk wordt aan $g'(x)$ de output ook $g'(x)$ wordt omdat $g'(x)$ al aan de Fourier-constraints voldoet. Hierdoor kan $g(x)$ in (5.3) vervangen worden door $g'(x)$. De formule voor het output-output algoritme wordt dan

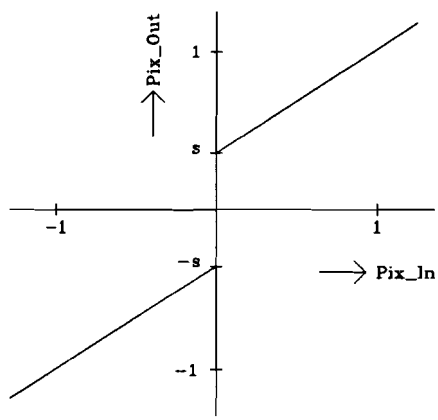
$$g_{k+1}(x)=g'_k(x)+\frac{1}{\alpha}[1-g'_k(x)]sign(g'_k(x))=g'_k(x)+s[1-g'_k(x)]sign(g'_k(x)). \tag{5.4}$$

Waarbij s de spronggrootte is van pixels die met een amplitude van nul de begrenzer inkomen.

De karakteristiek van deze kwantisator is in figuur 5.13 weergegeven. De enige stabiele punten van deze kwantisator zijn (-1,-1) en (1,1).

Het kwantiseren van deze kwantisator zit hem in het feit dat zolang s tussen 1 en -1 ligt voor alle pixels geldt dat zij na het clippen dichter bij 1 of -1 liggen.

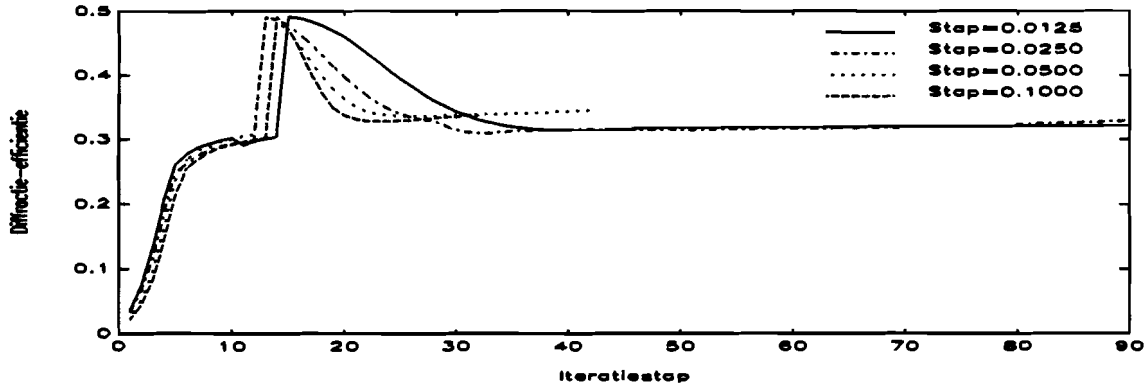
De transparantie-efficiëntie zal bij het kwantiseren met deze kwantisator stagneren zodra de pixel-verdeling uniform rond 1 en -1 ligt. Een pixel met amplitude 1-δ zal namelijk net zoveel transparantie-efficiëntie winnen als een pixel met amplitude $1+\delta$ zal verliezen.



Figuur 5.13 : Overdrachtskarakteristiek van de kwantisator die door het ouput-output algoritme beschreven wordt.

Als de signaal-ruis-verhouding stagneert of voldoet aan de gewenste waarde dan zal de sprong s vergroot worden. Uiteindelijk zal deze s gelijk aan 1 worden en gaat de kwantisator over in een hard limiter. Om te kijken hoe de waarde, waarmee de sprong s iedere keer verhoogd wordt, van invloed is op het verloop van dit kwantisatieproces en op het uiteindelijke resultaat voeren we de totale berekening voor verschillende verhogingswaarden uit.

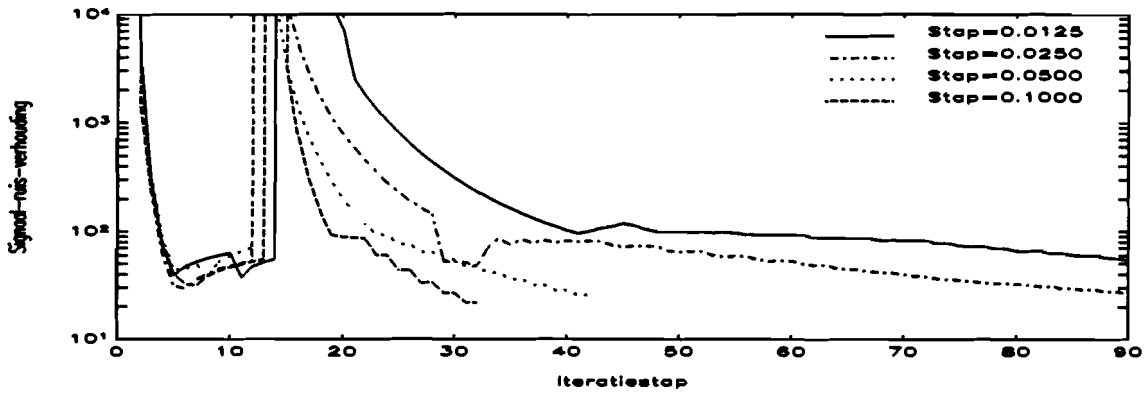
In de volgende figuur is voor verschillende verhogingswaarden de diffractie-efficiëntie weergegeven.



Figuur 5.14 : Verloop van de diffractie-efficiëntie voor verschillende verhogingswaarden.

Te zien is dat de diffractie-efficiëntie aan het begin van het kwantisatie-proces (Iteratiestap=12) een sprong naar de theoretisch maximaal haalbare waarde van 0.5 maakt. Dit komt omdat aan het einde van het iteratief verhogen van de transparantie-efficiëntie het object met deze maximale efficiëntie teruggezet wordt. De eerste keer dat er gekwantiseerd wordt is de grootte van de sprong s gelijk aan 0. De kwantisator-karakteristiek gaat dan over op $Pix_{out}=Pix_{in}$ en het hologram wordt niet veranderd. De bijbehorende reconstructie is dan dezelfde als die voor het kwantiseren en heeft dan een oneindig grote signaal-ruis-verhouding en een maximale diffractie-efficiëntie van 0.5.

Deze sprong in de signaal-ruis-verhouding is ook in de volgende figuur te zien waar het verloop van de signaal-ruis-verhouding weergegeven is.

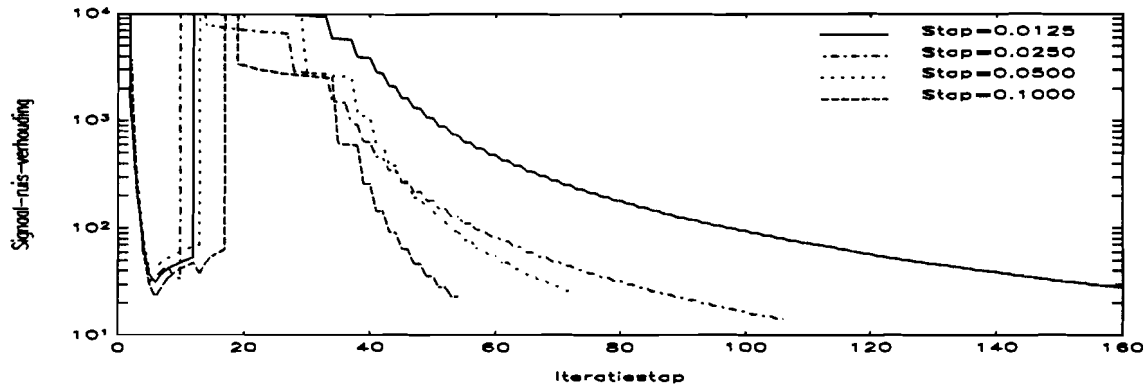


Figuur 5.15 : Verloop van de signaal-ruis-verhouding voor verschillende verhogingswaarden.

Deze figuur geeft goed weer dat de signaal-ruis-verhouding alleen maar blijft dalen. Het lijkt echter alsof dit komt omdat aan het begin van het kwantisatieproces de signaal-ruis-verhou-

ding ver boven de gewenste waarde ligt. Hierdoor wordt iedere iteratie de kwantisatorsprong s verhoogd. Vandaar dat hier iedere iteratie de signaal-ruis-verhouding daalt. Pas als de signaal-ruis-verhouding onder de gewenste waarde komt begint het stagneren van deze signaal-ruis-verhouding het proces te sturen. De sprong s is op dit moment echter zo groot dat vanaf dit moment de signaal-ruis-verhouding ook niet meer stijgt maar direct stagneert.

Om dit vermoeden nader te bekijken zijn deze berekeningen nogmaals uitgevoerd maar nu met een veel grotere gewenste signaal-ruis-verhouding van 100.000. Voor deze berekening is het verloop van de signaal-ruis-verhouding in de volgende figuur weergegeven.



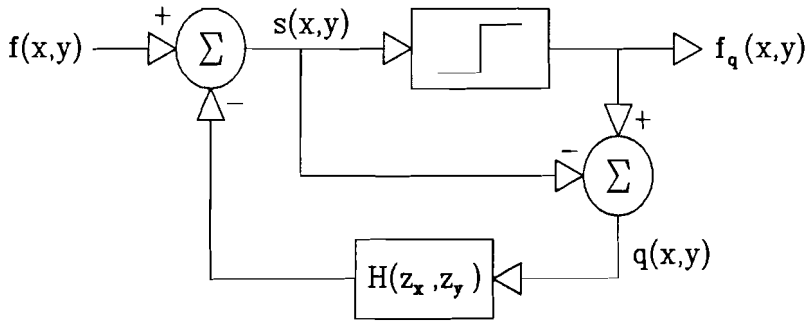
Figuur 5.16 : Verloop van de signaal-ruis-verhouding voor verschillende stapgroottes en een gewenste signaal-ruis-verhouding van 100.000.

In deze figuur is te zien dat het vermoeden onterecht was en dat ook bij een zo grote gewenste signaal-ruis-verhouding de signaal-ruis-verhouding blijft dalen. Blijkbaar levert het output-output algoritme geen afname van de ruis op het object op. Een reden hiervoor ligt in het feit dat de begrenzer maar twee stabiele punten heeft. Bijna alle pixels worden tijdens het begrenzen in amplitude gewijzigd. Hierdoor zal het proces nooit naar een oplossing met een kleine fout convergeren.

5.2.4 Toepassing van error-diffusie voor de kwantisatie.

Om de iteratieve kwantisatie-methoden te kunnen vergelijken met een niet-iteratieve is er gekeken naar kwantisatie van het hologram met behulp een veel gebruikt kwantisatieproces [13] namelijk error-diffusie. Bij error diffusie wordt de fout die bij het kwantiseren van een pixel geïntroduceerd wordt over de nog te kwantiseren pixels uitgespreid.

In de volgende afbeelding is het error-diffusie-proces grafisch weergegeven.



Figuur 5.17 : Blokschema van het error-diffusie-proces.

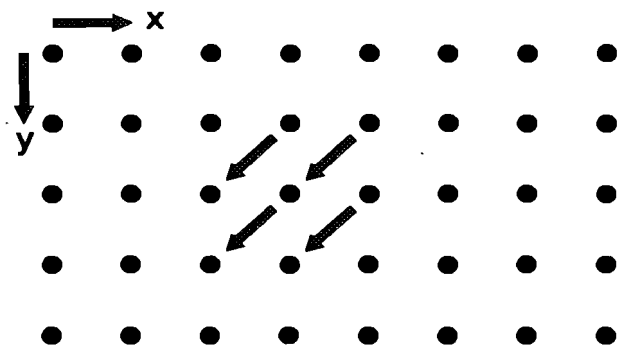
Voor de kwantisatieruis geldt

$$E_{quant}=f(x,y)-f_q(x,y)=(1-H(z_x,z_y))E_{clip}, \tag{5.5}$$

waarbij E_{clip} de ruis van de kwantisator voorstelt.

Omdat het hier alleen gaat om het iteratief kwantiseren te vergelijken met een niet iteratieve methode, is ervoor gekozen om het error-diffusie proces niet tot het uiterste uit te werken. Vandaar dat er is gekozen om de fout in een richting namelijk diagonaal, loodrecht op de lijn van het object en zijn twin, mee te nemen.

In de volgende figuur is dit weergegeven.



Figuur 5.18 : Richting van het meenemen van de kwantisatie-fout.

De mate waarin de kwantisatie-fout meegenomen wordt, wordt bepaald door de foutverspreidingsfactor α, zodat voor de fout-verdelings-functie H geldt

$$H=\alpha z_x^{+1} z_y^{-1} \tag{5.6}$$

Bij deze fout-verdelings-functie geldt voor de modulus van de kwantisatieruis

$$|E_{quant}| = |(1-H)E_{clip}| = |(1-\alpha z_x^{+1} z_y^{-1})|\,|E_{clip}| = C\,|E_{clip}| \text{ met } C = |1-\alpha z_x^{+1} z_y^{-1}| \tag{5.7}$$

Na transformatie naar het frekwentie-domein geldt

$$C(u,v) = 1-\alpha e^{j\frac{2\pi}{N}u} e^{-j\frac{2\pi}{N}v} \tag{5.8}$$

en

$$|C(u,v)| = \sqrt{[1-\alpha\cos(\frac{2\pi}{N}(u-v))]^2 + \alpha^2\sin^2(\frac{2\pi}{N}(u-v))} = \sqrt{1+\alpha^2-2\alpha\cos(\frac{2\pi}{N}(u-v))}. \tag{5.9}$$

Voor het vermogen van de ruis over het object geldt dan

$$N = \sum_{u=u1}^{u1+\Delta u} \sum_{v=v1}^{v1+\Delta v} |C(u,v)|^2\,|E_{clip}|^2. \tag{5.10}$$

De kwantisatieruis veronderstellen we wit zodat het minimaliseren van het ruis-vermogen over het object betekent het minimaliseren van

$$J = \sum_{u=u1}^{u1+\Delta u} \sum_{v=v1}^{v1+\Delta v} 1+\alpha^2-2\alpha\cos(\frac{2\pi}{N}(u-v)). \tag{5.11}$$

Voor de extrema geldt

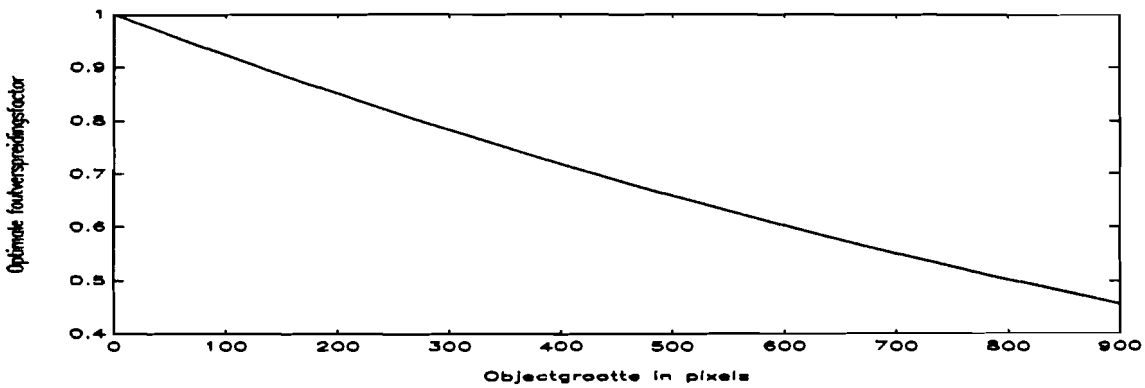
$$\frac{\partial J}{\partial \alpha} = 2\alpha\Delta u\Delta v - 2\sum_{u=u1}^{u1+\Delta u} \sum_{v=v1}^{v1+\Delta v} \cos(\frac{2\pi}{N}(u-v)) = 0. \tag{5.12}$$

Voor de optimale foutverspreidingsfactor geldt dan

$$\alpha_{opt} = \frac{1}{\Delta u\Delta v} \sum_{u=u1}^{u1+\Delta u} \sum_{v=v1}^{v1+\Delta v} \cos(\frac{2\pi}{N}(u-v)). \tag{5.13}$$

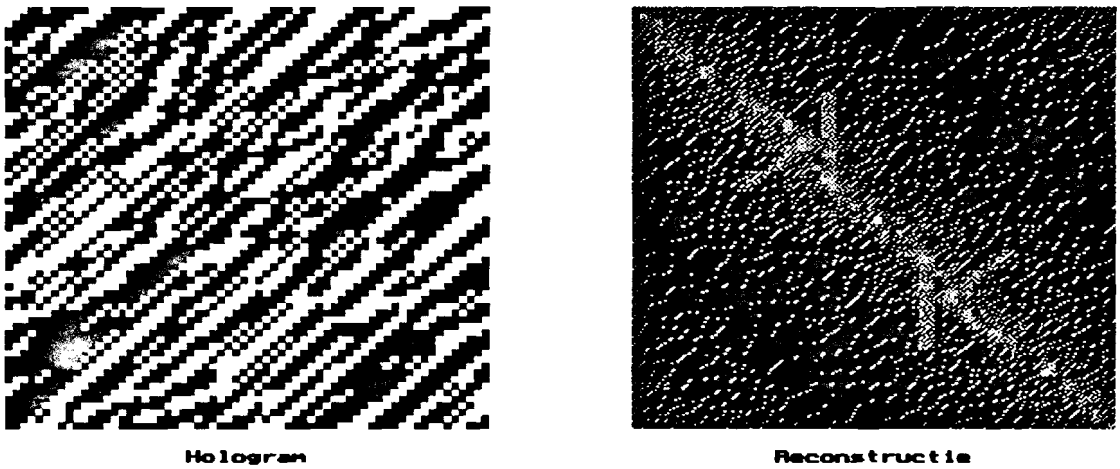
Volgens (5.12) is dit een minimum.

De optimale foutverspreidingsfactor volgens (5.13) is onafhankelijk van de plaats van het object op de hoofddiagonaal. Als het object op de hoofddiagonaal verschoven wordt veranderen voor de gehele sommatie u en v met dezelfde waarde zodat u-v hetzelfde blijft. De optimale foutverspreidingsfactor is wel afhankelijk van de grootte van het object. In de volgende figuur is voor een reconstructie van 64x64 voor verschillende objectgroottes de optimale foutverspreidingsfactor weergegeven.



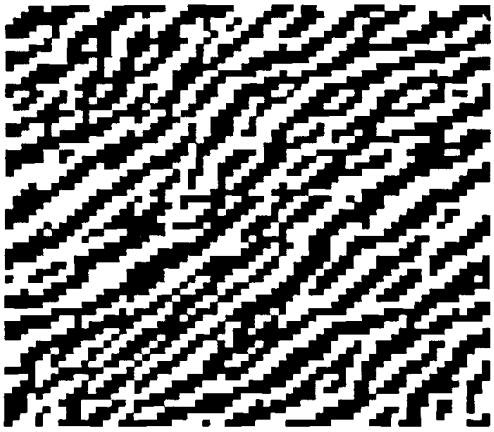
Figuur 5.19 : De optimale foutverspreidingsfactor voor verschillende objectgroottes.

Voor verschillende waarden van α is het hologram, na het iteratief verhogen van de transparantie-efficiëntie, met error diffusie gekwantiseerd. In de volgende figuren is het zo berekende hologram en de bijbehorende reconstructie voor drie interessante waarden van α weergegeven.

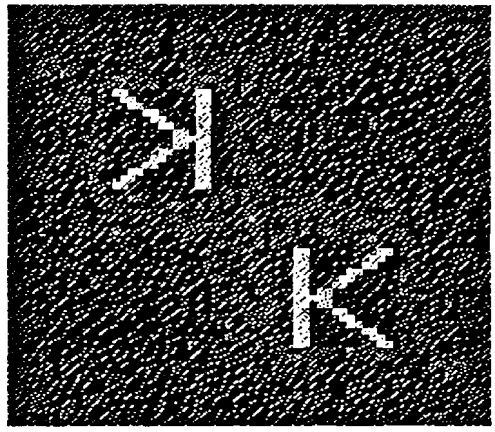


Figuur 5.20 : Hologram met bijbehorende reconstructie gekwantiseerd met error diffusie met een foutverspreidingsfactor α=-1.0.

In deze figuur is het cosinus-vormige verloop van de ruis met een maximum op de hoofddiagonaal *(u-v=0)*. Dit komt overeen met de theoretische verwachting voor de ruisverdeling uit (5.9).



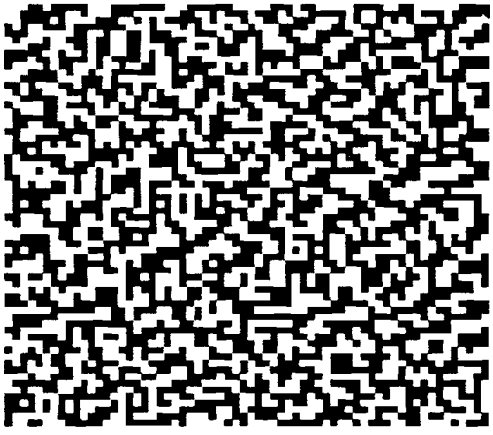
Hologram



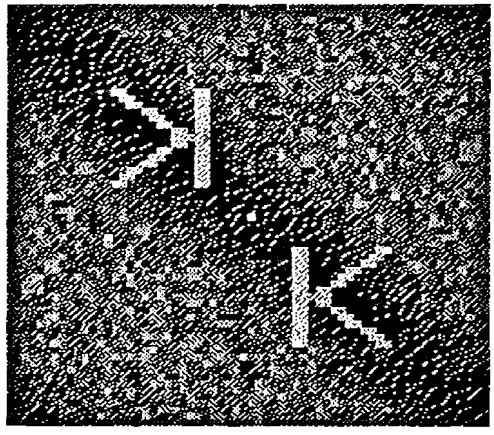
Reconstructie

Figuur 5.21 : Hologram met bijbehorende reconstructie gekwantiseerd met error-diffusie met een foutverspreidingsfactor $\alpha=0$.

Een foutverspreidingsfactor van 0 komt overeen met een hard limiter. De ruis op de reconstructie komt nu overeen met de kwantisatorruis E_{clip}. In deze figuur is te zien dat de aanname van witte kwantisatorruis vrij goed met de praktijk overeenkomt.



Hologram

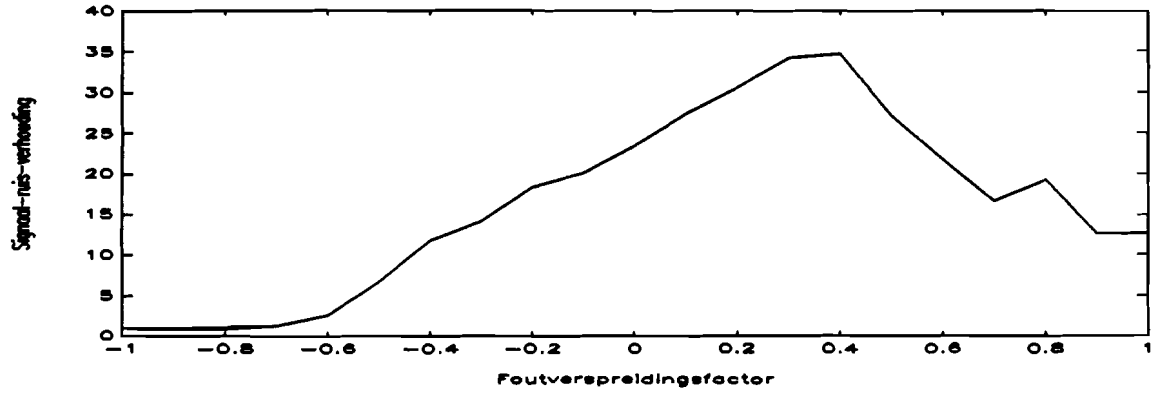


Reconstructie

Figuur 5.22 : Hologram met bijbehorende reconstructie gekwantiseerd met error-diffusie met een foutverspreidingsfactor $\alpha=1$.

Ook in deze figuur is het cosinus-vormige verloop van de ruis, maar nu met een minimum op $(u\text{-}v=0)$, goed te zien.

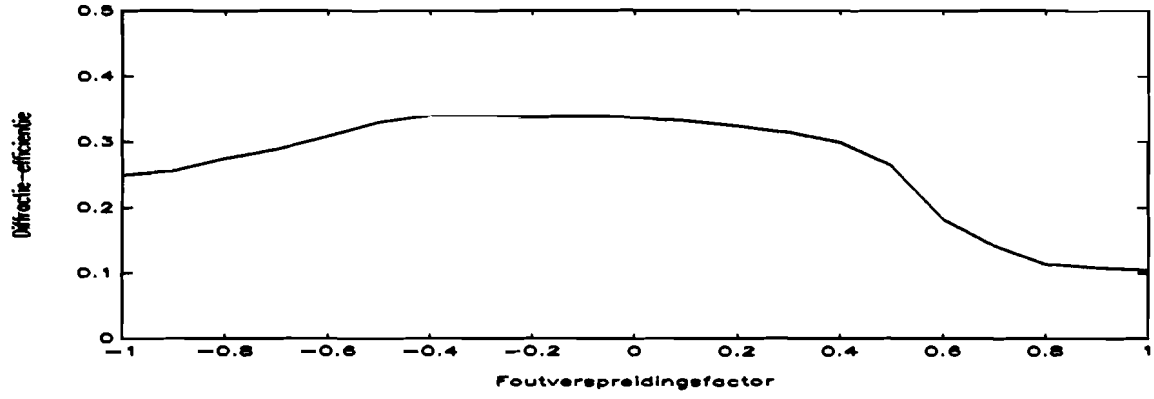
In de volgende figuur is de berekende signaal-ruis-verhouding als functie van de foutverspreidingsfactor weergegeven.



Figuur 5.23 : Signaal-ruis-verhouding als functie van de foutverspreidingswaarde bij error-diffusie.

Volgens deze berekening is de optimale foutverspreidingsfactor voor een 16x16 object 0.4. Als we de optimale foutverspreidingsfactor voor dit formaat object met (5.13) uitrekenen geeft dit een waarde, 0.81, die ongeveer twee keer zo groot is.

In de volgende figuur is de diffractie-efficiëntie voor deze verschillende foutverspreidingsfactoren weergegeven.



Figuur 5.24 : De diffractie-efficiëntie als functie van de foutverspreidingsfactor.

In deze figuur is te zien dat bij een foutverspreidingsfactor van 0.8 het deel van het licht wat op het object terecht komt ongeveer drie keer zo klein is als bij een foutverspreidingsfactor van 0.4. Het feit dat de optimale foutverspreidingsfactor wat betreft de signaal-ruis-verhouding niet op 0.8 ligt komt doordat bij deze foutverspreidingsfactor het signaal-vermogen veel kleiner is dan bij een foutverspreidingsfactor van 0.4. Hierdoor wordt de signaal-ruis-verhouding veel slechter.
In deze afbeeldingen is te zien dat met een niet iteratieve kwantisatiemethode een signaal-ruis-verhouding van ongeveer 40 te behalen is. Dit is veel minder dan met de besproken iteratieve kwantisatie-methode van een kwantisator met een dalende clipgrens.

5.3 Conclusies en aanbevelingen met betrekking tot het kwantisatie-proces.

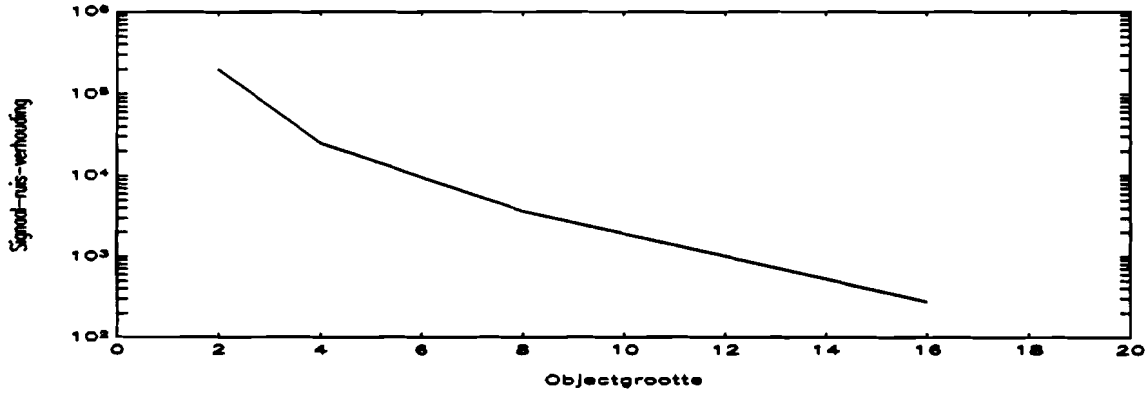
Met het aanpassen van het Gerberch-Saxton algoritme zijn we gekomen tot een zeer goede iteratieve kwantisatie-methode. Door de lokale operaties in het spectrum, die hier overeenkomen met niet-causale operaties in het hologramvlak, is deze kwantisatie-methode veel krachtiger dan een algoritme dat opgebouwd is uit causale operaties in het hologramvlak, zoals bijvoorbeeld error-diffusie. Om het uiterste uit het proces te halen moeten de volgende regels in acht genomen worden

- Tijdens het kwantisatieproces moeten schaalfactor-, fase- en amplitudevrijheid buiten het object gewaarborgd worden. Amplitude-vrijheid op het object levert alleen maar een trager proces op.

- Het kwantiseren moet geleidelijk gebeuren. Zodra het gehele hologram gekwantiseerd wordt stagneert het proces direct.

- De stepwise begrenzer met stijgende grenswaarde levert de beste prestaties. Deze prestaties zijn duidelijk hoger dan met een niet iteratieve methode.

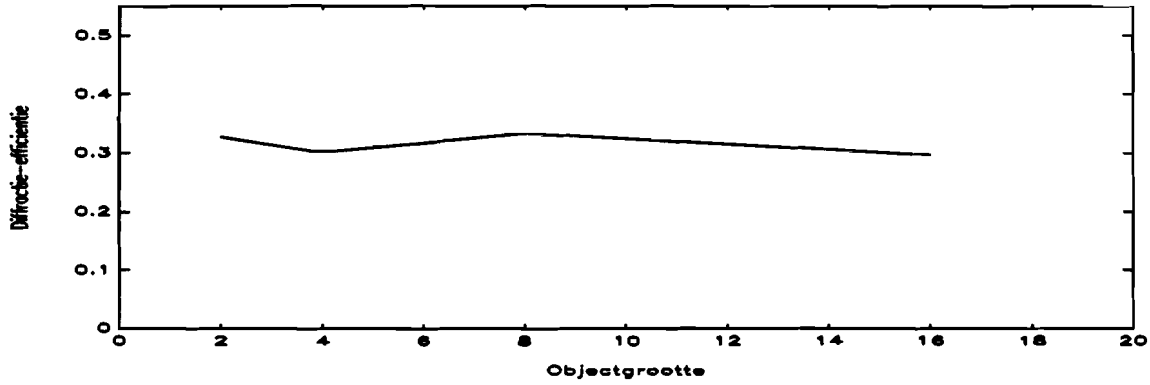
Tijdens het onderzoek naar het initiële hologram is er geconcludeerd dan de maximaal haalbare signaal-ruis-verhouding afhankelijk is van de verhouding van de object-grootte tot de reconstructie-grootte. Op dit moment is het echter mogelijk om ook een orde-grootte van deze maximaal haalbare signaal-ruis-verhouding te geven. In de volgende figuur is voor verschillende object-afmetingen de maximaal haalbare signaal-ruis-verhouding weergegeven. Dit zijn dezelfde objecten als voorheen (fig. 3.10).



Figuur 5.25 : Maximaal haalbare signaal-ruis-verhouding als functie van de object-afmetingen voor een 64x64 grote reconstructie.

Te zien is dat voor zeer kleine objecten zeer grootte signaal-ruis-verhoudingen te realiseren

zijn. De diffractie-efficiëntie bij deze maximale signaal-ruis-verhouding is nagenoeg onafhankelijk voor deze objecten. Dit is in de volgende figuur, waar de diffractie-efficiëntie voor de verschillende object-afmetingen weergegeven is, goed te zien.



Figuur 5.26 : Diffractie-efficiëntie voor verschillende object-afmetingen in een 64x64 grote reconstructie.

6. Simulated Annealing

Tijdens het kwantiseren met een kwantisator met stijgende grenswaarde, paragraaf 5.2.2, bleek dat al vrij vroeg in het kwantisatie-proces de uiteindelijk oplossing nagenoeg bekend is. Dit doet vermoeden dat het zelfs mogelijk is dat er al heel vroeg in het totale proces bepaalde pixels hun uiteindelijke waarde bereiken. Dit zou betekenen dat het proces al vroeg een bepaalde richting in gaat en in deze richting de beste lokale oplossing berekent.

Om te zorgen dat er meerdere richtingen door het proces bekeken worden wordt er een soort simulated annealing ingevoerd. Simulated annealing betekent dat er met een bepaalde kans een foute beslissing wordt genomen om zo meerdere oplossingen te bekijken. Deze kans wordt gekoppeld aan een procestemperatuur die daalt naarmate het proces langer gaat duren. Hierdoor is het altijd zeker dat er op den duur een stabiele toestand op zal treden.

In de in dit verslag gebruikte iteratieve methodes worden er echter geen beslissingen genomen waarvan de grootte van de fout bekend is. Het is dan ook niet mogelijk om bewust een foute beslissing te nemen. Om toch een soort simulated annealing in het proces in te bouwen gaan we ervan uit dat een hologram-pixel dat op een bepaald moment een positieve reëele waarde heeft uiteindelijk 1 zal worden. Het zal dan een foute beslissing zijn om dit pixel een negatieve waarde te geven. De grootte van de beslissingsfout wordt dan bepaald door de amplitude van het pixel.

Dit komt er op neer dat tijdens het iteratief verhogen van de transparantie-efficiëntie en tijdens het kwantisatieproces, nadat het object in de reconstructie teruggezet is en de bijbehorende pixelverdeling bepaald is, de pixels met een kans gelijk aan

$$P_{pixel}=e^{-\frac{Amplitude}{Procestemperatuur}} \tag{6.1}$$

in de oorsprong gespiegeld worden.

Deze kans wordt kleiner

- als de amplitude van het pixel groter wordt.
- naarmate het proces verder vordert en de procestemperatuur daalt.

De procestemperatuur wordt iedere iteratie verlaagd zodat de kans dat pixels met een amplitude gelijk aan 1 gespiegeld worden gehalveerd wordt.

Na iedere iteratie geldt dan voor de nieuwe procestemperatuur

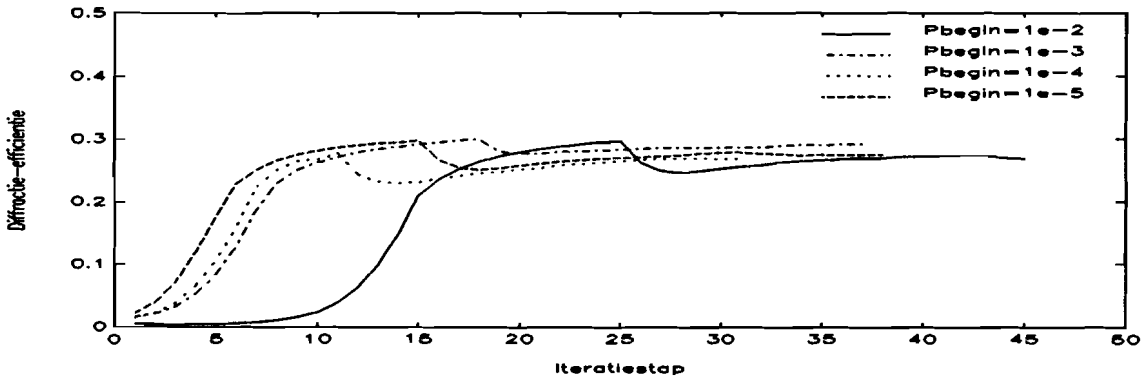
$$Tp_{nieuw} = \frac{Tp_{oud}}{1+Ln(2)Tp_{oud}} \tag{6.2}$$

waarin Tp de procestemperatuur voorstelt.

De kans waarmee pixels, met een amplitude gelijk aan 1, aan het begin van het proces gespiegeld worden kan aan het programma meegegeven worden. Voor de procestemperatuur aan het begin van het proces geldt dan

$$Tp_{begin} = \frac{-1}{Ln(P_{begin})}. \tag{6.3}$$

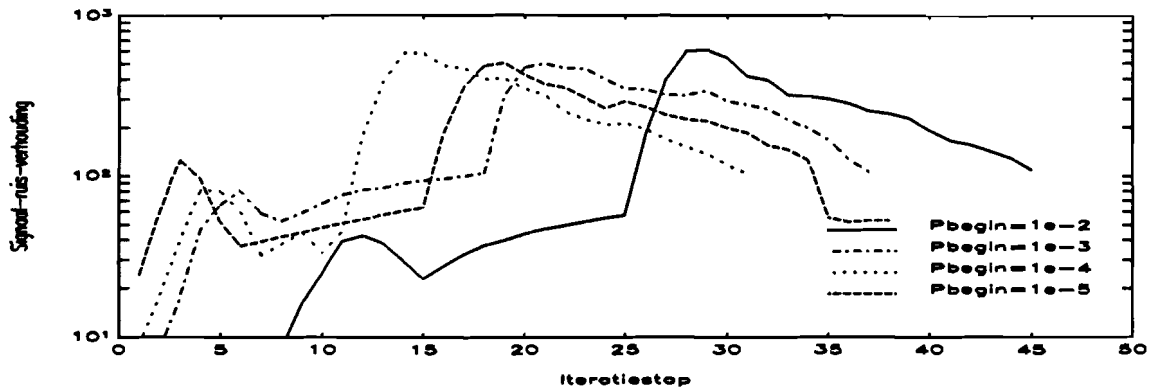
Met verschillende beginkansen zijn de berekeningen uitgevoerd en in de volgende figuur is het verloop van de diffractie-efficiëntie weergegeven.



Figuur 6.1 : Verloop van de diffractie-efficiëntie voor het gebruik van simulated annealing met verschillende beginkansen.

In deze figuur is te zien dat het voor de diffractie-efficiëntie niet veel uitmaakt hoe groot de beginkans is. Het proces is alleen minder stabiel als de beginkans groter is en duurt dan veel langer.

In de volgende figuur is het verloop van de signaal-ruis-verhouding weergegeven.



Figuur 6.2 : Verloop van de signaal-ruis-verhouding bij het gebruik van simulated annealing met verschillende beginkansen.

De uiteindelijke signaal-ruis-verhouding kan wel beter worden met het gebruik van simulated annealing. Het is echter niet te garanderen met welke beginkans het beste resultaat behaald wordt omdat het verloop van het algoritme min of meer random wordt.

Het is het beste om een hologram een paar keer te berekenen met verschillende beginkansen en uit de verschillende eindresultaten het beste hologram te kiezen.

7. Uitwerking van een 2-dimensionale Fast-Fourier-transformatie.

Het totale algoritme is zeer rekenintensief. Het grootste deel van de rekentijd wordt ingenomen door de Fourier-transformatie. Om het algoritme zover mogelijk te optimaliseren wordt hier een 2-dimensionale Fast-Fourier-transformatie [14] uitgewerkt die een duidelijke snelheidsverbetering oplevert.

Voor de 2-dimensionale-Fourier-transformatie geldt

$$G(u,v)=\sum_{k=0}^{N-1}\sum_{l=0}^{N-1} e^{-j\frac{2\pi}{N}(ku+lv)}=\sum_{k=0}^{N-1}\sum_{l=0}^{N-1} W_N^{ku+lv}. \tag{7.1}$$

Met

$$W_N=e^{-j\frac{2\pi}{N}}.$$

Als we deze sommatie op splitsen over combinaties van even en oneven matrix-coördinaten dan gaat deze over in

$$G(u,v)=\sum_{k=0}^{\frac{N}{2}-1}\sum_{l=0}^{\frac{N}{2}-1} g(2k,2l)W_N^{2ku+2lv}+\sum_{k=0}^{\frac{N}{2}-1}\sum_{l=0}^{\frac{N}{2}-1} g(2k+1,2l+1)W_N^{2ku+2lv+u+v}+$$

$$+\sum_{k=0}^{\frac{N}{2}-1}\sum_{l=0}^{\frac{N}{2}-1} g(2k+1,2l)W_N^{2ku+2lv+u}+\sum_{k=0}^{\frac{N}{2}-1}\sum_{l=0}^{\frac{N}{2}-1} g(2k,2l+1)W_N^{2ku+2lv+v}=$$

$$=\sum_{k=0}^{\frac{N}{2}-1}\sum_{l=0}^{\frac{N}{2}-1} g(2k,2l)(W_N^2)^{ku+lv}+\sum_{k=0}^{\frac{N}{2}-1}\sum_{l=0}^{\frac{N}{2}-1} g(2k+1,2l+1)(W_N^2)^{ku+lv}W_N^{u+v}+$$

$$\sum_{k=0}^{\frac{N}{2}-1}\sum_{l=0}^{\frac{N}{2}-1} g(2k+1,l)(W_N^2)^{ku+lv}W_N^{u}+\sum_{k=0}^{\frac{N}{2}-1}\sum_{l=0}^{\frac{N}{2}-1} g(2k,2l+1)(W_N^2)^{ku+lv}W_N^{v}. \tag{7.3}$$

Omdat geldt

$$W_N^2=(e^{-j\frac{2\pi}{N}})^2=e^{-j\frac{2\pi}{N/2}}=W_{N/2} \tag{7.4}$$

is (7.3) een combinatie van vier *N/2-punts* 2-dimensionale Fourier-transformaties. Voor deze combinatie geldt dan

$$G(u,v)=A(u,v)+B(u,v)W_N^{u+v}+C(u,v)W_N^{u}+D(u,v)W_N^{v}. \tag{7.5}$$

Waarin A, B, C en D $N/2$-punts 2-dimensionale Fourier-transformaties zijn. Deze $N/2$-punts 2-dimensionale Fourier-transformaties hebben afmetingen $N/2xN/2$ en zijn periodiek met periode $N/2$ in beide dimensies. Hierdoor is het mogelijk om zodra A, B, C en D berekend zijn de Fourier-transformatie met de volgende vergelijkingen te berekenen.

$$G(u,v)=A(u,v)+B(u,v)W_N^{u+v}+C(u,v)W_N^{u}+D(u,v)W_N^{v}. \tag{7.6}$$

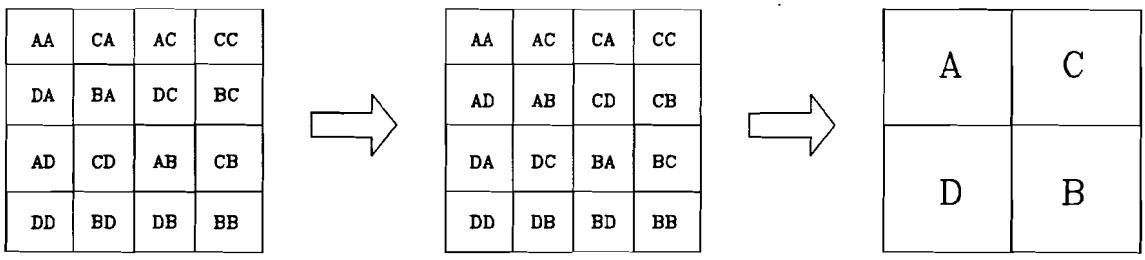
$$G(u+\frac{N}{2},v)=A(u,v)-B(u,v)W_N^{u+v}-C(u,v)W_N^{u}+D(u,v)W_N^{v}. \tag{7.7}$$

$$G(u,v+\frac{N}{2})=A(u,v)-B(u,v)W_N^{u+v}+C(u,v)W_N^{u}-D(u,v)W_N^{v}. \tag{7.8}$$

$$G(u+\frac{N}{2},v+\frac{N}{2})=A(u,v)+B(u,v)W_N^{u+v}-C(u,v)W_N^{u}-D(u,v)W_N^{v}. \tag{7.9}$$

Dit is de butterfly voor een 2-dimensionale Fast-Fourier-transformatie.

Bij de berekening van A, B, C en D is het echter mogelijk om weer een opsplitsing over de even en oneven coördinaten-combinaties te maken. Dit kan net zolang totdat er een 2x2-matrix overblijft waarvan de Fourier-getransformeerde direct met de butterfly berekend kan worden. In de volgende figuur is dit grafisch weergegeven.



Figuur 7.1 : *Grafische weergave van de 2-dimensionale Fast-Fourier-transformatie.*

De letters in deze figuur komen overeen met de verschillende delen van de Fourier-getransformeerde waarbij de $g(k,l)$ die op deze coördinaat staat bij betrokken dient te worden. Zo betekent AB dat de $g(k,l)$ die op deze coördinaat staat. Bij het berekenen van $A(u,v)$ voor de tweede iteratieslag als $B(u,v)$ in de butterfly geplaatst moet worden.

In de figuur bestaat de eerste transformatie uit een bit-reversal. Dit betekend dat van zowel de k-coördinaat als van de l-coördinaat de binaire representatie gespiegeld wordt. Hierdoor komen de $g(k,l)$ waarvan zowel k als l het vaakst door 2 deelbaar is in de linkerbovenhoek van de matrix terecht. Deze $g(k,l)$ zal bij elke opsplitsing bij de sommatie over even k's en l's terechtkomen. De $g(k,l)$ waarvan de coördinaten nooit door twee deelbaar zijn komen in de rechter-onderhoek van de matrix terecht. Deze $g(k,l)$ zal bij elke opsplitsing bij de sommatie over oneven k's en l's terechtkomen.

De derde matrix uit de figuur geeft de matrix na de eerste iteratie-slag weer. De kleinde 2x2 matrices zijn Fourier-getransformeerd met behulp van de butterfly en leveren nu de A, B, C en D op voor de volgende iteratie.

Om te bekijken hoeveel sneller deze FFT2-methode is zullen we kijken naar het aantal vermenigvuldigigen die nodig zijn voor de volgende combinaties

- De directe methode, voor elke $G(u,v)$ wordt de sommatie (7.1) berekend. Dit betekend voor elke $G(u,v)$ N^2 vermenigvuldigingen. Om $G(u,v)$ in zijn totaal te berekenen betekend dit dus N^4 vermenigvuldigingen.

- De Fourier-transformatie kan ook opgesplitst worden volgens

$$G(u,v)=\sum_{k=0}^{N-1}\sum_{l=0}^{N-1} g(k,l)e^{-j\frac{2\pi}{N}(ku+lv)}=\sum_{k=0}^{N-1} e^{-j\frac{2\pi}{N}(ku)}\sum_{l=0}^{N-1} g(k,l)e^{-j\frac{2\pi}{N}(lv)}. \qquad (7.10)$$

  Dit betekent dat de 2-dimensionale Fourier-transformatie uitgevoerd kan worden door eerst de kolommen van de matrix 1-dimensionaal te Fourier-transformeren en daarna de rijen nogmaals 1-dimensionaal te transformeren. Volgens de literatuur [14] kost een 1-dimensionale Fourier-transformatie $N/2*log_2(N)$ vermenigvuldigingen. Deze 1-dimensionale Fourier-transformatie moet $2N$ keer uitgevoerd worden, dus volgens deze methode kost de totale Fourier-transformatie $N^2*log_2(N)$ vermenigvuldigingen.

- De 2-dimensionale Fast-Fourier-transformatie. Bij iedere opslitsing kost de berekening van de butterfly 3 vermenigvuldigingen. Deze levert de Fourier-getransformeerde van 4 punten en moet dus $N^2/4$ keer berekend worden. Dit kost per opsplitsing $3N^2/4$ vermenigvuldigingen. De matrix kan in totaal $log_2(N)$ keer opgesplitst worden. Dit komt neer op een totaal van $3/4\ N^2\ log_2(N)$ vermenigvuldigingen.

Ter illustratie vergelijken we het aantal vermenigvuldigingen die voor deze drie berekeningsmethodieken nodig zijn voor *N=8*, *N=64* en *N=128*.

	FFT2	**2N*FFT1**	**Direkte methode**
N=8	144	192	4096
N=64	18432	24576	16777216
N=128	86016	114688	268435456

De besparing van de hier beschreven methode ten opzichte van de direkte methode is natuurlijk veel groter dan ten opzichte van de *2N*FFT1* methode. Hierbij moet wel opgemerkt worden dat de *FFT2* qua implementatie veel meer mogelijkheden heeft dan de *2N*FFT1*. Voor het behalen van een behoorlijke snelheidswinst met deze methode is het wel noodzakelijk dat de matrix waarop de twee-dimensionale *FFT* uitgevoerd wordt volledig in het geheugen aanwezig is. Als er delen van deze matrix op de disk bewaard moeten worden, bijvoorbeeld wegens geheugengebrek van het gebruikte systeem, dan is deze methode veel trager dan *2N*FFT1*.

8. Conclusies en aanbevelingen

In dit verslag is een totaal-algoritme ontwikkeld voor de berekening van binaire amplitude hologrammen. Dit totaal-algoritme, bestaande uit drie delen, is m.b.v. TURBO-C geïmplementeerd en geoptimaliseerd.

Voor het tweede en derde deel van het totaal-algoritme is gekozen om het Gerchberg-Saxton algoritme aan te passen. Deze aanpassing heeft geleid tot een algoritme wat bestaat uit zowel lokale operaties in het hologramvlak als in het reconstructievlak. Deze lokale operaties in het reconstructievlak komen hier overeen met niet-causale operaties in het hologramvlak. Dit is de reden waarom dit algoritme veel krachtiger is dan een algoritme, zoals bijvoorbeeld error-diffusie, dat bestaat uit alleen causale operaties in het hologramvlak.

Om deze lokale operaties uit te kunnen voeren is het echter wel noodzakelijk om tussen het hologramvlak en het reconstructievlak op en neer te switchen. Dit betekend dat het algoritme zeer veel rekentijd opeist. Om deze rekentijd binnen acceptabele grenzen te houden is het noodzakelijk dat de matrix waarin de ene keer de reconstructie en de andere keer het hologram opgeslagen zit volledig in het geheugen aanwezig is. Omdat TURBO-C niet meer dan 1 Megabyte geheugen kan addresseren wordt hiermee de matrix-grootte tot 128x128 beperkt. Het is daarom ook aan te bevelen om het programma op een groter systeem te implementeren zodat grotere hologrammen berekend kunnen worden.

De transformatie bestaat uit een tweedimensionale Fourier-transformatie en zijn inverse. Om het programma qua snelheid nog te verbeteren is in dit verslag een 2-dimensionale Fast-Fourier-Transformatie ontwikkeld. Deze levert een aanzienlijke snelheidsverbetering van het totale programma op.

Het binaire amplitude hologram wat door het programma berekent heeft een zeer goede diffractie-efficiëntie en er zijn hoge signaal-ruis-verhoudingen te realiseren.

Literatuurlijst

[1] Just, D. and Ling, D.T. — Neural Networks for binarizing computer-generated holograms. Opt. Commun., Vol 81(1991), No 1-2, p. 1-5.

[2] Wyrowski, F. — Diffraction efficiency of analog and quantized digital amplitude holograms; analysis and manipulation. J. Opt. Soc. Am., Vol 7(1990), No. 3, p. 383-393.

[3] Wyrowski, F. — Iterative quantization of digital amplitude holograms Appl. Opt., Vol 28(1989), No. 18, p. 3864-3870.

[4] Weissbach, S. et al. — Digital phase holograms: coding and quantization with an error diffusion concept. Opt. Commun., Vol 72(1989) No. 1-2, p. 37-41.

[5] Jennison, B.K. and Allebach J.P. — Iterative approaches to computer-generated holography. Opt. Eng., Vol 28(1989), No. 6, p. 629-637.

[6] Wyrowski, F. and Bryngdahl, O. — Iterative Fourier-transform algorithm applied to computer holography. J. Opt. Soc. Am., Vol 5(1988), No. 7, p. 1058-1065.

[7] Wyrowski, F. et al. — Computer-generated holography: hologram repetition and phase manipulations. J. Opt. Soc. Am., Vol 4(1987), No. 4, p. 694-698.

[8] Sasaki, O. and Yamagami, T. Phase-retrieval algorithms for nonnegative and finite-extent objects.
J. Opt. Soc. Am., Vol 4(1987), No.4, p. 720-726.

[9] Wyrowski, F. et al. Computer Holography: Object dependent determistic diffusers.
Opt. Commun., Vol 63(1987), No. 2, p. 81-84,

[10] Fienup, J.R. Phase Retrieval algorithms: a comparison.
Appl. Opt., Vol 21(1982), No. 15, p. 2758-2769.

[11] Fienup, J.R. Iterative method applied to image reconstruction and to computer-generated holograms.
Opt. Eng., Vol. 19(1980), No. 3, p. 297-305.

[12] Butterweck, H.J. Grondslagen van de moderne optica.
Herdruk november 1990. Eindhoven : Technische Universiteit Eindhoven. Dictaatnr. 5010.

[13] Jenniskens, A.J.W. Error diffusie toegepast als quantisatiemethode voor computer gegenereerde hologrammen.
Eindhoven : Technische Universiteit Eindhoven, Vakgroep Theoretische Elektrotechniek, 1991.

[14] Lim, J.S. Two-dimensional signal and image processing.