HUPP 1

David Tonderski - davton

1 Uppgift 1

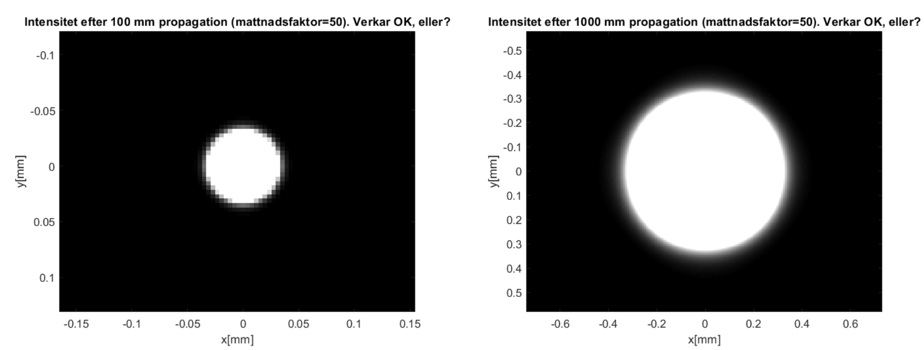
MATLAB-koden bifogar jag i appendix.

2 Uppgift 3

I uppgifterna används i approximationsandan oftast en värdesiffra, då jag har fått känslan att det mest är storleksordningar och simulationsfärdigheter som är intressanta, medan exakta värden ändå inte är exakta i verkliga sammanhang.

För $f = 10\,\text{cm}$ blir $D_{spot} \approx 0.04\,\text{mm}$ (figur 1a). Det teoretiska värdet är $\approx 0.03\,\text{mm}$. Rätt storleksordning, tumregeln stämmer bra! Det (mer) exakta värdet för konstanten är $C = \frac{D_{spot} \cdot 2\omega}{\lambda \cdot L} \approx 1.3$, vilket tydligen är av storleksordning 1.

För $f = 1\,\text{m}$ är $D_{spot} \approx 0.4\,\text{mm}$ (figur 1b). Det teoretiska värdet är $\approx 0.3\,\text{mm}$. Ännu en gång visar tumregeln sin användbarhet! Konstanten blir återigen $C \approx 1.26$, då vi ju multiplicerar och dividerar den förra konstanten med 10.



(a) Gaussiskt fjärrfält med $f = 10\,\text{cm}$.

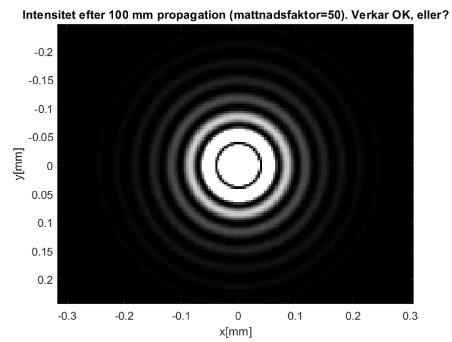
(b) Gaussiskt fjärrfält med $f = 1\,\text{m}$.

Figure 1: Fjärrfälten för de Gaussiska fälten.

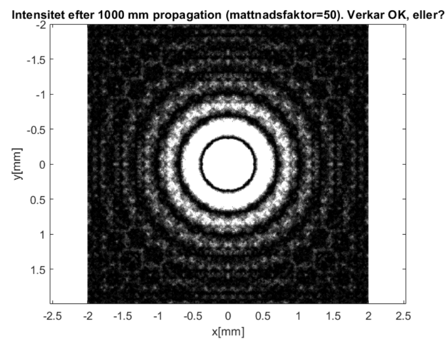
3 Uppgift 4

Nu till den cirkulära strålen. För $f = 10\,\text{cm}$ fås $D_{spot} \approx 0.08\,\text{mm}$, alltså dubbelt så stor som för det Gaussiska fältet! Som tur var är det fortfarande rätt storleksordning. Konstanen antar nu värdet $C \approx 2.53$.

För $f = 1\,\text{m}$ ses (nu börjar jag få slut på synonymer) $D_{spot} \approx 0.8\,\text{mm}$ enligt förväntan. Återigen blir konstanten samma som för det förra fältet, $C \approx 2.53$.



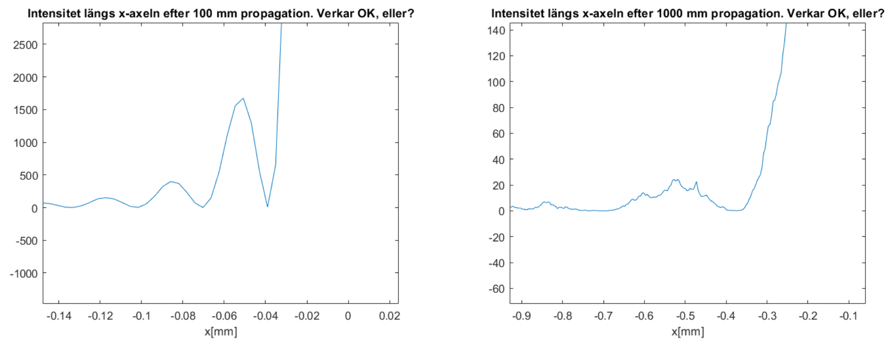
(a) Cirkulärt fjärrfält med $f = 10\,\text{cm}$.



(b) Cirkulärt fjärrfält med $f = 1\,\text{m}$.

Figure 2: Fjärrfälten för de cirkulära fälten.

Numeriska fel börjar bli en faktor i $f = 1\,\text{m}$. Detta ses i figur 2b, där det snygga (!) Airy-mönstret är ganska brusigt. Det är också tydligt när man tittar på intensiteten i x-led för fjärrfältet. Intensiteten är brusig och omonoton i figur 3b, jämför med den fina fördelningen i figur 3a. Fenomenet uppstår eftersom en del av fältet når ut till kanten av det numeriska fönstret och då skickas tillbaka in. Detta vet jag bara på grund av att jag gjorde uppgiften förra året (fast inte lika fint!).



(a) Intensitet i fjärrfältet för $f = 10\,\mathrm{cm}$. (b) Intensitet i fjärrfältet för $f = 1\,\mathrm{m}$.

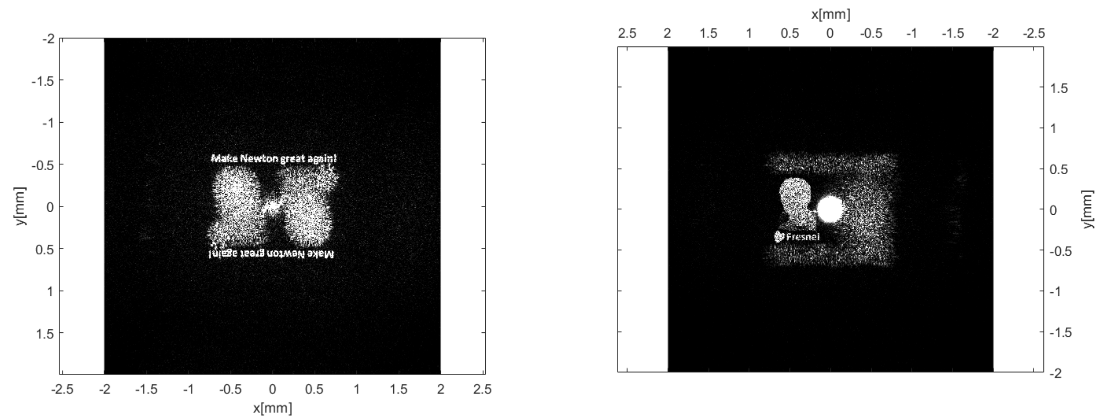
Figure 3: Intensiteten x-led ($y = 0$) för fjärrfälten för de cirkulära fälten.

4 Uppgift 6

Nu till den roligaste delen! Först antaganden; vi sätter $n = 1$ för luft, antar att den infallande vågen är plan, samt att avståndet mellan ögönlinsen och väggen är mycket större än avståndet mellan ögonlinsen och näthinnan, så $f_{öga} = 20\,\mathrm{mm}$.

Utan De Vises Lins fås med en måttnadsfaktor på 10000 det ofarliga meddelandet "Make Newton great again!". Det riktiga meddelandet blir synligt när man lägger på en lins med fokallängd $0.143\,\mathrm{m}$ (alltså 7 dioptrier som jag lärde mig under förra årets HUPP1!) och ändrar måttnadsfaktorn till 15 (blir inte detta ett problem? Intuitivt känns det som att måttnadsfaktorerna borde vara av samma storleksordning för de båda meddelandena för att det ska vara möjligt att läsa av båda). Det säger "♡ Fresnel!" och visar en väldigt fin bild på ett (tyvärr för mig oidentifierbart) ansikte. Meddelandena visas i figur 4. MATLAB-koden bifogas i appendix.

Avslutningsvis har jag en till fråga - hur kommer det sig att det ofarliga meddelandet ser likadant ut när man vänder det upp-och-ner, men inte det hemliga?



(a) Det ofarliga meddelandet.

(b) Det riktiga meddelandet med De Vises Lins.

Figure 4: Meddelandena kodade i DOEn.

A MATLAB-kod

A.1 HUPP1.m

```
clear
close all
%full_white_value=64; % äldre matlabversion - detta värde
     plottas som vitt (max) med image-kommandot
full_white_value=255; % nyare matlabversion - prova denna
     om din plot verkar mörk!

N=1024; % NxN är matrisstorleken (rekommenderad storlek N
    =1024)
sidlaengd_Plan1=4e-3; % det samplade områdets storlek (i
    x- eller y-led) i Plan 1 (rekommenderad storlek 4 mm)
a=sidlaengd_Plan1/N; % samplingsavstånd i Plan 1 (och
    Plan 2 eftersom vi använder PAS)
L=1000e-3; % propagationssträcka (dvs avstånd mellan Plan
     1 och 2)

lambda_noll=633e-9; % vakuumvåglängd för rött ljus från
    en HeNe-laser
n_medium=1; % brytningsindex för medium mellan Plan 1 och
     2
k = 2*pi*n_medium/lambda_noll; % k-vektorns längd
```

```
xvekt=-N/2*a:a:(N/2-1)*a; % vektor med sampelpositioner i
     x-led
yvekt=xvekt; % och y-led
[xmat,ymat]=meshgrid(xvekt,yvekt); % koordinatmatriser
    med x- och y-värdet i varje sampelposition
rmat=sqrt(xmat.^2+ymat.^2); % avståndet till origo i
    varje sampelpunkt. Observera att alla operationer är
    elementvisa!

%******* Fält i Plan 1
f_lins=1000e-3; % fokallängd på linsen före Plan 1
T_lins=exp(-1i*k*rmat.^2/(2*f_lins)); %
    Transmissionsfunktion för en lins (linsen är TOK)

D_apertur=2e-3;
T_apertur=rmat<(D_apertur/2); % Transmissionsfunktion för
     en cirkulär apertur ("pupill")

omega_in=1e-3; % 1/e2-radie (för intensiteten, dvs 1/e-
    radie för amplituden) för infallande Gaussiskt fält
E_in_gauss=exp(-rmat.^2/omega_in^2); % Infallande fält:
    Gaussiskt med plana vågfronter och normalinfall (dvs
    konstant fas, här=0)

E_in_konstant=ones(N,N); % Infallande fält: Plan våg med
    normalt infall

E1_gauss=E_in_gauss.*T_lins; % Fältet i Plan 1 (precis
    efter linsen) för gaussisk stråle
E1_cirkular=E_in_konstant.* T_lins .* T_apertur; % Fältet
     i Plan 1 (precis efter linsen) för konstant fält som
    passerat genom cirkulär apertur
E1=E1_gauss; % Välj fall!

I1=abs(E1).^2; % intensiteten är prop mot kvadraten på fä
    ltets amplitud (normalt struntar man i
    proportionalitetskonstanten)

figure(1)
image(xvekt*1e3,yvekt*1e3,I1/max(max(I1))*
    full_white_value)
title(['Intensitet i Plan 1. Verkar OK, eller?'])
xlabel('x [mm]')
ylabel('y [mm]')
colormap(gray)
drawnow
```

```
axis('equal')

figure(2)
imagesc(xvekt*1e3,yvekt*1e3,angle(E1))
title(['Fas i Plan 1. Verkar OK, eller?'])
xlabel('x[mm]')
ylabel('y[mm]')
colormap(gray)
colorbar
drawnow
axis('equal')


pause % tryck på valfri tangent för att fortsätta

%**** Och nu propagerar vi till Plan 2!
E2=PAS(E1,L,N,a,lambda_noll,n_medium); % Propagation med
    PAS-funktionen

I2=abs(E2).^2;

mattnadsfaktor_plot=50; % anger hur många gånger maxvä
    rdet ska vara mättat i plotten (>1, kan vara bra om
    man vill se svagare detaljer)
figure(3)
image(xvekt*1e3,yvekt*1e3,I2/max(max(I2))*
    full_white_value*mattnadsfaktor_plot)
title(['Intensitet efter ' num2str(L*1e3) ' mm
    propagation (mattnadsfaktor=' num2str(
    mattnadsfaktor_plot) '). Verkar OK, eller?'])
xlabel('x[mm]')
ylabel('y[mm]')
colormap(gray)
drawnow
axis('equal')

figure(4)
plot(xvekt*1e3,I2(N/2+1,:))
title(['Intensitet längs x-axeln efter ' num2str(L*1e3) '
    mm propagation. Verkar OK, eller?'])
xlabel('x[mm]')
drawnow
```

A.2 PAS.m

```
function E2=PAS(E1,L,N,a,lambda_noll,n_medium)
```

```
% Varje sampelpunkt i k-planet motsvarar en plan våg med
    en viss riktning (kx,ky,kz)
delta_k=2*pi/(N*a); % samplingsavstånd i k-planet
kxvekt=-N/2*delta_k:delta_k:(N/2-1)*delta_k; % vektor med
     sampelpositioner i kx-led
kyvekt=kxvekt; % och ky-led
[kxmat,kymat]=meshgrid(kxvekt,kyvekt); % k-vektorns x-
    resp y-komponent i varje sampelpunkt i k-planet

k=2*pi*n_medium/lambda_noll; % k-vektorns längd (skalär)
    för en plan våg i ett material med brytningsindex
    n_medium *** Ej klar
kzmat=sqrt(k^2 - kxmat.^2 - kymat.^2); % k-vektorns z-
    komponent i varje sampelpunkt i k-planet *** Ej klar (
    Obs! Matlab tillåter att en skalär direkt adderas/
    subtraheras med matris, man behöver alltså tex inte
    skriva "skalär*ones(N,N)-matris")

fasfaktor_propagation=exp(1i*kzmat*L); % faktorn varje
    sampelpunkt i k-planet (som ju motsvarar plan våg i
    viss riktning) multas med för att propagera sträckan L
     i z-led

A= a^2/(2*pi)*fft2c(E1); % Planvågsspektrum i Plan 1

B=A.*fasfaktor_propagation; % Planvågsspektrum i Plan 2 (
    Planvågsspektrum i Plan 1 multat med fasfaktorn för
    propagation för varje plan våg)

E2= delta_k^2*N^2*ifft2c(B);
```

A.3 Uppgift6.m

```
clear
close all
%full_white_value=64; % äldre matlabversion - detta värde
     plottas som vitt (max) med image-kommandot
full_white_value=255; % nyare matlabversion - prova denna
     om din plot verkar mörk!

N=1024; % NxN är matrisstorleken (rekommenderad storlek N
    =1024)
sidlaengd_Plan1=4e-3; % det samplade områdets storlek (i
    x- eller y-led) i Plan 1 (rekommenderad storlek 4 mm)
a=sidlaengd_Plan1/N; % samplingsavstånd i Plan 1 (och
```

```
    Plan 2 eftersom vi använder PAS)
L=20e-3; % propagationssträcka (dvs avstånd mellan Plan 1
     och 2)

lambda_noll=633e-9; % vakuumvåglängd för rött ljus från
    en HeNe-laser
n_medium=1; % brytningsindex för medium mellan Plan 1 och
     2
k = 2*pi*n_medium/lambda_noll; % k-vektorns längd

xvekt=-N/2*a:a:(N/2-1)*a; % vektor med sampelpositioner i
     x-led
yvekt=xvekt; % och y-led
[xmat,ymat]=meshgrid(xvekt,yvekt); % koordinatmatriser
    med x- och y-värdet i varje sampelposition
rmat=sqrt(xmat.^2+ymat.^2); % avståndet till origo i
    varje sampelpunkt. Observera att alla operationer är
    elementvisa!
load T_DOE_gen2; % laddar in transmissionsfunktionen för
    DOEn
%% UTAN DE VISES LINS
clf,clc
f_lins=20e-3; % fokallängd på linsen i -ogat
T_lins=exp(-1i*k*rmat.^2/(2*f_lins)); %
    transmissionsfunktion för en lins (linsen -ar TOK)
E_in=ones(N,N); % infallande fält: Plan våg med normalt
    infall
E_ut=E_in.*T_lins.*T_DOE_gen2;
E2=PAS(E_ut,L,N,a,lambda_noll,n_medium); % propagation
    med PAS-funktionen
I2=abs(E2).^2; % Intensiteten

mattnadsfaktor_plot=10000; % anger hur många gånger maxvä
    rdet ska vara mättat i plotten (>1, kan vara bra om
    man vill se svagare detaljer)
figure(3)
image(xvekt*1e3,yvekt*1e3,I2/max(max(I2))*
    full_white_value*mattnadsfaktor_plot)
xlabel('x[mm]')
ylabel('y[mm]')
colormap(gray)
drawnow
axis('equal')
%% MED DE VISES LINS
clf, clc
f_lins=20e-3; % fokallängd på linsen i -ogat
```

```
f_lins_v=1.43e-1; % fokallängd på de vises lins
T_lins=exp(-1i*k*rmat.^2/(2*f_lins)); %
    Transmissionsfunktion för en lins (linsen -ar TOK)
T_lins_v=exp(-1i*k*rmat.^2/(2*f_lins_v));
E_in=ones(N,N); % Infallande f-alt: Plan våg med normalt
    infall
E_ut=E_in.*T_lins.*T_DOE_gen2.*T_lins_v;
E2=PAS(E_ut,L,N,a,lambda_noll,n_medium); % Propagation
    med PAS-funktionen
I2=abs(E2).^2; % Intensiteten

mattnadsfaktor_plot=15; % anger hur många gånger maxvä
    rdet ska vara mättat i plotten (>1, kan vara bra om
    man vill se svagare detaljer)
figure(4)
image(xvekt*1e3,yvekt*1e3,I2/max(max(I2))*
    full_white_value*mattnadsfaktor_plot)
xlabel('x[mm]')
ylabel('y[mm]')
colormap(gray)
drawnow
camroll(180)
axis('equal')
```