Diffraction and Fourier optics

[Version 9.5.2019]

Nanophotonics and Metrology Laboratory
Institute of Microengineering,
School of Engineering,
École Polytechnique Fédéderale de Lausanne,
EPFL-STI-IMT-NAM
ELG 237
Station 11
CH-1015 Lausanne

Web : http://nam.epfl.ch

Contact person: Dr. Toralf SCHARF,
Phone: +41 21 6954286
E-mail : toralf.scharf@epfl.ch

Contents

1 Objective and overview

The practical work should introduce the following subjects to you:

- Visualize diffraction for different single apertures (pinhole, slit, triangle...)
- Simulate diffraction problems with MATLAB
- Compare the results with diffraction gratings of various parameters

Please read the reference documents provided.

2 Safety Issues

In this experiments laser sources and low power electrical equipment are used. The laser is of class II. Class II: low-power visible lasers that emit a radiant power below 1 mW. The concept is that the human aversion reaction to bright light will protect a person. Only limited controls are specified. (http://www.osha.gov, Laser Hazards)
The laser is safe because the blink reflex of the eye will limit the exposure to less than 0.25 seconds. It only applies to visible-light lasers (400–700 nm). Intentional suppression of the blink reflex could lead to eye injury. In our experiment the laser sources are collimated and should be handled with care. A strongly divergent beam will not be focussed on the eyes retina and represents often no danger. Collimated beams will lead to small focus spots onto the retina special care is needed. ***Do not stare into a collimated beam!***
The electrical equipment used in the experiments is based on USB power (5V, 0.5 A, 2.5 W) and not subjected to any particular security issues. Nevertheless you should **not produce short circuits** on the printed circuit board (PCB) or to the computers USB connection to avoid damage to the material. Make proper use of screwdrivers. Do not force any mechanical parts.

3 Background

3.1. Scalar Diffraction

The text below is a modified copy of the book from **David G. Voelz, Computational Fourier optics : a MATLAB tutorial, SPIE Bellingham, 2011**.

Diffraction refers to the behavior of an optical wave when its lateral extent is confined; for example, by an aperture. It accounts for the fact that light rays do not follow strictly rectilinear paths when the wave is disturbed on its boundaries. In our everyday experience we rarely notice diffractive effects of light. The effects of reflection (from a mirror), or refraction (due to a lens) are much more obvious. In fact, the effects of diffraction become most apparent when the confinement size is on the order of the wavelength of the radiation. Nevertheless, diffraction plays a role in many optical applications and it is a critical consideration for applications involving high resolution, such as astronomical imaging, or long propagation distances such as laser radar, and in applications involving small structures such as photolithographic processes. The propagation behavior of an optical wave is fundamentally governed by Maxwell's equations. In general, coupling exists between the wave's electric field E with components (Ex, Ey, Ez) and its magnetic field H with components (Hx, Hy, Hz). There is also coupling between the individual components of the electric field, as well as between the magnetic components. However, consider a wave that is propagating in a dielectric medium that is linear (field quantities from separate sources can be summed), isotroptic (independent of the wave polarization, i.e., the directions of E and H), homogeneous (permittivity of the medium is independent of position), nondispersive (permittivity is independent of wavelength), and nonmagnetic (magnetic permeability is equal to the vacuum permeability). In this case, Maxwell's vector expressions become decoupled, and the behavior of each component of the electric or magnetic fields can be expressed independently from the other components. Scalar diffraction refers to the propagation behavior of light under this ideal situation. The long list of assumptions for the medium suggests a rather limited application regime for scalar diffraction theory. However, scalar diffraction can clearly be used for describing free-space optical (FSO) propagation, which refers to transmission through space.

Monochromatic Fields and Irradiance

Let

$$U_1(x,y) = A_1(x,y)\exp\left[j\phi_1(x,y)\right] \qquad \text{Eq. 1}$$

be the field in the x–y plane is located at some position "1" on the z axis.
Detectors do not currently exist that can follow the extremely high-frequency oscillations ($>10^{14}$ Hz) of the optical electric field. Instead, optical detectors respond to the time-averaged squared magnitude of the field. So, a quantity of considerable interest is the irradiance, which is defined here as

$$I_1(x,y) = U_1(x,y)U_1(x,y)^* = \left|U_1(x,y)\right|^2 \qquad \text{Eq. 2}$$

Irradiance is a radiometric term for the flux (watts) per unit area falling on the observation plane. It is a power density quantity that in other laser and Fourier optics references is often called "intensity." Expression (2) actually represents a shortcut for determining the time-averaged square magnitude of the field and is valid when the field is modelled by a complex phasor.

Optical Path Length and Field Phase Representation

The refractive index n of a medium is the ratio of the speed of light in vacuum to the speed in the medium. For example, a typical glass used for visible light might have an index of about 1.6. For light propagating a distance d in a medium of index n, the *optical path length* (OPL) is defined as

$$\mathrm{OPL} = nd \qquad \text{Eq. 3}$$

The OPL multiplied by the wavenumber k shows up in the phase of the complex exponential used to model the optical field. Think of k as the "converter" between the distance spanned by one wavelength and 2 π (rad of the phase). If a plane wave propagates a distance d through a piece of glass with index n, then the OPL is as indicated in Eq. (3), and the field phasor representation is

$$U(d) = A\exp(jknd) \qquad \text{Eq. 4}$$

In effect, the wavelength shortens to λ/n in the glass. There are other variations of this theme; for example, $\exp(jkr)$, where r is a radial distance in vacuum. Phasor forms associated with the optical field can also be a function of transverse position x and y; for example,

$$\exp\left[j\frac{k}{2z}\left(x^2+y^2\right)\right] \quad \text{Eq. 5}$$

This is known as a "chirp" term and indicates a field phase change as the square of the transverse position. This type of term appears in a variety of situations to model a contracting or expanding optical field.

3.2 Diffraction and Rayleigh–Sommerfeld solution

Consider the propagation of monochromatic light from a 2D plane (source plane) indicated by the coordinate variables η and ξ (Fig. 1). At the source plane, an area: defines the extent of a source or an illuminated aperture. The field distribution in the source plane is given by $U_1(\eta,\xi)$, and the field $U_2(x, y)$ in a distant observation plane can be predicted using the first Rayleigh–Sommerfeld diffraction solution

$$U_2(x,y)=\frac{z}{j\lambda}\iint_{\Sigma}U_1(\xi,\eta)\frac{\exp(jkr_{12})}{r_{12}^2}d\xi d\eta \quad \text{Eq. 6}$$

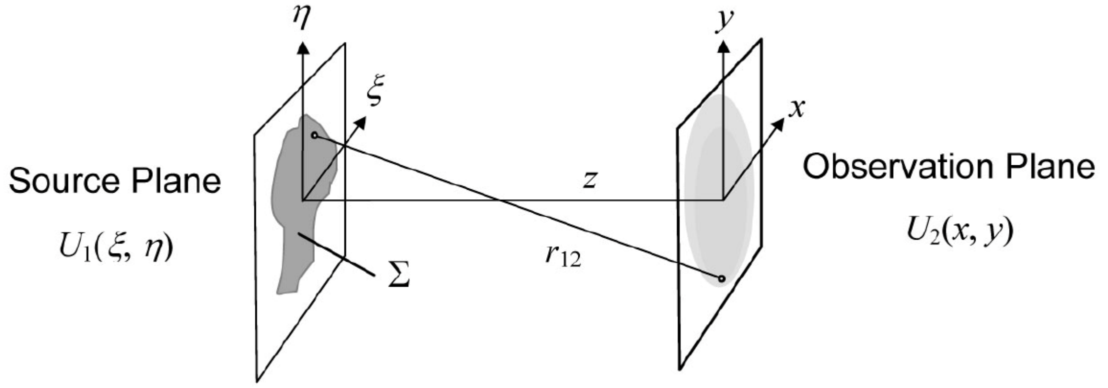


Figure 1 Geometry for a situation where light propagates from a source plane to an observation plane

Here, λ is the optical wavelength; k is the wavenumber, which is equal to $2\pi/\lambda$ for free space; z is the distance between the centres of the source and observation coordinate systems; and r_{12} is the distance between a position on the source plane and a position in the observation plane. η and ξ are variables of integration, and the integral limits correspond to the area of the source : With the source and observation positions defined on parallel planes, the distance r_{12} is

$$r_{12}=\sqrt{z^2+(x-\xi)^2+(y-\eta)^2} \quad \text{Eq. 7}$$

Expression (6) is a statement of the Huygens–Fresnel principle. This principle supposes the source acts as an infinite collection of fictitious point sources, each producing a spherical wave associated with the actual source field at any position (ξ,η). The contributions of these spherical waves are summed at the observation position (x, y), allowing for interference. The extension of Eqs. (6) and (7) to nonplanar geometries is straightforward; for example, involving a more complicated function for r, but the planar geometry is more commonly encountered, and this is our focus here. Expression (6) is, in general, a superposition integral, but with the source and observation areas defined on parallel planes, it becomes a convolution integral, which can be written as

$$U_2(x,y)=\iint U_1(\xi,\eta)h(x-\xi,y-\eta)d\xi d\eta \quad \text{Eq. 8}$$

where the general form of the Rayleigh–Sommerfeld impulse response is

$$h(x,y)=\frac{z}{j\lambda}\frac{\exp(jkr)}{r^2} \qquad \text{Eq. 9}$$

and

$$r=\sqrt{z^2+x^2+y^2} \qquad \text{Eq. 10}$$

The Fourier convolution theorem can be applied to rewrite the integral equation Eq. (8) as

$$U_2(x,y)=\Im^{-1}\left\{\Im\{U_1(x,y)\}\Im\{h(x,y)\}\right\} \qquad \text{Eq. 11}$$

For this convolution interpretation the source and observation plane variables are simply relabeled as x and y. An equivalent expression for Eq. (11) is

$$U_2(x,y)=\Im^{-1}\left\{\Im\{U_1(x,y)\}\Im\{H(f_x,f_y)\}\right\} \qquad \text{Eq. 12}$$

where H is the Rayleigh–Sommerfeld transfer function given by

$$H(f_x,f_y)=\exp\left(jkz\sqrt{1-(\lambda f_x)^2-(\lambda f_y)^2}\right) \qquad \text{Eq. 13}$$

Strictly speaking, $\sqrt{f_x^2+f_y^2}<1/\lambda$ must be satisfied for propagating field components. An angular spectrum analysis is often used to derive Eq. (13). **This solution only requires that r >> λ, the distance between the source and the observation position, be much greater than the wavelength.**

Fresnel approximation

The square root in the distance terms of Eq. (7) or (10) can make analytic manipulations of the Rayleigh–Sommerfeld solution difficult and add execution time to a computational simulation. By introducing approximations for these terms, a more convenient scalar diffraction form is developed. Consider the binomial expansion

$$\sqrt{1+b}=1+\frac{1}{2}b-\frac{1}{8}b^2+\ldots \qquad \text{Eq. 14}$$

where b is a number less than 1, then expand Eq. (7) and keep the first two terms to yield

$$r_{12}\approx z\left[1+\frac{1}{2}\left(\frac{x-\xi}{z}\right)^2+\frac{1}{2}\left(\frac{y-\eta}{z}\right)^2\right] \qquad \text{Eq. 15}$$

This approximation is applied to the distance term in the phase of the exponential in Eq. (8), which amounts to assuming a parabolic radiation wave rather than a spherical wave for the fictitious point sources. Furthermore, use the approximation $r_{12} = z$ in the denominator of Eq. (6) to arrive at the *Fresnel diffraction* expression:

$$U_2(x,y)=\frac{e^{jkz}}{j\lambda z}\iint U_1(\xi,\eta)\exp\left\{j\frac{k}{2z}\left[(x-\xi)^2+(y-\eta)^2\right]\right\}d\xi d\eta \qquad \text{Eq. 16}$$

This expression is also a convolution of the form in Eq. (8), where the impulse response is

$$h(x,y)=\frac{e^{jkz}}{j\lambda z}\exp\left[\frac{jk}{2z}(x^2+y^2)\right] \qquad \text{Eq. 17}$$

and the transfer function is

$$H\left(f_x, f_y\right) = e^{jkz} \exp\left[j\pi\lambda z\left(f_x^2 + f_y^2\right)\right] \quad \text{Eq. 18}$$

The expressions in Eqs. (11) and (12) are again applicable in this case for computing diffraction results.

Another useful form of the Fresnel diffraction expression is obtained by moving the quadratic phase term that is a function of *x* and *y* outside the integrals:

$$U_2(x,y) = \frac{\exp(jkz)}{j\lambda z} \exp\left[j\frac{k}{2z}\left(x^2+y^2\right)\right] \times$$

$$\iint \left\{U_1(\xi,\eta)\exp\left[j\frac{k}{2z}\left(\xi^2+\eta^2\right)\right]\right\} \exp\left[-j\frac{2\pi}{\lambda z}(x\xi + y\eta)\right] d\xi d\eta \quad \text{Eq. 19}$$

Along with the amplitude and chirp multiplicative factors out front, this expression is a Fourier transform of the source field times a chirp function where the following frequency variable substitutions are used for the transform:

$$f_\xi \to \frac{x}{\lambda z} \qquad\qquad f_\eta \to \frac{y}{\lambda z} \quad \text{Eq. 20}$$

The accuracy of the Fresnel expression when modelling scalar diffraction at close ranges suffers as a consequence of the approximations involved. A criterion which is commonly used for determining when the Fresnel expression can be applied is the Fresnel number. The Fresnel number is given by

$$N_F = \frac{w^2}{\lambda z} \quad \text{Eq. 21}$$

where ***w* is the half width of a square aperture in the source plane**, or the radius of a circular aperture, and *z* is the distance to the observation plane. If **N_F is larger than 1** for a given scenario, then it is commonly accepted that the observation plane is in the near field region, where the Fresnel approximations, typically, lead to useful results. However, for relatively "smooth" fields over the source aperture, the Fresnel expression can be applicable up to Fresnel numbers of even 20 or 30. In a geometrical optics context, the Fresnel expression describes diffraction under the paraxial assumption, where only rays that make a small angle (< ~0.1 rad) relative to the optical axis are considered.

Fraunhofer approximation

Fraunhofer diffraction, which refers to diffraction patterns in a regime that is commonly known as the "far field," is arrived at mathematically by approximating the chirp term multiplying the initial field within the integrals of Eq. 19 as unity. The assumption involved is

$$z \gg \left(\frac{k\left(\xi^2+\eta^2\right)}{2}\right)_{max} \quad \text{Eq. 22}$$

and results in the Fraunhofer diffraction expression:

$$U_2(x,y) = \frac{\exp(jkz)}{j\lambda z} \exp\left[j\frac{k}{2z}\left(x^2+y^2\right)\right] \iint U_1(\xi,\eta)\exp\left[-j\frac{2\pi}{\lambda z}(x\xi + y\eta)\right] d\xi d\eta \quad \text{Eq. 23}$$

The condition of Eq. (22), typically, requires very long propagation distances relative to the source support size. However, a form of the Fraunhofer pattern also appears in the propagation analysis involving lenses. The Fraunhofer diffraction expression is a powerful

tool and finds use in many applications such as laser beam propagation, image analysis, and spectroscopy.
Along with multiplicative factors out front, the Fraunhofer expression can be recognized simply as a Fourier transform of the source field with the variable substitutions

$$f_\xi \to \frac{x}{\lambda z} \qquad\qquad f_\eta \to \frac{y}{\lambda z} \qquad \text{Eq. 24}$$

The Fraunhofer expression cannot be written as a convolution integral, so there is no impulse response or transfer function. But, since it is a scaled version of the Fourier transform of the initial field, it can be relatively easy to calculate, and as with the Fresnel expression, the Fraunhofer approximation is often used with success in situations where Eq. (22) is not satisfied. For simple source structures such as a plane-wave illuminated aperture, the Fraunhofer result can be useful even when Eq. (22) is violated by more than a factor of 10, particularly if the main quantity of interest is the irradiance pattern at the receiving plane. Using the Fresnel number N_F, the commonly accepted requirement for the Fraunhofer region is $N_F << 1$.
Looking at the equations above one can see that in this region (Fraunhofer region), **$U_2(x,y)$ is just the two-dimensional Fourier transform of $U_1(x,y)$ except for a multiplicative phase factor which does not affect the intensity of the light.** This regime is also called Fraunhofer Diffraction or Fraunhofer Approximation. The examples presented here are calculated numerically assuming Fraunhofer approximation, i. e. simple two-dimensional Fourier transform.

3.2 Gratings and Periodic Functions

The following explanations are a shortened version of the document: Fourier Optics in Examples fourier.tex kb 20020205 by Klaus Betzler, Fachbereich Physik, Universität Osnabrück.

When the two-dimensional pattern is only structured in one dimension, that also shows up in the Fourier transform, yet in a reciprocal meaning. This is visualized by Figs. 2 and 3.

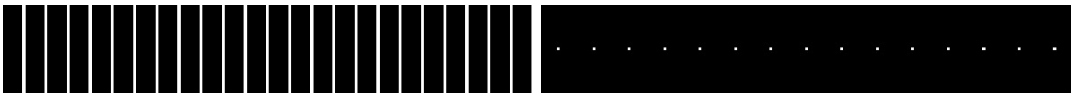


Figure 2: Array of lines (left) and the corresponding two-dimensional Fourier transform (right).

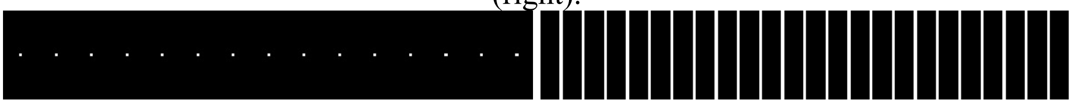


Figure 3: Array of points (left) and the corresponding two-dimensional Fourier transform (right).

In Fig. 2 the pattern is constant in the vertical dimension, its Fourier transform shows a delta function behavior in this dimension, yielding a linear array of points. Vice versa for Fig. 3. That’s due to the fact that the Fourier transform of a constant is the delta function and vice versa.
The number of elements in the original pattern strongly determines the sharpness of the diffraction pattern. Fig. 4 demonstrates this using a one-dimensional regular structure of points as source pattern. Depending on the number of points used, the diffraction pattern varies in sharpness.

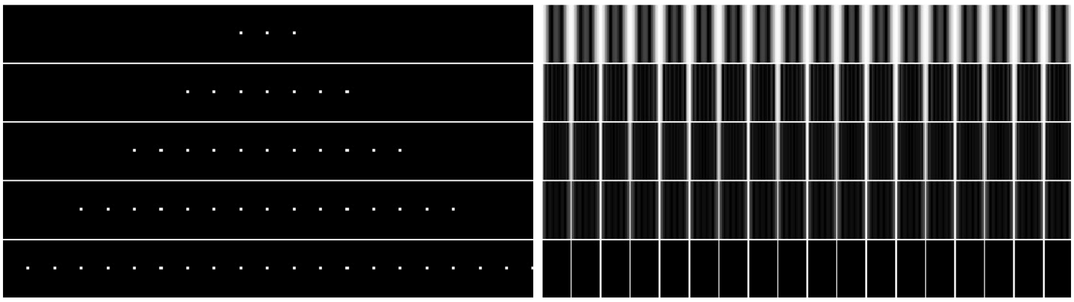


Figure 4: Dependence of the diffraction pattern on the number of source objects used. The sharpness increases with this number (from top to bottom: 3, 7, 11, 15, 19).

A special problem in Fourier transform is the fact that one always has to deal with limited data, albeit theory assumes unlimited data to be transformed. Using limited data means to make a transformation of the product of the unlimited data with a rectangular function. The Fourier transform in that case is the convolution of the two transforms. As the transform of a rectangular function shows expressed side wings, these also show up in the transform of the product, mainly convoluted to each of the peaks of the transform. This spurious additional intensity may affect the pattern of the Fourier transform producing fictitious information. The effect is shown in Fig. 5, the peaks are smeared out, surrounded by undesired side wings.

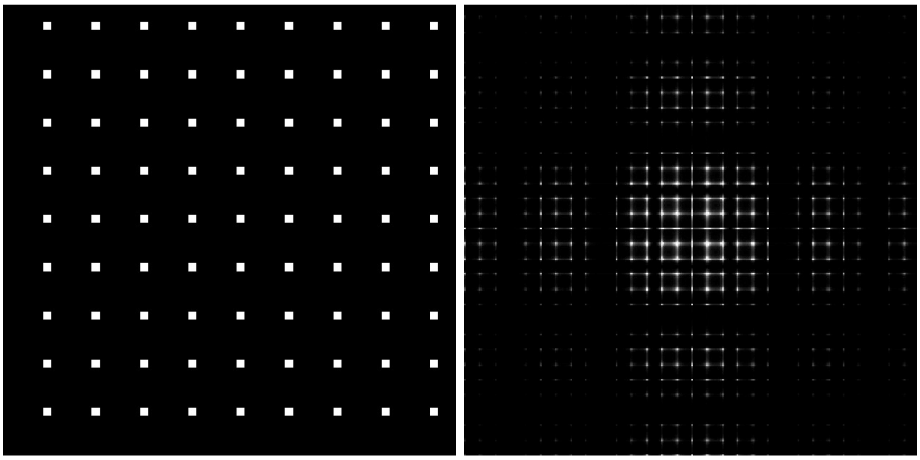


Figure 5: Sharply limited pattern (left) and its Fourier transform (right). Side wings appear at the peaks.

The effect can by reduced by smoothing the sharp edges of the original pattern. This treatment is called apodization or windowing. The pattern is multiplied by an appropriate apodization or windowing function. In Fig. 6 this is done using $\sin^2$ functions

$$\text{pattern} = \text{pattern}\sin^2\left(\frac{x\pi}{L_x}\right)\sin^2\left(\frac{y\pi}{L_y}\right) \qquad \text{Eq. 25}$$

L_x and L_y respectively, are the sizes of the original pattern.

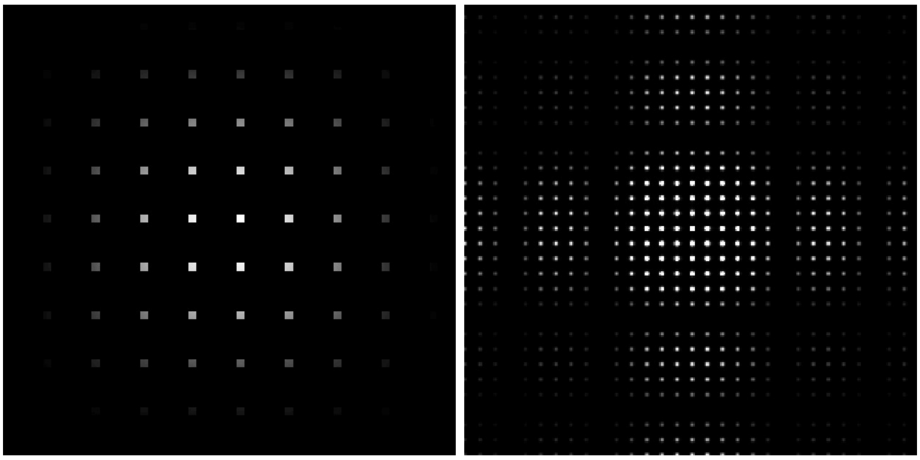


Figure 6: The pattern of Fig. 5 multiplied by an apodization function. The side wings in the Fourier transform disappear.

The resulting Fourier transform now is free from any side wings, yet the peaks are are broadened slightly. This is due to the suppression of the outer parts of the pattern which corresponds to an overall size and thus information reduction.
According to the Sampling Theorem the sampling frequency in discrete Fourier transform (DFT) must be at least twice the highest frequency to be detected. If this requirement cannot be met, Aliasing occurs, i. e. frequencies above this limit are aliased by corresponding lower frequencies.

The extent of the diffraction pattern is complementary to the size of the single diffracting elements. Fig. 7 shows this reciprocal behavior using a single circular shape as example.

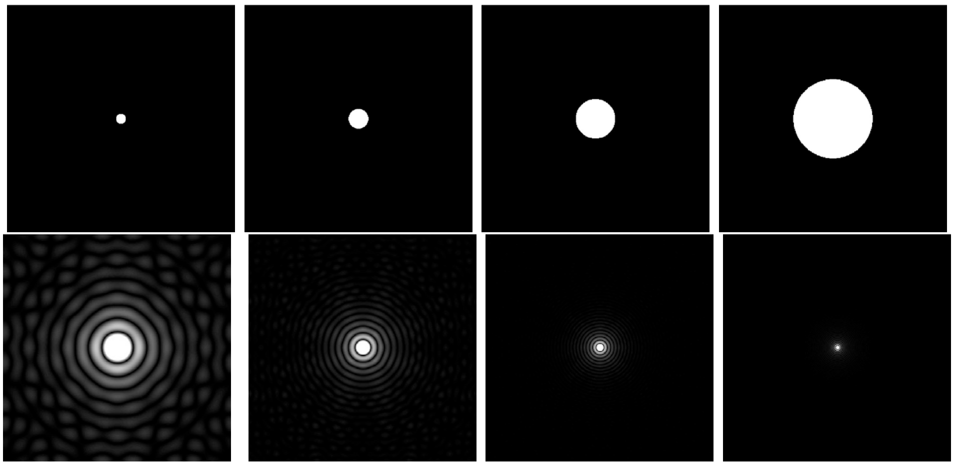


Figure 7: Circular apertures of different size (upper row) and their corresponding Fourier transforms (lower row). The intensities are normalized to their respective maximum value.

Similar results are produced by a two-dimensional regular array of objects.
A similar reciprocity as for the element size and the extent of the diffraction pattern of course must be valid for the periodicity of original and transform. This is one of the essentials of the Fourier transform.
In Fourier transform convolution and product are complementary mathematical operations. The Fourier transform of a product of two functions equates the convolution of the Fourier

transforms of the two functions. Vice versa, the Fourier transform of a convolution of two functions equates the product of the two Fourier transforms of the single functions. A regular array of identical elements can be treated as a convolution of an array of corresponding points and a single element. The Fourier transform then must equate the product of the two elementary transforms. Translated to diffraction optics this means that the diffraction pattern of a regular array can be calculated as the product of the diffraction pattern of a single element and the interference pattern of the point array. Figs. 8 – 10 visualize this property of the Fourier transform.

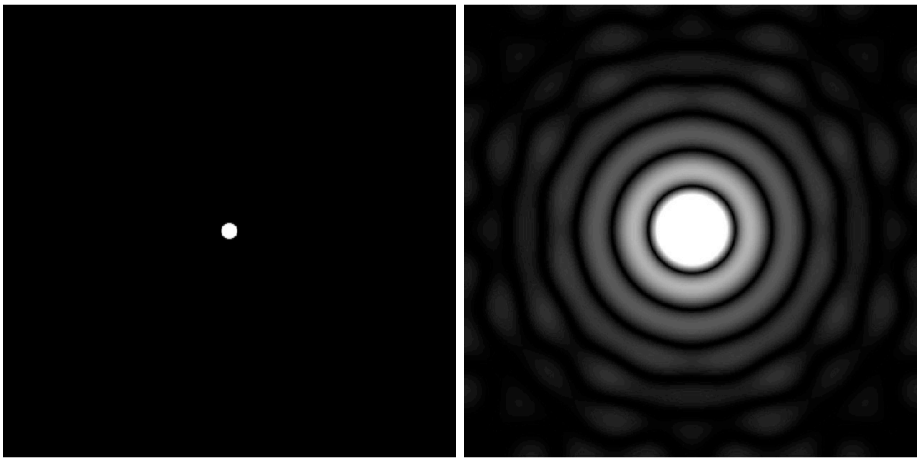


Figure 8: Single circular aperture and its Fourier transform.

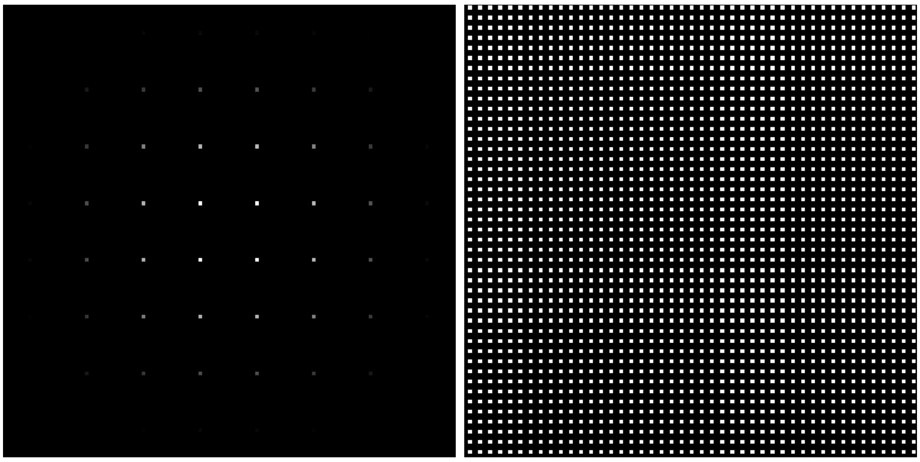
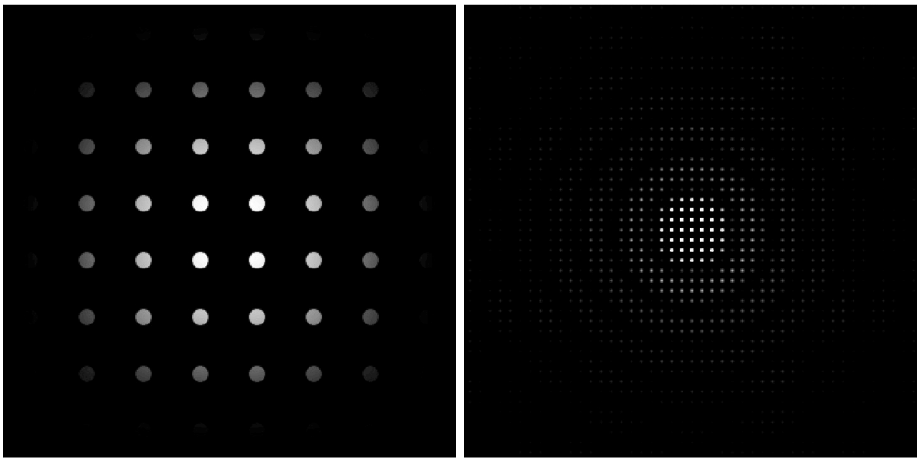


Figure 9: Regular array of points and corresponding Fourier transform



.Figure 10: Regular array of circular apertures (*convolution* of single aperture and point array) and its corresponding Fourier transform (*product* of the respective Fourier transforms).

Note in the simulation in Fig. 8.-10. apodization was applied.
A diffraction grating is a (one-dimensional) array of identical slits or mirror elements. The diffraction pattern can be calculated by one or two-dimensional Fourier transform in a similar way as discussed if we can assume Fraunhofer Approximation. Various typical features are shown in the following figures (appropriate aliasing is used to get sharp diffraction patterns)

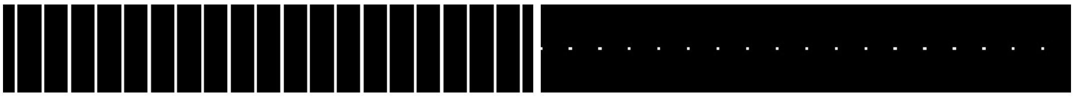


Figure 11: Ideal grating (narrow slits) and corresponding diffraction pattern.

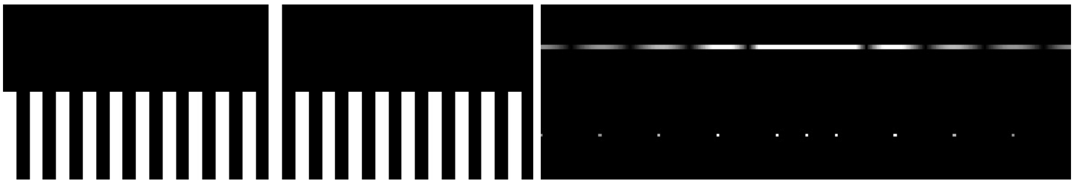


Figure 12: Slit width equates one half of the period: The minima of the slit diffraction function (top right) correspond to the maxima N=±2, ±4,...of the grating diffraction resulting in missing maxima (bottom right).

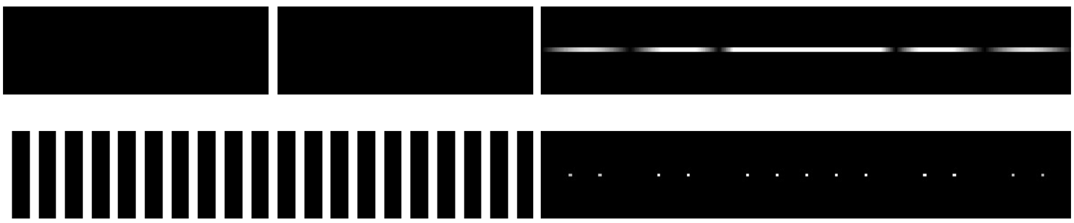


Figure 13: Slit width equates one third of the period: The minima of the slit diffraction function (top right) correspond to the maxima N=±3, ±6,...(result bottom right).

Fraunhofer Diffraction Example

It is extremely difficult (impossible?) to find closed-form diffraction solutions using the Rayleigh–Sommerfeld expression for most apertures. The Fresnel expression is more tractable, but solutions are still complicated even for simple cases such as a rectangular aperture illuminated by a plane wave. Analytic Fraunhofer diffraction analysis is easier and, for our purposes, serves as a check on some of the computer results. Consider a circular aperture illuminated by a unit amplitude plane wave. The complex field immediately beyond the aperture plane is

$$U_1(\xi,\eta) = \mathrm{circ}\left(\frac{\sqrt{\xi^2+\eta^2}}{w}\right) \qquad \text{Eq. 26}$$

To find the Fraunhofer diffraction field, the Fourier transform is taken as

$$\Im\{U_1(\xi,\eta)\} = w^2 \frac{J_1\left(2\pi w\sqrt{f_\xi^2+f_\eta^2}\right)}{w\sqrt{f_\xi^2+f_\eta^2}} \qquad \text{Eq. 27}$$

Then, with the substitutions in Eq. (24), and applying the leading amplitude and phase terms of Eq. (23), the field is found with

$$I_2(x,y) = \left(\frac{w^2}{\lambda z}\right)^2 \left[\frac{J_1\left(2\pi \frac{w}{\lambda z}\sqrt{x^2+y^2}\right)}{\frac{w}{\lambda z}\sqrt{x^2+y^2}}\right]^2 \qquad \text{Eq. 28}$$

Some of the $w/\lambda z$ terms could be cancel out, but the symmetry of this form is helpful for programming.

4 Setup and equipment

4.1 Materials

- The camera (1600x1200 pixels, colour, pixel size 2.835 um) C600 from Logitech
- Three different light sources (Halogen, LED, laserdiode) USB driven
- Sheet polarizer for intensity regulation
- Objective lens for the Logitech C600
- Planconvex lens diameter D=9 mm, f=12 mm (Thorlabs)
- Mechanical setup
- Wafer with diffracting microstructures

4.2 Analytical calculation of Fraunhofer diffraction

Let's exercise MATLAB to display an irradiance pattern. We want to calculate

$$I(x,y)=\left(\frac{w^2}{\lambda z}\right)^2\left[\frac{J_1\left(2\pi\frac{w}{\lambda z}\sqrt{x^2+y^2}\right)}{\frac{w}{\lambda z}\sqrt{x^2+y^2}}\right]^2 \qquad \text{Eq. 28}$$

For the example below I have chosen parameters at will. You should use the code and change the parameters to the actual setting in your experiment!

The following parameters have been used:

Aperture (round or square) half wdith	$\boldsymbol{w}$ **= 1 mm**
Wavelength	**λ = 633 nm (He–Ne laser wavelength)**
Distance from aperture	**z = 50 m**

The Fresnel number constraint will assure that we can use Fraunhofer diffraction formula (Fourier Transform) and requires $w^2/\lambda z < 0.1$ or $z > 10w^2/\lambda$, which leads to $z > 15.8$ m. We'll use $z = 50$ m.

Now to choose some mesh parameters. A first parameter is the **size of the screen L** which would in our case correspond to the **detector size**!
A good display size for the function is if the array side length is perhaps five times wider than the pattern's central lobe. The Bessel function J_1 has a first zero when the argument is equal to 1.22. If $y = 0$, then the first zero in the pattern occurs when

$$2\pi\frac{w}{\lambda z}x = 1.22\pi \qquad \text{Eq. 29}$$

Solve for x to get half the center lobe width and double this result to get the full width of the center lobe

$$D_{lobe} = 1.22\frac{\lambda z}{w} \qquad \text{Eq. 30}$$

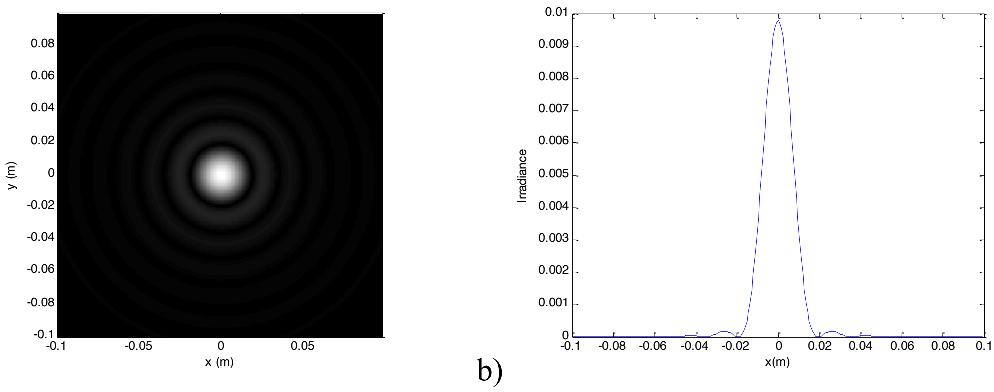
We will choose $L = 5 \times 1.22\, \lambda z/w = 0.2$ m.

Screen or detector size L **0.2 m**

A possible implantation of then code is given below and available in the file “fraun_circ.m”. The code uses functions that should be in the same directory (“jjnc.m” and “rect.m”)

```
%fraun_circ - Fraunhofer irradiance plot
L=0.2; %observation screen size or detector size for a squared detector
(measured in m)

M=250; % number of samples
dx=L/M; % corresponindg sample interval in space
x=-L/2:dx:L/2-dx; y=x; %defines the vector for the screen/detector
[X,Y]=meshgrid(x,y);

w=1e-3; %x aperture half-width, form not specified
lambda=0.635e-6;%wavelength
z=50; % propagation distance, distance from the aperture to the detector

k=2*pi/lambda; %wavenumber, often used in calculation
lz=lambda*z;% parameter neeeded for caculation

%Irradiance calculation with analytical formula
Irr=(w^2/lz)^2.*(jinc(w/lz*sqrt(X.^2+Y.^2))).^2;

figure(1) %irradiance image plot with different options
imagesc(x,y,nthroot(Irr,2));
xlabel('x (m)'); ylabel('y (m)');
colormap('gray');
axis square;
axis xy;

figure(2) %x-axis profile plot with different options
plot(x,Irr(M/2+1,:),'LineWidth',2);
xlabel('x(m)'); ylabel('Irradiance');
```

In bold the line were the irridance calculation is effectively done.
Running the script produces the results in Fig. 14. The Fraunhofer pattern of a circular aperture is commonly known as the ***Airy pattern***. The central core of this pattern, whose width is given in Eq. (30), is known as the ***Airy disk***.



a) b)

Figure 14 Fraunhofer irradiance (a) image pattern and (b) x-axis profile for a circular aperture. This is known as the Airy pattern.

4.3 Diffraction with a collimated beam setup and verification

Mechanical setup:

Start with the breadboard and mount the translation stage.

Figure 15: Breadboard with translation stage

The stage is fixed with screws arranged asymmetric (see the right picture). Continue by mounting the adapter plate and the intermediate piece as below.

Figure 16: Adapter plate and intermediate fixing have to be screw together.

The cameras PCB (printed circuit board) special holder is mounted as shown below.

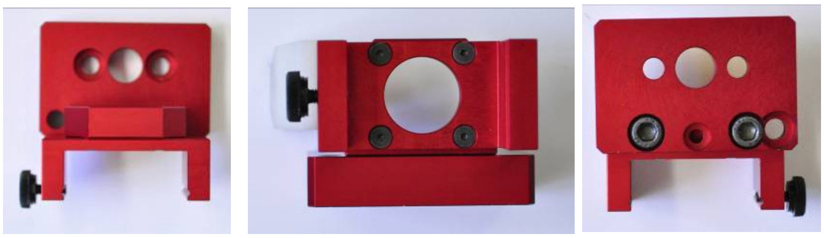


Figure 17: Camera holder (left) mounted on the adapter plate with the intermediate piece. Right: as seen from below.

The assembly has to be fixed on the translation stage before the camera PCB is put into place.

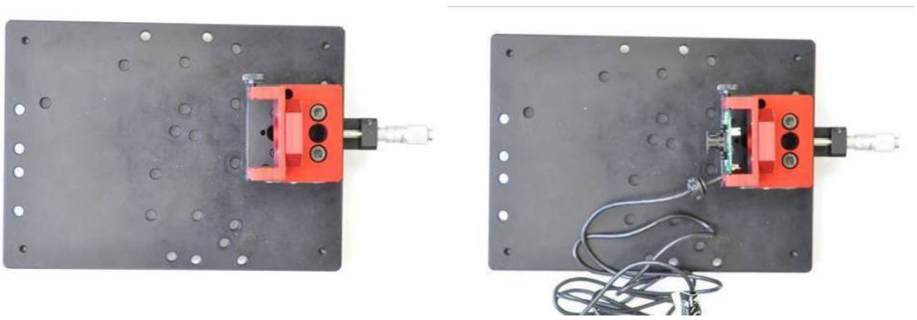


Figure 18 Mechanical holder to fix the camera

Figure 19: Camera PCB in the mount ready for shooting.

Mount the source holder at the other side of the breadboard so that the source is in front of the camera and put the laser in place. Switch on the laser (just connect to USB of the PC) and apply the lens cap

Figure 20: Source mounted with the lens cap (on the right) to achieve focussing and collimation

You should have a setup similar to that shown below.

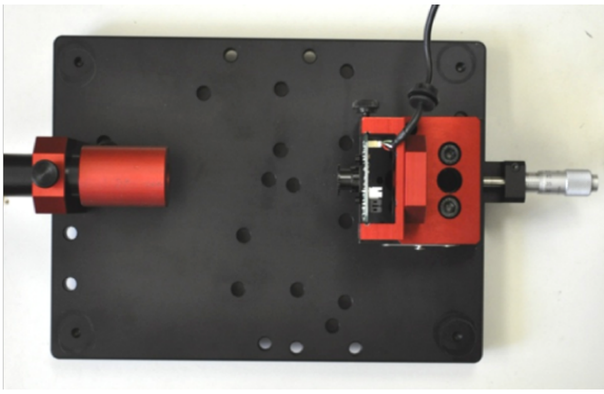


Figure 21: Collimated beam setup. The laser with the lens cap on the left illuminates the camera to the right.

The red tube fixed over the source carries a plano-convex lens (f=12 mm,Ø=9 mm) made from BK7 (Thorlabs L1576). In addition a polarizer is fixed inside the tube (Edmund Scientific part number ES45668). Moving the tube longitudinally (along the optical axis) focuses the light. Turning the tube adjusts the intensity (because there is a polarizer in the tube and the source is polarized too).

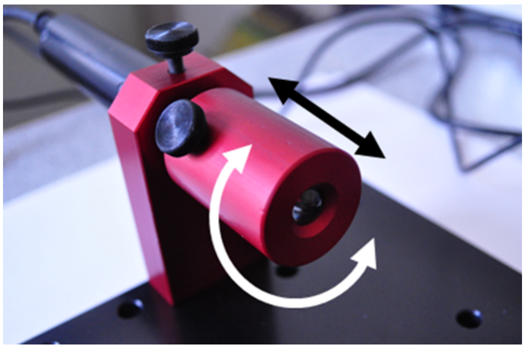


Figure 22. Focalisation (black arrow) and intensity adjustment with the lens cap.

A collimated beam shows nearly no change of size when seen at different distance. This can be probed with a sheet of paper as shown in the image sequence below.

- Moving the tube longitudinally (along the optical axis or body of the source) focuses the light.
- Turning the tube adjusts intensity (because there is a polarizer in the tube and the source is polarized too).
-

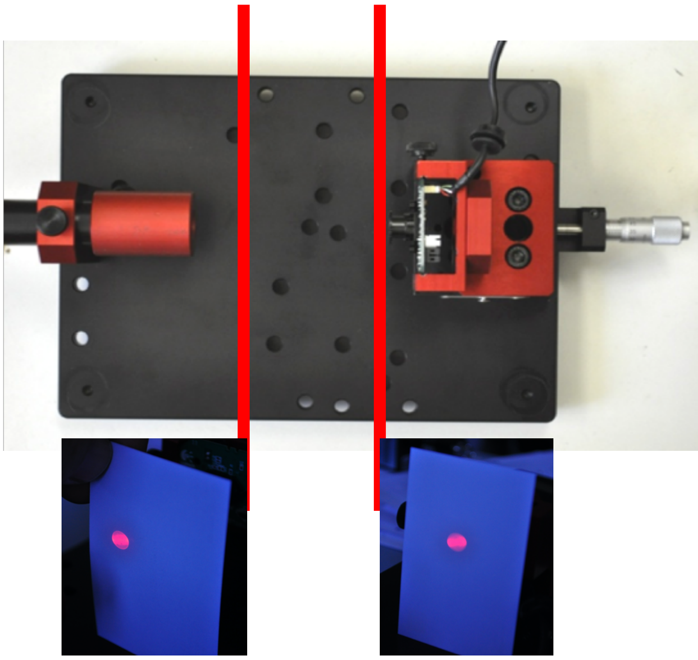


Figure 23: Light spot at different distances from the collimated source. There is nearly no change.

Adjust the lens cap to have the best possible collimation. To make it happen the lens cap has to be moved along the optical axis to adjust the focus and the sheet of paper has to be used regularly to probe the collimation quality.
For our measurements we will use two different arrangements: without camera objective for direct observation of diffraction and with objective to observe grating diffraction. The IR filter is a part of the objective holder of the C600 camera objective. (only remove the lens part and leave the holder mounted). Use the sheet polarizer for additional intensity adaption. The sheet polarizer can just be put between lens and camera.

- Remove the Logitech C600 camera objective **but leave the IR filter!** (just unscrew the objective lens body).
- Set the linear stage at the middle position (5 mm)

NOTE: The IR filter has to left mount in front of the camera.

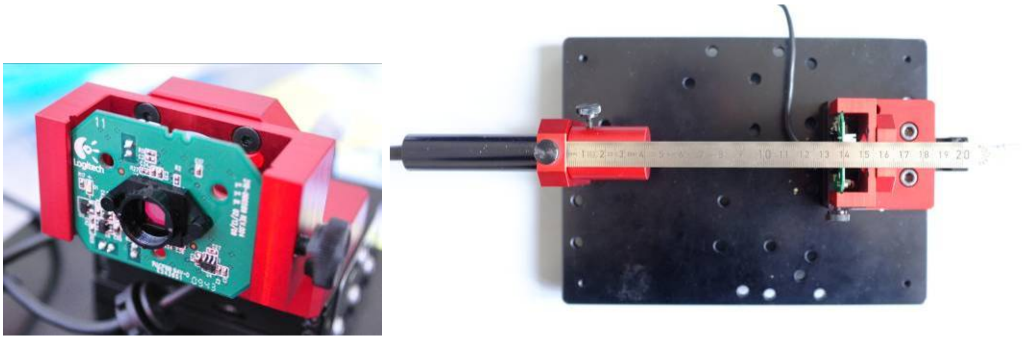


Figure 24. Details of the direct diffraction setup. The objective is removed and the distance between the detector surface and lens cap is set to a convenient value (i.e. 10mm)

The diffractive structures will be put in front of the lens cap. So the object plane is the end of the lens cap. It is important to know the distance between object and the observation plane. Measure this carefully with the ruler and adjust with the linear stage to a convenient position. Put different diffracting objects in front of the lens cap and take images of their diffraction patterns. Analyse the results with the scripts given below.

Diffraction at a circular aperture

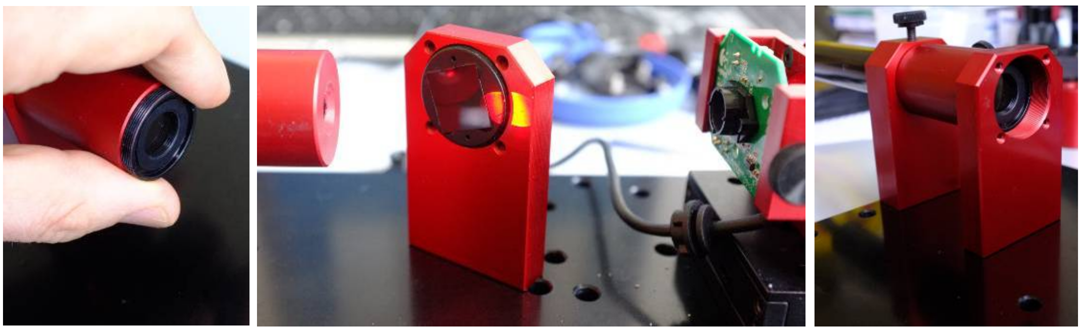


Figure 25 Left: Pinhole hold in front of the lens cap at the right position. It could also be mounted in the support and set in front of the source (Right).

Use the script that you have implemented above and analytically calculate the diffraction pattern**. You need to adjust the parameters in the script to your experimental setup**. Be careful that all the parameters are correctly chosen and note the parameters in your report.

Parameters to adjust for simulation:

```
L=0.2; %observation screen size or detector size for a squared detector
(measured in m)
```

```
w=1e-3; %x aperture half-width, form not specified
lambda=0.635e-6;%wavelength
z=50; % propagation distance, distance from the aperture to the detector
```

Remember: Our detector (L) is 4.536mm x 3.416mm. The aperture half width (w) is 50 micron. Our wavelength is 635 nm and a typical propagation distance is 7 cm (to be measured by you!). Use now your parameters for simulation and compare with measurement.

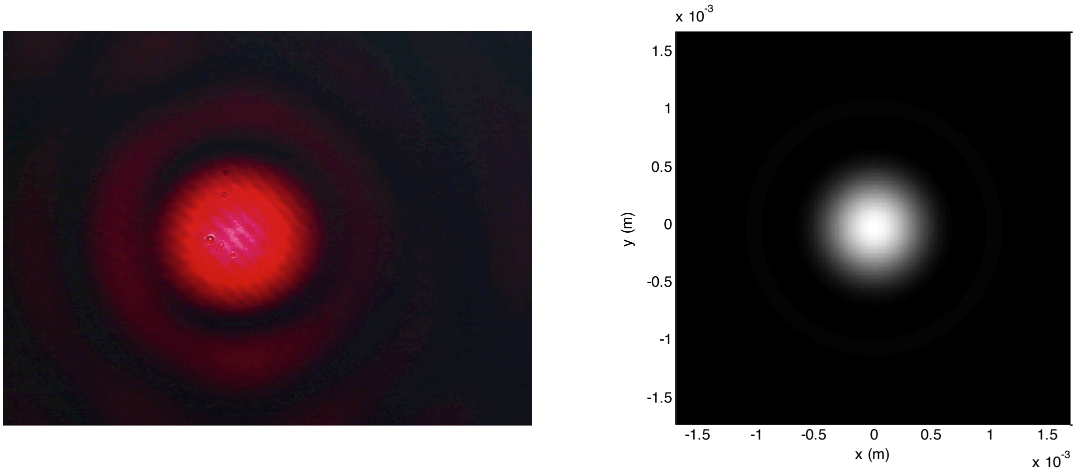


Figure. 26 Diffraction pattern measured and simulated. The height of the simulation field matches the detector height (3.4 mm)! Propagation distance in my case 7 cm (you might have different values!!!). Pinhole diameter 100 micron (half size 50 micron)

Rectangular aperture diffraction

The diffractive structure are on different kind of substrates. There is a wafer (glassplate 4 inch), metal foils and a print on plastic.

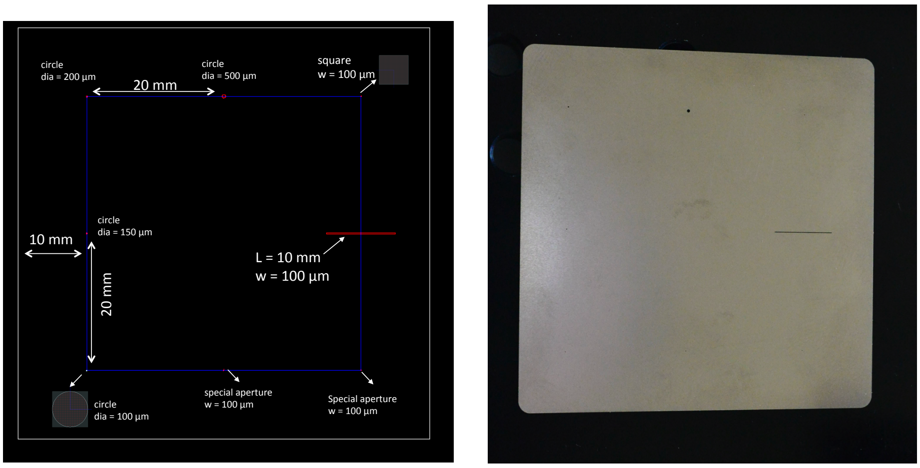


Figure 27. Layout and real metal foil used for some of the diffraction experiments. There are different elements such us circular aperture square apertures and slits.

You need to identify yourself which one is the square aperture!! It is one of the dark square on the foil or wafer. Positioning is very important. Take your time to centre it carefully. Adjust intensity – take images.

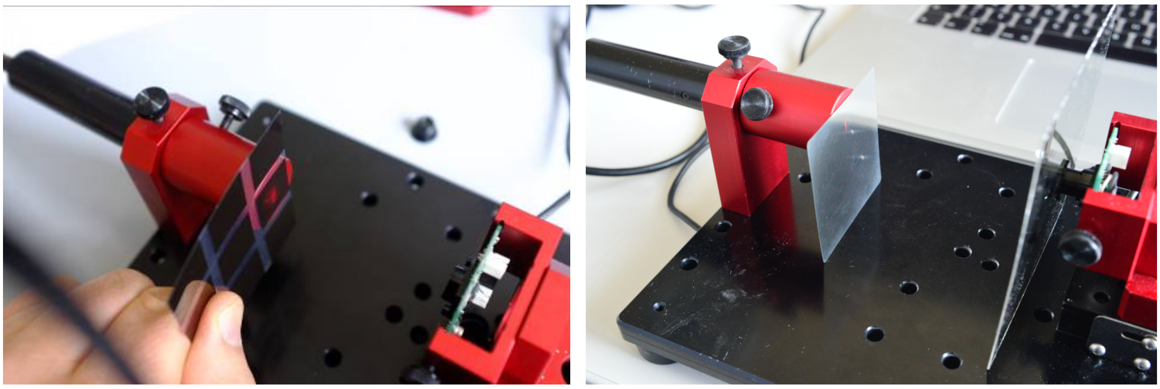


Figure 28. Rectangular aperture applied for diffraction. Please note that not all diffractive structures have the same quality. Here we use the plastic foil to the left and the metal foil to the right.

The idea is that you take images of the diffraction pattern and compare it with simulation. We return to the script.
Use the script that is already implemented above and change the irradiance function to:

```
%irradiance
I2=(4*w^2/lz)^2.* (sinc(2*w/lz*X).* sinc(2*w/lz*Y)).^2;
```

You need to adjust the parameters in the script to the current problem. Be careful that all the parameters are correctly chosen and note the parameters in your report.

Parameters to adjust:

```
L=0.2; %observation screen size or detector size for a squared detector
(measured in m)
w=1e-3; %x aperture half-width, form not specified
lambda=0.635e-6;%wavelength
z=50; % propagation distance, distance from the aperture to the detector
```

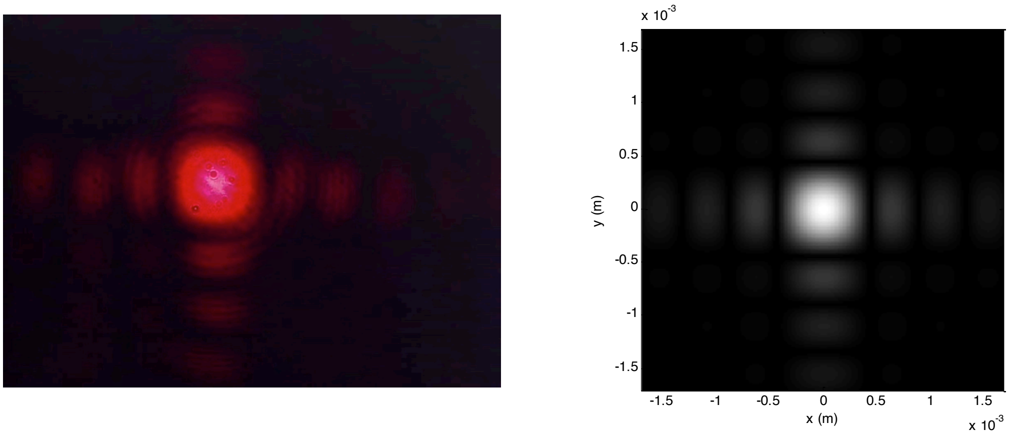


Figure 29. Diffraction of a square aperture measured and calculated. Left measured, right simulated

TO BE DONE FOR THE REPORT: Take pictures of the diffraction pattern for the 100 micron pinhole and 100 micron rectangle. Measure the distance from the diffracting object to the camera chip (usually 100mm). Calculate the theoretical 2 dimensional diffraction pattern with Matlab and plot it aside to the measured ones. Write your simulation parameters in the report.

4.4 Fraunhofer diffraction for a slit

In this experiment with a slit we want to measure the diffracted intensity and simulate the diffraction pattern with Fraunhofer propagation.

Simulation of diffraction pattern

In the case of slit diffraction it is possible to compare line plots of the intensity with simulation. Averaging helps to smooth the signal. The idea here is to use a FFT diffraction script that calculates the theoretical diffraction plot and charge the measurement date in the same plot for comparison.
To compare with simulation you should use the script as given in the file “fft_example_1d.m” that performs the Fourier transform.

```
fft_example
w = 0.055; %slit half-width (m)
L = 2; % vector side length (m), corresponds to the sizte of the detector
or screen
M = 200; %number of samples used for calculation
dx = L/M; %sample interval (m), corresponding sampling in space
x = -L/2:dx:L/2-dx; %coordinate vector, corresponding x
f = rect(x/(2*w)); %signal vector, amplitude vector at the entrance
figure(1) %plotting the slit amplitude for the whole field
plot(x,f); %plot f vs x
figure(2) %plotting the slit amplitude for a zoom into the axis and show
sampling
plot(x,f,'-o'); %plot f vs x
axis([-0.2 0.2 0 1.5]);
xlabel('x(m)');
figure(3 )%plotting the slit amplitude for a zoom into the axis lotted
axainst sampling points, gives the same graph as figure 2 only x axis
changed
plot(f,'-o');
axis([80 120 0 1.5]);
xlabel('index');
% Next the  propagation needs to be don by doing a Fourier transform FFT
% first step is preparing data, a shift of the data is necessary
f0=fftshift(f); %shift f
figure(4) %plot of the orginal data but resampled on the vector, needed to
do FFT
plot(f0);
axis([0 200 0 1.5]);
xlabel('index');
% Fourier transfrom can be executed
F0=fft(f0)*dx; %FFT and scale
figure(5)
plot(abs(F0)); %plot magnitude
title('magnitude');
xlabel('index');
figure(6)% Phase of the propagation signal
plot(angle(F0)); %plot phase
title('phase');
xlabel('index');
% After FFT the signal needs to be shifted back to be centered
F=fftshift(F0); %center F
% All calculations were done on a sampled signal, one needs to find the
% correct axis in space, the frequecy coordinates will be established now
fx=-1/(2*dx):1/L:1/(2*dx)-(1/L); %freq cords
figure(7) %Plot of amplitude (magnitude) against the frequency coordinates
plot(fx,abs(F)); %plot magnitude
```

```
title('magnitude');
xlabel('fx (cyc/m)');
figure(8) %Plot of phase against the frequency coordinates
plot(fx,angle(F)); %plot phase
title('phase');
xlabel('fx (cyc/m)');
figure(9) %Plot of intensity against the frequency coordinates
plot(fx,(abs(F)).^2, 'Linewidth',2); %plot intensity
title('instensity');
xlabel('fx (cyc/m)');
```

You have to modify the script. Especially the parameters of the propagation have to be changed to consider the correct geometry of our problem with a 100 micron slit. You need to change:

```
w = 0.055; %slit half-width (m)
L = 2; % vector side length (m), corresponds to the size of the detector or
screen
```

The result of this calculation is the vector "F" that contains then diffraction pattern and will be used in the next step of the experiment.

Simulations for adjusted parameters result in the following sequence of figures

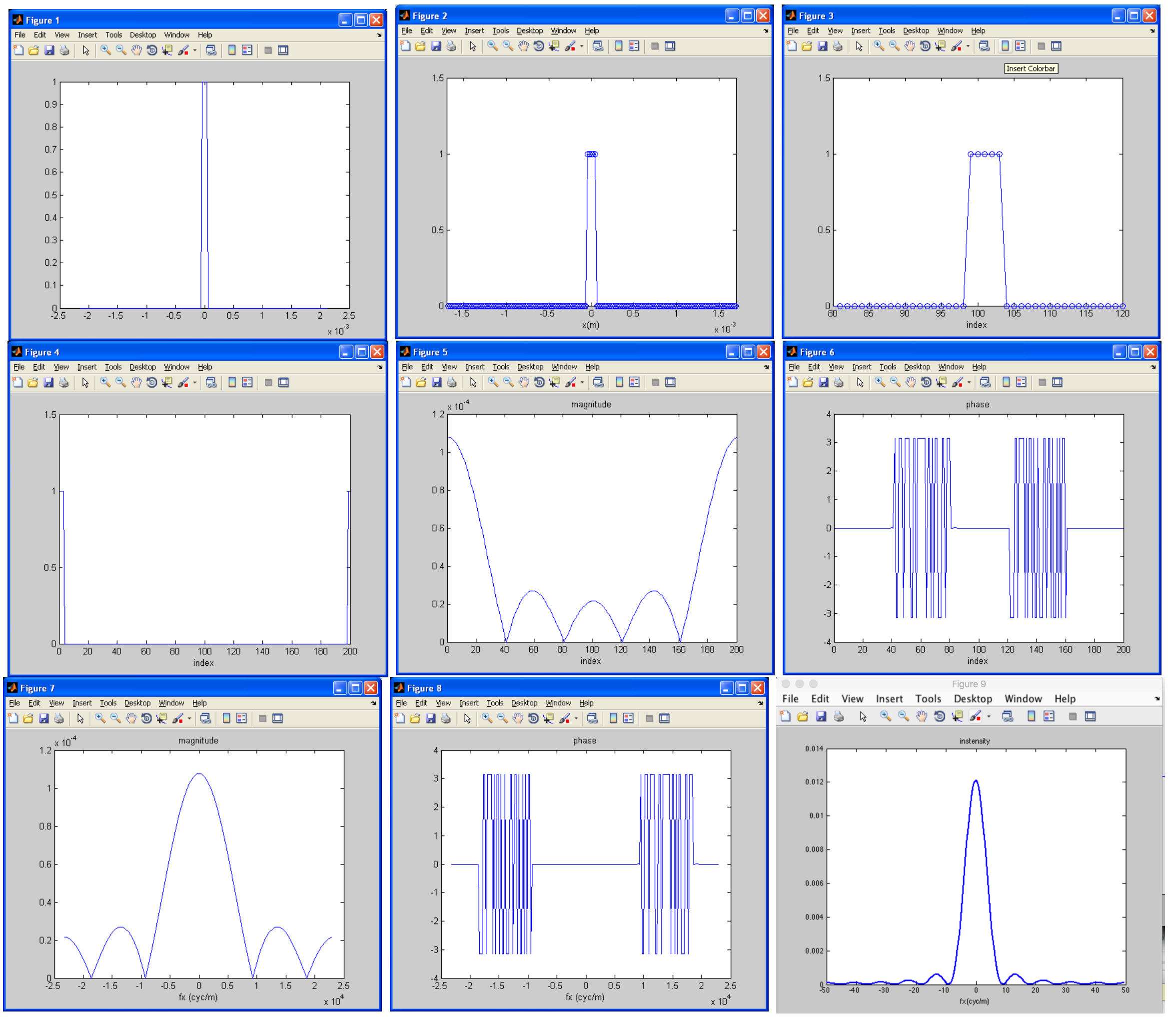


Figure 30 Simulation of diffraction with Fourier transform and Fraunhofer propagation. The slit geometry, Diffraction pattern and Irradiance and on the bottom the final result, the shifted field parameter, magnitude and phase and the **intensity**

If you want to learn more about the function of this script there is a commented version at the end of this document in the appendix.

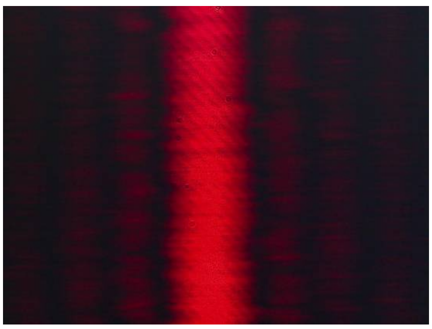
Measurement of Slit diffraction

For measurement we can use the metal foil with different diffraction object that contains a slit of you use us use the red cap with the slit. You can hold the slit as shown below or use the metal foil and just put in front of the lens cap like. Just be sure you use the slit.

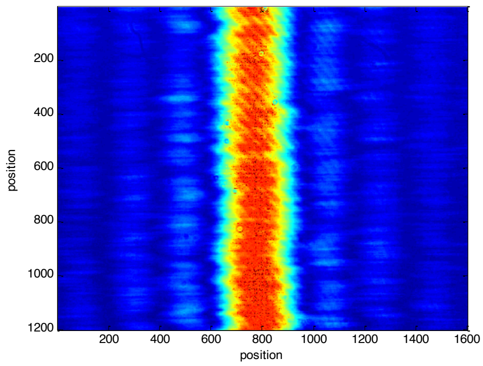
Figure 31 The slit has to be hold in front of the source as shown. Position has to be carefully adjusted to get the most uniform diffraction pattern.

- Align the slit
- Adjust intensity to avoid saturation with the polarizer
- Take images with the slit aligned vertically

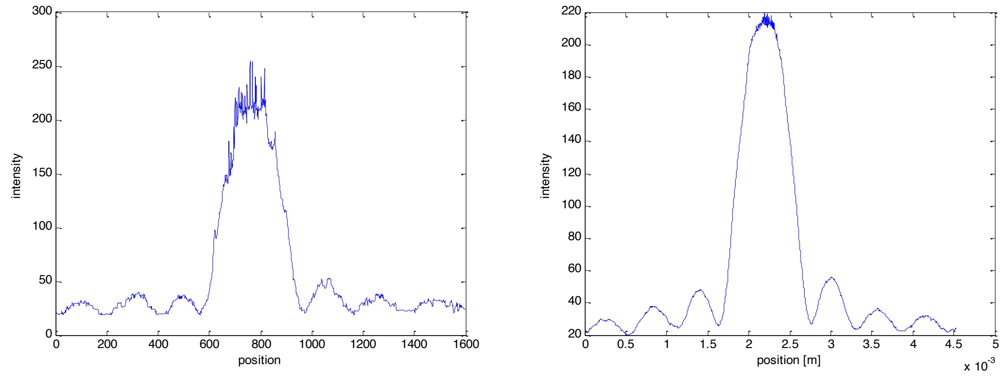
An example of the measurement is shown below. Make a measurement and plot the images as shown below:



A)



B)



C) D)

Figure 32 Evaluation of the intensity pattern for slit diffraction. A) Image, B) 2 d intensity as seen in Matlab, C) Single line intensity and D) averaged line intensity.

You can use the script you “AVG_line_ROI_red_single.m” to create the plots above.

Take care that you have a well aligned (vertical!!!) diffraction pattern that allow averaging procedures and avoid saturation.

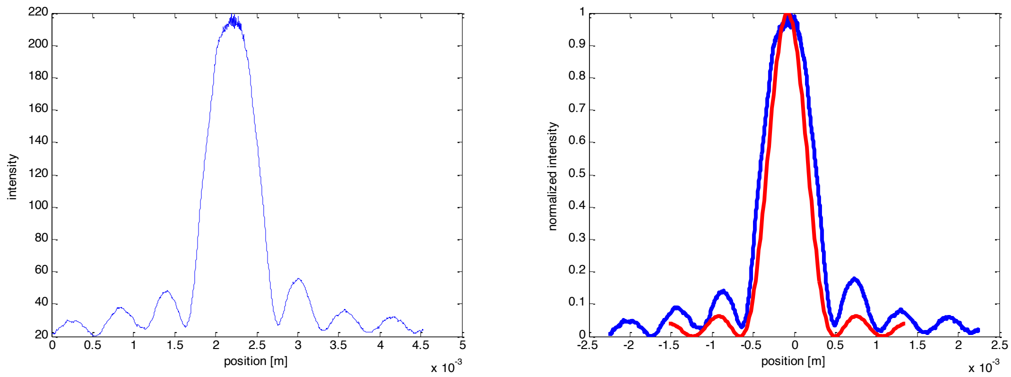


Figure 33 Example plot with matching parameters between measurement and simulation. Such plots could be created with a modified script (AVG_line_ROI_red_compare.m). Please see appendix for details.

To assure the validity of the Fraunhofer approximation the calculation of the Fresnel number has to be done with the following formula

$$N_F = \frac{w^2}{\lambda z}$$

For the slit we can use the analytical diffraction equation to control our results. The zeros of intensity for the **first order** minima of a slit are given by the equation

$$\sin\theta = \frac{\lambda}{2w}$$

where θ is the diffraction angle, 2w is the slit width and λ is the wavelength.
In the experiment the distance from the slit to the camera is z and with position. On our camera image the position x of the minimum are given by the relation

$$\tan\theta = \frac{x}{z}$$

and with the diffraction angle θ from above one finds

$$x = z\tan\theta = z\tan\left(\arcsin\left(\frac{\lambda}{2w}\right)\right)$$

TO BE DONE FOR THE REPORT: Take a picture of the diffraction pattern for the 100 micron slit. Measure the distance of the slit to the camera chip (usually about 100mm). Simulate the diffraction pattern with Fraunhofer diffraction, Fourier transform and Matlab and plot it aside to the measured ones. Calculate the Fresnel number for your experiment and write your simulation parameters in the report. With the help of the Fresnel number judge if the Fraunhofer approximation is valid in our case. Present diffraction pattern line plots and compare the position of minimum intensity for measurement and theory. Compare the first minimum of measured diffraction curve with theory. Calculate the value for the first minimum and compare them with your measurement and simulation!

4.5 Grating diffraction and Fourier transform

Grating diffraction is observed by illuminating the grating with a plane wave and focusing the light onto the detector. For the measurement we are using the original camera objective. One uses collimated laser light that leads to extremely small spot sizes. Saturation is an issue and needs to be carefully controlled.
The sheet polarizer is used to adjust the light intensity. It is placed in front of the collimation lens. By rotating the source in its mount, the light intensity is varied. In addition, the settings of the camera are changed to prevent overexposure. We are looking for the smallest focus size that can be achieved for point sources located away from the detector.

The laser diode is itself a polarized source. Sources are polarized in a certain direction and rotation of the source with respect to the sheet polarizer allows intensity adjustment (cosine square law when rotated by a certain angle Θ). To measure the focalisation properties follow the following procedure:

- Mount the objective in front of the camera

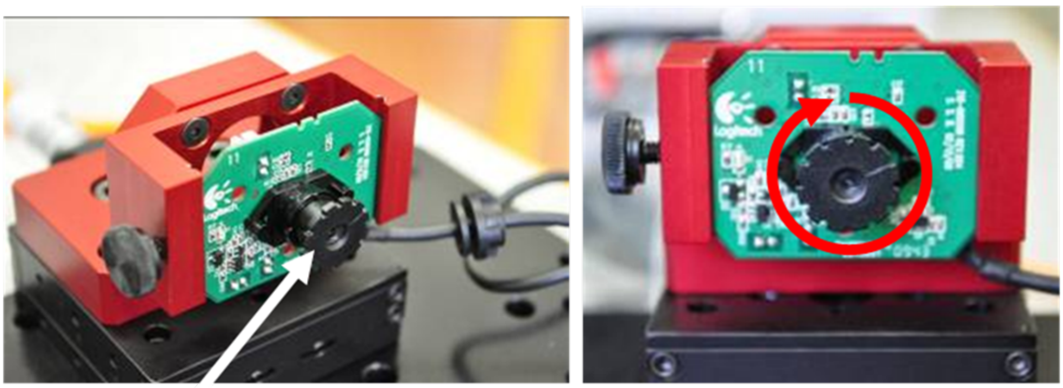


Figure 36 Mount the objective. Rotating allows to focus the collimated source to the smallest spot size

For convenience we will not use the collimated laser beam as before but work with a situation where the laser is put as far as possible and imaged onto the detector by the objective.

- **REMOVE THE LENS CAP (!) from the source.**

- Adjust the intensity again using the sheet polarizer and rotate the source to lower the intensity
- Adjust the exposure settings (low gain if possible, play with exposure time)
- It is very important to avoid saturation because it falsifies the measurement. Note that this is not always possible

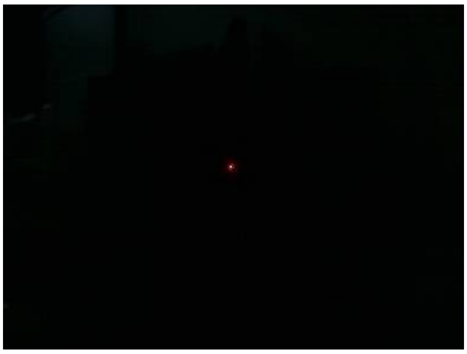
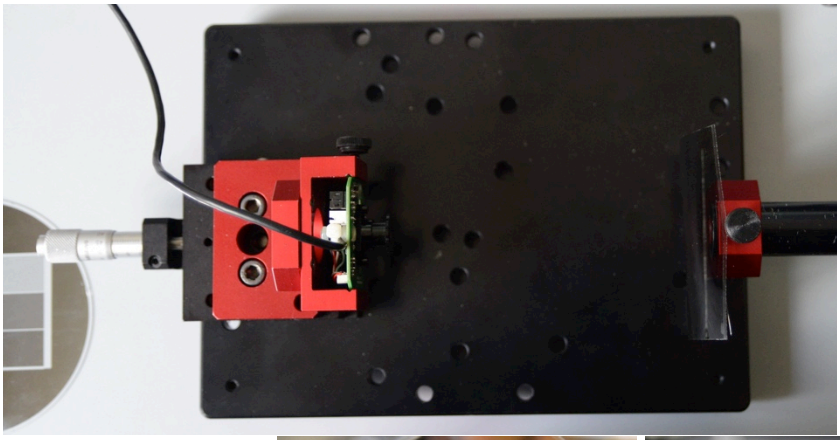


Figure 37. Example images for the smallest spot size of the laser. The intensity has to be adapted depending on the diffractive object. Avoid saturation (not always possible!).

Now the diffractive structures can be applied. Here we will **put them directly in front of the camera objective.**



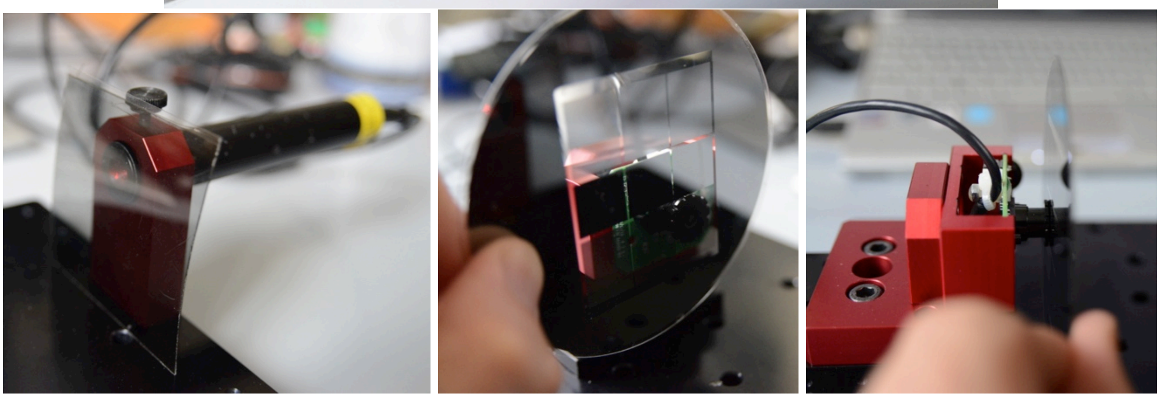


Figure 38. Setup with sheet polarizer and the application of the diffractive structure in front of the objective.

There are several structures implemented on the sample. We want to study the influence of the gratings period and aspect ratio on the diffraction pattern and compare this with simulations.

Structure type	Period (µm)	Slit width (µm)
Line grating	20	10
Line grating	40	10
Line grating	60	10
Line grating	60	20
Line grating	60	30
2d grating - squared	50	10
2d grating - squared	50	25
2d grating - triangle	50	25
2d grating – circle	50	25

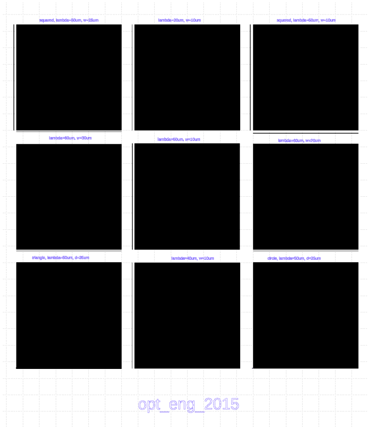


Figure 39 Diffractive structures present on the high resolution wafer used in our experiment. A larger print out is attached at the end.

In our experiment we will put the diffractive structure in front of the lens. Because of the lens the formula describing the diffraction properties for Fraunhofer diffraction will change. One finds

$$U_2(x,y)=\frac{\exp(jkf)}{j\lambda f}\exp\left[j\frac{k}{2f}\left(x^2+y^2\right)\right]\iint U_1(\xi,\eta)\exp\left[-j\frac{2\pi}{\lambda f}(x\xi+y\eta)\right]d\xi d\eta \quad \text{Eq. 34}$$

where f is the focal length of the lens. The equation is similar to expression Eq. (23) above but the definition of the spatial frequency has changed. Instead of Eq. 24 one finds now

$$f_x=\frac{x}{\lambda f} \qquad f_y=\frac{y}{\lambda f} \qquad \text{Eq. 35}$$

So everything is expressed in terms of the focal lengths.

This will have consequences on the calculation of the position of the diffraction spots on the detector.

Do the following:

- Determine the centre peak position by taking an image without structure.
- Introduce the diffractive structure **in front of the objective**
- Adjust intensity
- Take images for all gratings and name them correctly
- Take care that the images are aligned horizontally for further processing

The resulting images are plotted below.

Figure 40 Diffraction patterns of gratings of different variants that are found on the slide. From top: Λ=20μm, a=5 μm, Λ=40μm, a=5 μm, Λ=60μm, a=5 μm, Λ=60μm, a=20 μm, Λ=60μm, a=30 μm. The pictures need to be aligned perfectly horizontal which is not the case in this example.

Diffraction pattern simulation

Next we will compare a simulated diffraction pattern with measurements. The script you have developed serves as a basis. We want to visualize the effect of the slit/rectangle and the grating period and this is best done by plotting two graphs in each simulation plot – one for the slit alone and one for the grating. You need to modify your script to generate a grating.
The number of samples has to be increased for correct sampling. Use 10000 and change M to

```
M = 10000; %number of samples
```

We will use the "square" function already implemented in Matlab to do the work for us.
`x = square(ζ)` generates a square wave with period 2π for the elements of vector ζ. `square(ζ)` is similar to `sin(ζ)`, but creates a square wave with peaks of ±1 instead of a sine wave.
`x = square(ζ,duty)` generates a square wave with specified duty cycle, `duty`, which is a number between 0 and 100. The *duty cycle* is the percent of the period in which the signal is positive.

To generate a grating you can use the following sequence

```
p = 60e-6; %grating_period

dutycycle=2*w/p*100; %grating width translated in dutycycle 0..100
f = rect(x/(2*w)); %original signal vector of the rectangle
f_grating = (square(x/p*2*pi+(pi*dutycycle/100),dutycycle)+1)/2;
%normalized signal vector for the grating (values between 0 and 1)
```

Here we already implemented the dutycycle and correct period calculation by multiplying with 2 π and shifting to have a well centred grating.
You must add the following lines to plot the grating diffraction pattern

After	`f0=fftshift(f); %shift f`
Add	`f0_grating=fftshift(f_grating); %shift f`
After	`F0=fft(f0)*dx; %FFT and scale`
Add	`F0_grating=fft(f0_grating)*dx; %FFT and scale`
After	`F=fftshift(F0); %center F`
Add	`F_grating=fftshift(F0_grating); %center F`

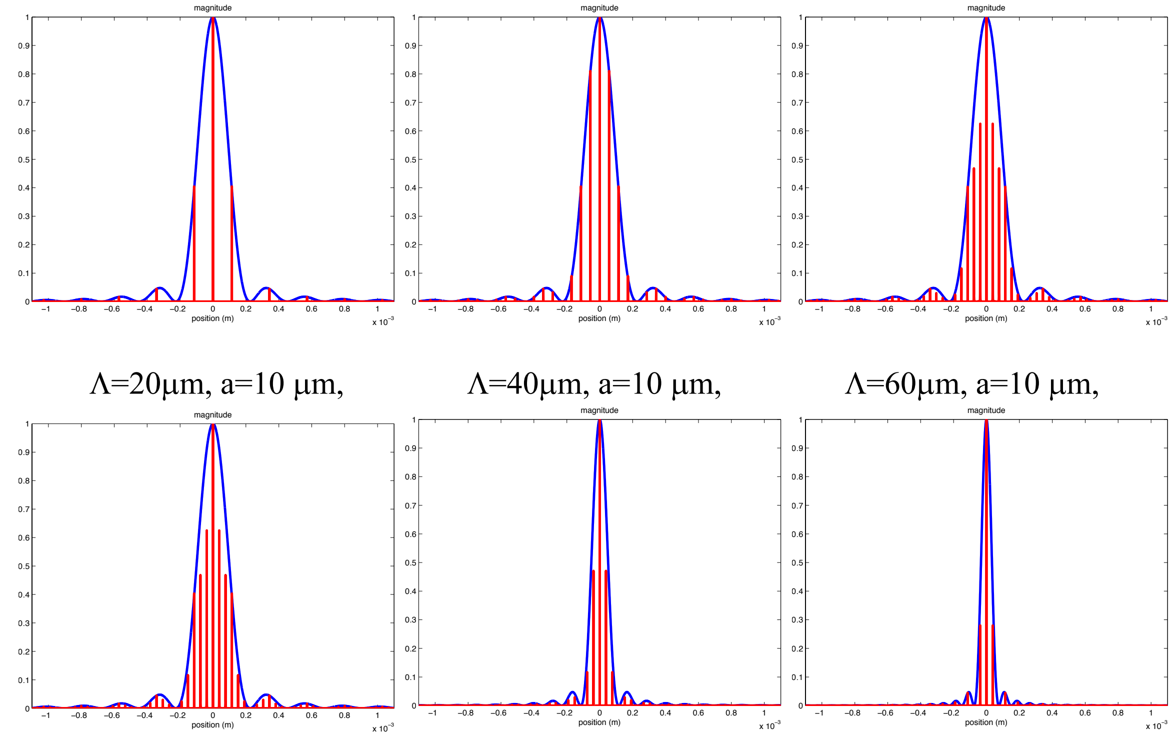
At the end you need to plot both curves after correct scaling. Replace the original instructions for figure 9 by:

```
figure(9)
lambda=0.635e-6;%wavelength
focal_lengths= 3.6e-3; %focal lengths
position_sim = fx*focal_lengths*lambda;
plot(position_sim ,(abs(F)/max(abs(F))).^2,'b-
',position_sim,(abs(F_grating)/max(abs(F_grating))).^2,'r-','linewidth',3);
%plot magnitude after propagation
title('intensity');
xlabel('position (m)');
axis([-L/4 L/4 0 1])
```

A normalization of the intensity (F^2) was done to allow easy comparison. The last thing is to change the parameters to model correctly our geometry.

```
w = 5e-6; %rectangle half-width (m)
L = 4.4e-3; %vector side length (m)
```

Note that w is the **rectangle half width**, that means the parameter a in Fig. 40 is 2w. The image below shows simulations of selected grating configurations.



Λ=20μm, a=10 μm,

Λ=40μm, a=10 μm,

Λ=60μm, a=10 μm,

Λ=60μm, a=10 μm,

Λ=60μm, a=20 μm,

Λ=60μm, a=30 μm

Figure 41. Simulated diffraction patterns for different situations. In Blue the diffraction of a single rectangle is shown. In red the grating diffraction is given. On the top the grating period is changed and below the slit width is modified.

Next the simulation can be compared with experiment. To do so a line plot of the measurements shown in Figure 40 should be used. The script “AVG_line_ROI_red_triple.m” will help you. It plots an image and three subplots with the diffraction patterns. You need to change to parameters in the script to your current situation. Determine the centre of you plots by introducing the name of your files and utilizing the zoom and curser function in MATLAB figure window.

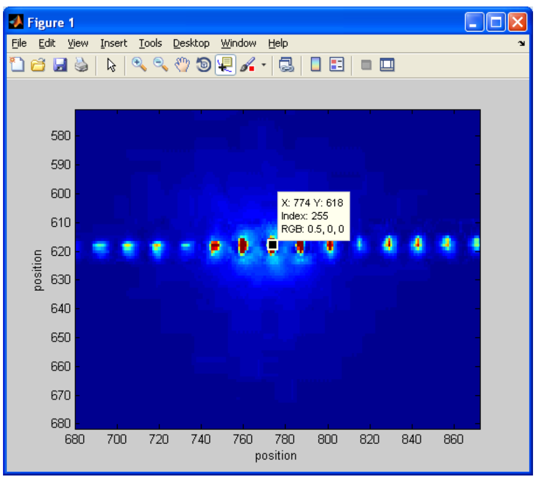


Figure 42. The centre spot of the diffraction image is best found visually. In this example it is `x_center = 773;y_center = 617;`.

Change the script at the corresponding lines and run it again. You will get results similar to the plots below.

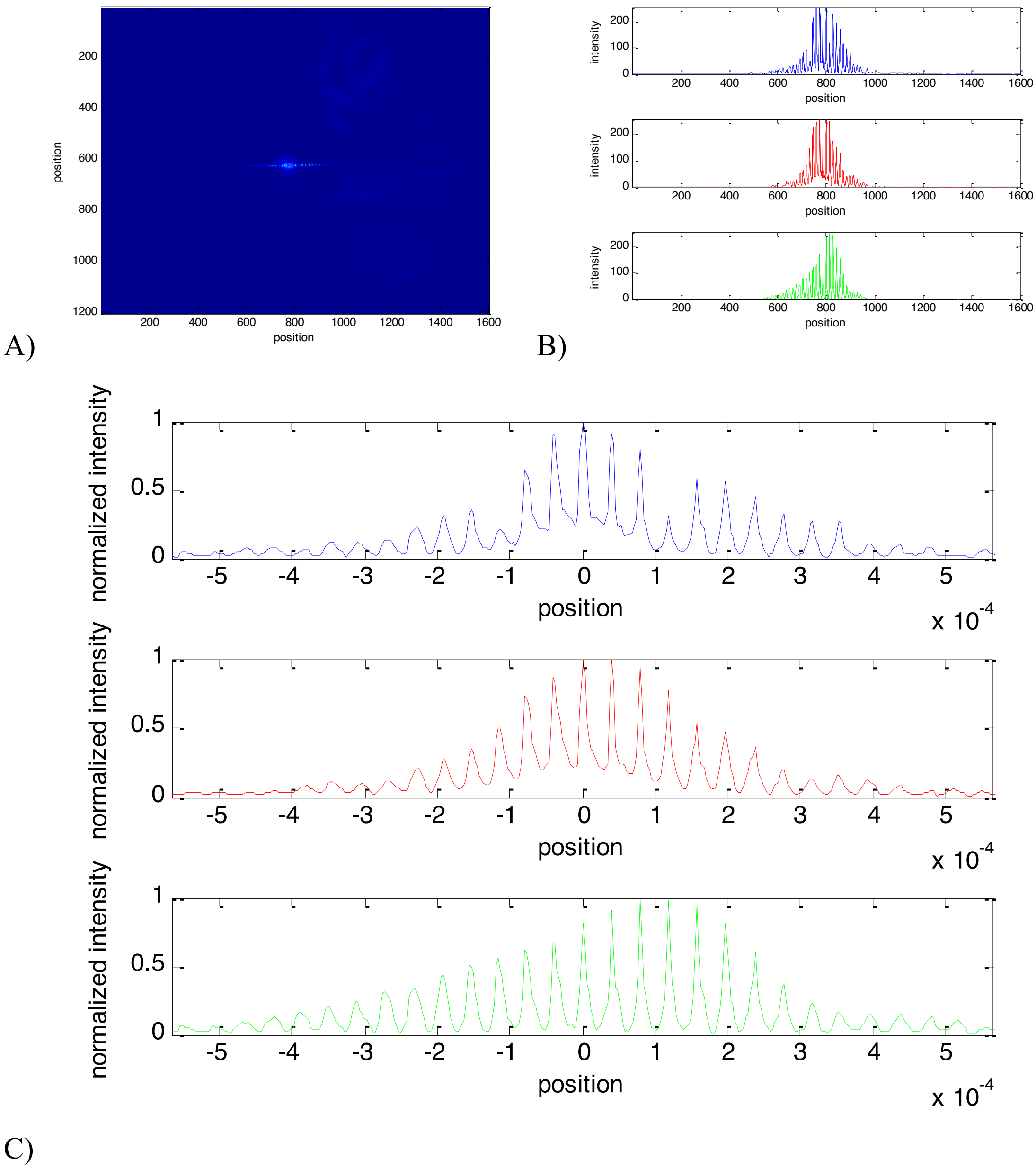
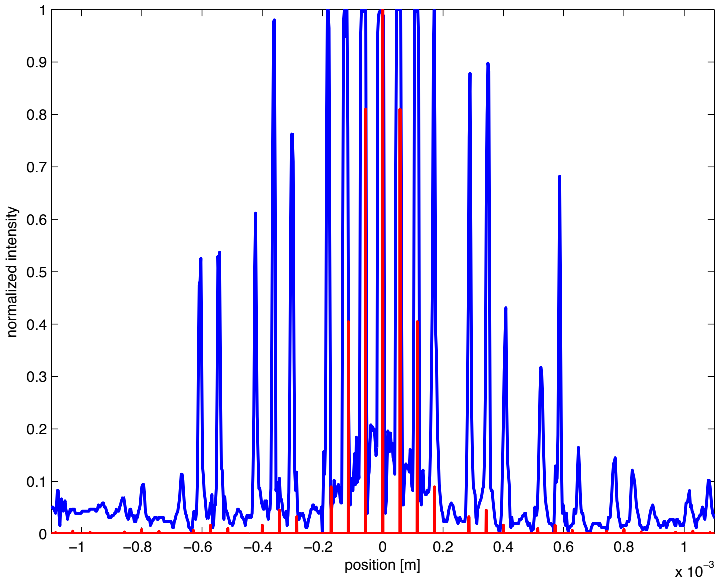
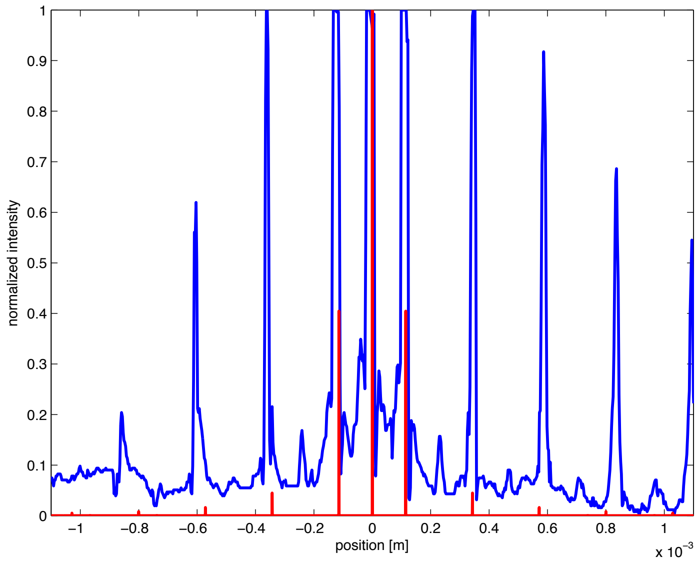


Figure 43. Measured diffraction plots. A) Intensity diffraction pattern. B) Original single line plots. C) Scaled line plots: From top: Λ=60μm and a=20 μm, Λ=60μm and a=10 μm, Λ=60μm and a=5 μm.

As one can see in Fig. 43 it is very difficult to see special features in such plots although the diffraction efficiency of different orders is well distinguishable by the naked eye. This is mainly an effect of the low dynamic of our camera and the quality of the gratings that are printed on a transparency. Such transparencies cannot be considered as optically good substrates. As the main result we want to plot the simulation and the diffraction intensity measurements and simulation in one plot for two examples: The grating with Λ=60μm and a=20 μm and the grating with Λ=20μm and a=5 μm. You can use the script "fft_example_1d_grating_compare.m" to do so.



Λ=40µm and a=10 µm



Λ=20µm and a=5 µm

Figure 44. Comparison of measured and simulated plots for two different gratings. In red is the simulated diffraction pattern and in blue the measurement

Figure 44 shows a comparison between two different gratings of different period. The periodicity is well reproduced.

TO BE DONE FOR THE REPORT: Present an image of the measured diffracted intensity for each grating similar to that given in Figure 40. Simulate the diffraction pattern for the grating geometries and show your own simulation results similar to this in Figure 41. Explain what happens if you change the period of the grating at constant rectangle (slit) width. Explain what happens if you change the rectangle (slit) width and keep the period of the grating constant. Compare the measurements with simulations for two grating configurations and plot a Figure similar to Figure 44. Discuss briefly features like spacing and height of diffraction peaks.

5. Summary of tasks of the experimental work

4.3 Diffraction with a collimated beam setup and verification

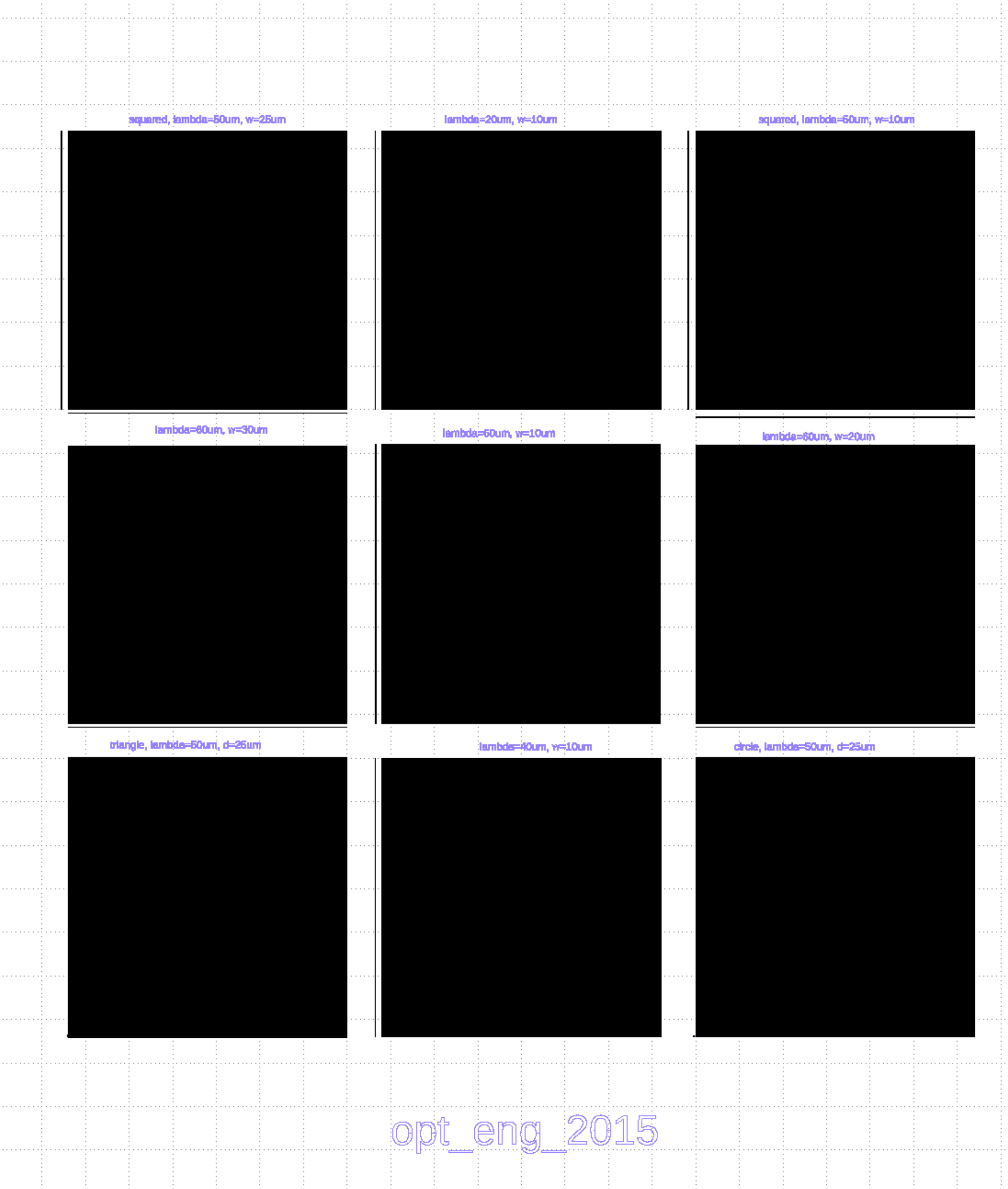
Take pictures of the diffraction pattern for the 100 micron pinhole and 100 micron rectangle. Measure the distance from the diffracting object to the camera chip (usually 100mm). Calculate the theoretical 2 dimensional diffraction pattern with Matlab and plot it aside to the measured ones. Write your simulation parameters in the report.

4.4. Fraunhofer diffraction and application of the Fourier transform

Take a picture of the diffraction pattern for the 100 micron slit. Measure the distance of the slit to the camera chip (usually about 100mm). Simulate the diffraction pattern with Fraunhofer diffraction, Fourier transform and Matlab and plot it aside to the measured ones. Calculate the Fresnel number for your experiment and write your simulation parameters in the report. With the help of the Fresnel number judge if the Fraunhofer approximation is valid in our case. Present diffraction pattern line plots and compare the position of minimum intensity for measurement and theory. Compare the first minimum of measured diffraction curve with theory. Calculate the value for the first minimum and compare them with your measurement and simulation!

4.5 Grating diffraction and Fourier transform

Present an image of the measured diffracted intensity for each grating similar to that given in Figure 40. Simulate the diffraction pattern for the grating geometries and show your own simulation results similar to this in Figure 41. Explain what happens if you change the period of the grating at constant rectangle (slit) width. Explain what happens if you change the rectangle (slit) width and keep the period of the grating constant. Compare the measurements with simulations for two grating configurations and plot a Figure similar to Figure 44. Discuss briefly features like spacing and height of diffraction peaks.


squared, lambda=50um, w=25um
lambda=20um, w=10um
squared, lambda=60um, w=10um
lambda=60um, w=30um
lambda=60um, w=10um
lambda=60um, w=20um
triangle, lambda=50um, d=25um
lambda=40um, w=10um
circle, lambda=50um, d=25um
opt_eng_2015

EPFL

MATLAB Scripts

AVG_line_ROI_red_single.m

```
% Read image 1
I_1 = imread('picture 3.jpg');

% select a channel (here red)
Red_1 = I_1(:,:,1);

% format conversion
Red_1 = double(Red_1);

%visiualize the image to define region of interest

imagesc(Red_1)
xlabel('position')
ylabel('position')


% select a line of interest ROI (1 x 1600), here center line at y = 600,
% x is from 1 to 1600)
y_center = 620,
CenterLine = Red_1(y_center:y_center,1:1600);
%plot a single line

figure
plot(CenterLine);
xlabel('position')
ylabel('intensity')

%average over several lines
% select a region of interest ROI with several number of lines N (N x
% 1600), x goes from 1 to 1600, averages will by taken over Roi_y from
% y_center-ROI_y to y_center+ROI_y!

ROI_y = 50,
ROI_Red_1  = Red_1((y_center-ROI_y):(y_center+ROI_y),1:1600);

%caculate the mean of the RIO over several rows
N_Avg_Red_1 = mean (ROI_Red_1);
position=(1:1:1600)*2.835e-6;

%plot the averaged line
figure
plot(position,N_Avg_Red_1);
xlabel('position [m]')
ylabel('intensity')
```

AVG_line_ROI_red_compare.m

You need to run the script that defines the simulation parameters like `position_sim` and `F` before you can execute the script below!

You need also to move the scale to match the center points.

```
% Read image 1
I_1 = imread('picture 3.jpg');
% select a channel (here red)
Red_1 = I_1(:,:,1);
% format conversion
Red_1 = double(Red_1);

%visiualize the image to define region of interest

imagesc(Red_1)
xlabel('position')
ylabel('position')


% select a line of interest ROI (1 x 1600), here center line at y = 600,
% x is from 1 to 1600)
y_center = 620;
CenterLine = Red_1(y_center:y_center,1:1600);
%plot a single line

%figure
%plot(CenterLine);
%xlabel('position')
%ylabel('intensity')

%average over several lines
% select a region of interest ROI with several number of lines N (N x
% 1600), x goes from 1 to 1600, averages will by taken over Roi_y from
% y_center-ROI_y to y_center+ROI_y!

ROI_y = 50;
ROI_Red_1  = Red_1((y_center-ROI_y):(y_center+ROI_y),1:1600);

%caculate the mean of the RIO over several rows
N_Avg_Red_1 = mean (ROI_Red_1);
position=(-800:1:799)*2.835e-6;

%plot the averaged line
figure
plot(position,N_Avg_Red_1/max(N_Avg_Red_1),'b-',position_sim-8e-
5,(abs(F)/max(abs(F))).^2,'r-','LineWidth',3);
xlabel('position [m]')
ylabel('normalized intensity')
```

AVG_line_ROI_red_triple.m

```
% Read images
I_1 = imread('picture 183.jpg');
I_2 = imread('picture 184.jpg');
I_3 = imread('picture 185.jpg');

% select a channel (here red)
Red_1 = I_1(:,:,1);
Red_2 = I_2(:,:,1);
Red_3 = I_3(:,:,1);

% format conversion
Red_1 = double(Red_1);
Red_2 = double(Red_2);
Red_3 = double(Red_3);

%visiualize the image to define region of interest

figure
imagesc(Red_1)
xlabel('position')
ylabel('position')
%figure
%imagesc(Red_2)
%xlabel('position')
%ylabel('position')
%figure
%imagesc(Red_3)
%xlabel('position')
%ylabel('position')


% select a line of interest ROI (1 x 1600), here center line at y = 600,
% x is from 1 to 1600)
x_center = 773;
y_center = 617;
CenterLine_1 = Red_1(y_center:y_center,1:1600);
CenterLine_2 = Red_2(y_center:y_center,1:1600);
CenterLine_3 = Red_3(y_center:y_center,1:1600);
%plot a single line

figure
subplot(311);
plot(1:1:1600,CenterLine_1,'b-');
xlabel('position')
ylabel('intensity')
axis([1 1600 0 255]);
subplot(312);
plot(1:1:1600,CenterLine_2,'r-');
xlabel('position')
ylabel('intensity')
axis([1 1600 0 255]);
subplot(313);
plot(1:1:1600,CenterLine_3,'g-');
xlabel('position')
ylabel('intensity')
axis([1 1600 0 255]);

%average over several lines
```

```matlab
% select a region of interest ROI with several number of lines N (N x
% 1600), x goes from 1 to 1600, averages will by taken over Roi_y from
% y_center-ROI_y to y_center+ROI_y!

ROI_y = 5;
ROI_Red_1  = Red_1((y_center-ROI_y):(y_center+ROI_y),1:1600);
ROI_Red_2  = Red_2((y_center-ROI_y):(y_center+ROI_y),1:1600);
ROI_Red_3  = Red_3((y_center-ROI_y):(y_center+ROI_y),1:1600);

%caculate the mean of the RIO over several rows
N_Avg_Red_1 = mean (ROI_Red_1)/max(mean (ROI_Red_1));
N_Avg_Red_2 = mean (ROI_Red_2)/max(mean (ROI_Red_2));
N_Avg_Red_3 = mean (ROI_Red_3)/max(mean (ROI_Red_3));

position=(1:1:1600)*2.835e-6-x_center*2.835e-6;
plot_width=200*2.835e-6;
%plot the averaged line

figure
subplot(311);
plot(position,N_Avg_Red_1,'b-');
xlabel('position')
ylabel('normalized intensity')
axis([(-plot_width) (+plot_width) 0 1]);
subplot(312);
plot(position,N_Avg_Red_2,'r-');
xlabel('position')
ylabel('normalized intensity')
axis([(-plot_width) (+plot_width) 0 1]);
subplot(313);
plot(position,N_Avg_Red_3,'g-');
xlabel('position')
ylabel('normalized intensity')
axis([(-plot_width) (+plot_width) 0 1]);
```

fft_example_1d_grating_compare.m

```
% fft_example
% simulation of the diffraction pattern
w = 10e-6; %rectangle half-width (m)
L = 4.4e-3; %vector side length (m)
M = 10000; %number of samples
dx = L/M; %sample interval (m)
x = -L/2:dx:L/2-dx; %coordinate vector
p = 60e-6; %grating_period
dutycycle=2*w/p*100; %grating width translated in dutycycle 0..100
f = rect(x/(2*w)); %original signal vector of the rectangle
f_grating = (square(x/p*2*pi+(pi*dutycycle/100),dutycycle)+1)/2;
%normalized signal vector for the grating (values between 0 and 1)
f0=fftshift(f); %shift f
f0_grating=fftshift(f_grating); %shift f
F0=fft(f0)*dx; %FFT and scale
F0_grating=fft(f0_grating)*dx; %FFT and scale
F=fftshift(F0); %center F
F_grating=fftshift(F0_grating); %center F
fx=-1/(2*dx):1/L:1/(2*dx)-(1/L); %freq cords
figure(1)
lambda=0.635e-6;%wavelength
focal_lengths= 3.6e-3; %focal lengths
position_sim = fx*focal_lengths*lambda;
plot(position_sim ,(abs(F)/max(abs(F))).^2,'b-
',position_sim,(abs(F_grating)/max(abs(F_grating))).^2,'r-','linewidth',1);
%plot magnitude after propagation
title('magnitude');
xlabel('position (m)');
axis([-L/4 L/4 0 1])
%preparation of the measured data
I_1 = imread('picture 183.jpg');
Red_1 = I_1(:,:,1);
Red_1 = double(Red_1);
%visiualize the image to define region of interest
figure(2)
imagesc(Red_1)
xlabel('position')
ylabel('position')
%definition of center position for scaling
x_center = 773;
y_center = 617;
CenterLine_1 = Red_1(y_center:y_center,1:1600);
%preparation of a centred scaling of the measurement
position=(1:1:1600)*2.835e-6-x_center*2.835e-6;
figure(3)
plot(position,CenterLine_1/max(CenterLine_1),'b-');
xlabel('position')
ylabel('normalized intensity')
axis([-L/4 L/4 0 1])

%plot measurement and simulation together
figure(4)
plot(position,CenterLine_1/max(CenterLine_1),'b-
',position_sim,(abs(F_grating)/max(abs(F_grating))).^2,'r-','LineWidth',2);
xlabel('position [m]')
ylabel('normalized intensity')
axis([-L/4 L/4 0 1])$
```

Appendix Fraunhofer diffraction and application of the Fourier transform

Creating Vectors

Suppose we want to program a simulation of a 1D spatial rectangle function with a half width of 0.055 m (full width of 0.2 m). This rectangle function will be created in a vector that corresponds to a physical length (side length L) of 1 m. If the number of samples in the vector is 200, then the sample interval -x (or “dx” in the code) is 0.01 m. Open a new New M-file. In the new window, enter the following lines:

```
% fft_example
w = 0.05; %rectangle half-width (m)
L = 1; %vector side length (m)
M = 200; %number of samples
dx = L/M; %sample interval (m)
```

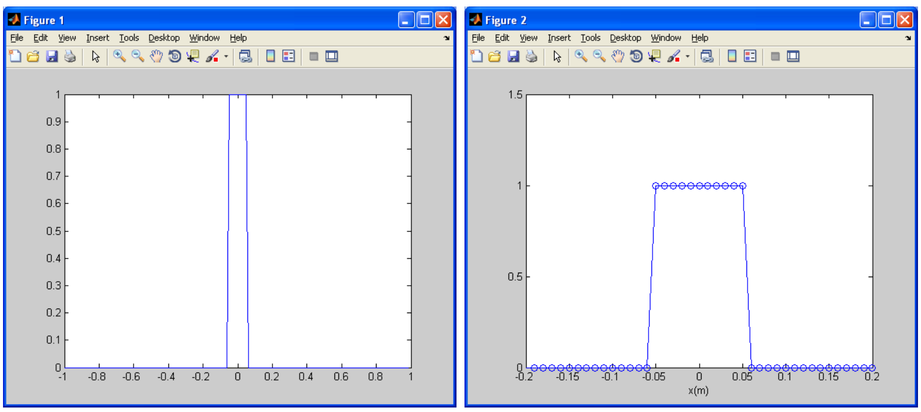
Add the following code:

```
x = -L/2:dx:L/2-dx; %coordinate vector
f = rect(x/(2*w)); %signal vector
```

The “*” indicates multiplication. In this code, the coordinate vector x is created with values ranging from 1 (= –L/2) to 0.99 (= L/2-dx) in steps of 0.01 (dx). The colons separate the range and step size. The subtraction of one step off the high limit results in a vector of 200 samples. The signal vector “f” is created with the help of the **previously defined “rect” function** and the use of the coordinate vector x. Since 2*w is a scalar, the interpretation is that each sample in x is divided by 2*w and the resulting vector is input to the rect function. To display what you have created in a plot of the f values against the x values, add the following lines:

```
figure(1)
plot(x,f); %plot f vs x
```

The “figure(1)” command opens a window labeled “Figure 1” and “plot” makes a plot. Before executing this script, you need to save it to a disk. Save “fft_example_1d”. Don’t use any spaces in the name, as that can be interpreted as a request to link to another function; to execute, click on Run in the Editor Window toolbar (a triangle icon or for older MATLAB versions an arrow beside a document icon). If there are no errors in your code, the plot of f should appear in the Figure 1 window.



A) B)

Figure 29 Left: A rect function example plot. Right Expanded view of sampled rect function.

If something is wrong with the code, you will see a message in the Command Window. The plot should look like Fig. 29A. Go back and edit the last plot command and add “axis” and “xlabel” command lines:

```
figure(2)
plot(x,f,'-o'); %plot f vs x
axis([-0.2 0.2 0 1.5]);
xlabel('x(m)');
```

The added option in the plot command ('-o') places the marker o at each sample in the plot. The axis command sets limits on the values plotted on x and y axes and the xlabel command labels the x-axis. The figure(2) command will open a second window for this new plot. Without this command, the new plot would overwrite the first plot in the Figure 1 window. Note that the plot-modifying statements come after the plot call. Execute the code (this also automatically saves the latest version of the code) and the plot appearing in the Figure 2 window (Fig. 29 B)) more clearly shows the rectangle function. Note the distance across the top of the rectangle corresponds to 0.1 m (11 samples) whereas the base of the rectangle corresponds to 0.12 m (13 samples). The correct interpretation for this sampled rectangle function is a width of 0.11 m, which lies between the two measures. In fact, because the rect function coding generates an odd number of samples, if a half width of 0.05 m were used rather than 0.055 m, the resulting signal vector would be identical to that shown in Fig. 30A, which means there would be a small disparity in the intended width and the digital representation.

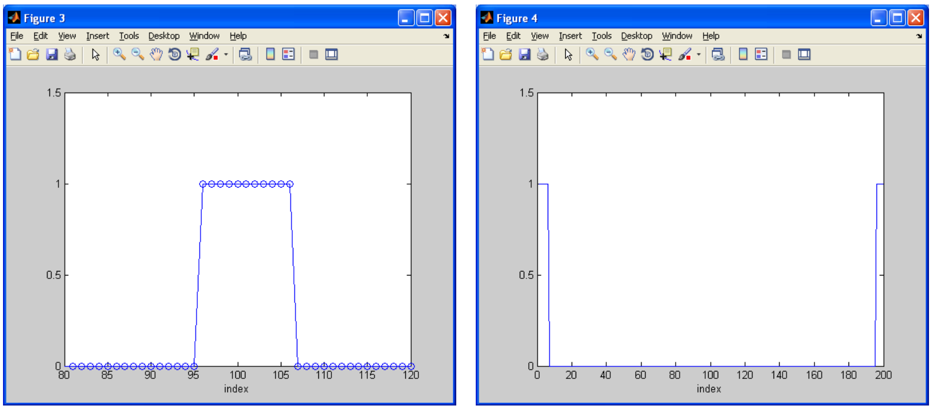
Shift for FFT

Disregarding the coordinate vector x for a moment, in terms of the 200 samples in f, a rectangle function with a width of 11 samples has been created that is centered in the middle of the vector f. This can be visualized by adding the following code:

```
figure(3)
plot(f,'-o');
axis([80 120 0 1.5]);
xlabel('index');
```

With the x argument removed from the plot command, the vector f is plotted as a function of the vector index values. The resulting plot (Fig. 30A) shows the center sample of the rectangle function at index position 101. Centering the rectangle in the middle of the vector allows for easy viewing, but the FFT algorithm expects the zero-coordinate value to be in the first index location. To shift the values for the FFT operation, the “fftshift” command can be applied, which takes the samples of the first half of the vector and swaps them with the samples in the second half. Add the following the code to the program and run the script to get the plot in Fig. 30B.

```
f0=fftshift(f); %shift f
figure(4)
plot(f0);
axis([0 200 0 1.5]);
xlabel('index');
```



A) B)

Figure 30 A) Sampled rect function versus index value, B) Shifted rect function versus index value.

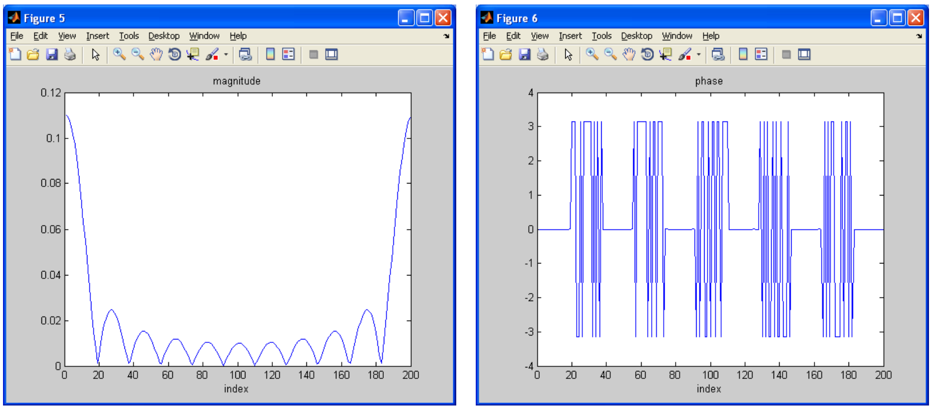
A close look at Fig 4 should convince you that the center sample of the rectangle is at index location 1, and five samples are found to the “right” and the remaining five to the “left,” where the left group is placed at the end of the vector

A few comments about shifting for the FFT: first, it is more straightforward to use an even number of samples M for the vector and center the function of interest at an index of M/2+1. The fftshift function will then shift the M/2+1 sample to index 1. In our example M = 200 and the definition of the x vector causes the rectangle function to be centered at the M/2+1 position (101). More care needs to be taken when shifting vectors that contain an odd number of samples. For example, if the fftshift function is used with an odd number of samples, then the ifftshift function should be used to undo the shift. For an even number of samples, the fftshift function works both forward and backward. A second comment is that without the shift operation the FFT algorithm generates a transform for a function that is translated from the zero position, which means a linear phase term will be present in the result (shift theorem!).

Computing the FFT and Displaying Results

To compute the FFT of the vector “f0,” add the following below the last piece of code:

```
F0=fft(f0)*dx; %FFT and scale
figure(5)
plot(abs(F0)); %plot magnitude
title('magnitude');
xlabel('index');
figure(6)
plot(angle(F0)); %plot phase
title('phase');
xlabel('index');
```



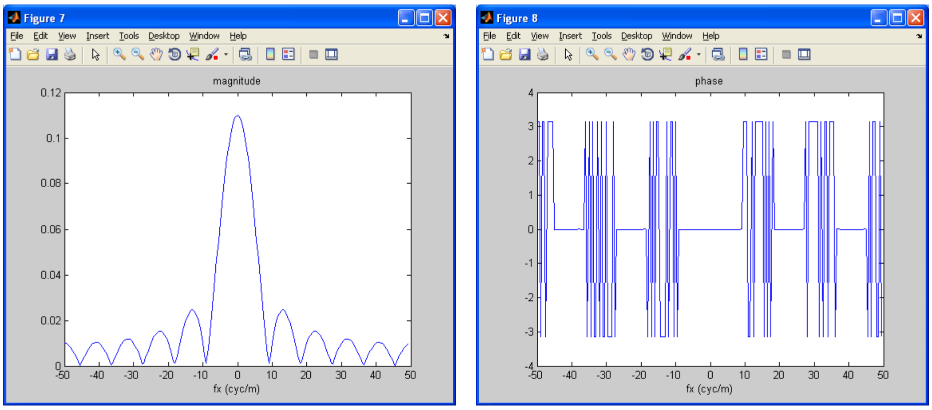
A) B)

Figure 31 A) Magnitude and (B) phase plots for FFT result versus index

Here, the 1D FFT algorithm in MATLAB is used. A capital letter is used for the frequency domain vector. Multiplying the result by the sample spacing dx is necessary to correctly approximate the analytic Fourier transform integral. Since each sample in F0 contains two pieces of information (the real part and the imaginary part); or alternatively, the magnitude and the phase, two plots can be used to display this result. The “abs” command takes the absolute value (magnitude) of the samples in F0 and the “angle” command extracts the phase of the samples, with output values ranging from (to (????-pi to pi???. Plot titles and x-axis labels have been included in this code. Run the script and the plots should look like those in Fig. 31.

The sinc function nature of the magnitude result in Fig. 31 is obvious where the peak of the sinc is centered at index 1. However, note that the samples are not located exactly where the “zeros” of the magnitude would occur. Thus, the valleys shown in the curve do not necessarily appear to reach zero. The phase plot may look curious, but there are essentially three phase values in the plot: 0, -π and π. However, - π and π represent the same value in a modulo 2 π format, so the sharp transitions in the plot between - π and π are not particularly important, and the phase is effectively constant in these places. These sharp 2 π transitions occur because of slight numerical differences between samples. With this interpretation, you should understand that the important phase transition is from 0 to π (or- π), which occurs about every 18 points. Comparing the phase plot with the magnitude plot, it is apparent that the π transition occurs at the magnitude zeros. Furthermore, a phase value of π is equivalent to placing a minus sign on the magnitude value. Combining all of this information, the magnitude and phase plots of Fig. 31 represent a real-valued sinc function where the values in the main lobe are positive, the values in the first lobe are negative, the second lobe values are positive, and so on. In this case the real-valued sinc function could have simply been displayed on one plot; but, in general, Fourier transform results are complex. Once again for display reasons, it is helpful to center the FFT result in the vector. In addition, the spatial frequency coordinates need to be determined. Add the following to your script (cutting and pasting from the earlier code can help accomplish this quickly):

```
F=fftshift(F0); %center F
fx=-1/(2*dx):1/L:1/(2*dx)-(1/L); %freq cords
figure(7)
plot(fx,abs(F)); %plot magnitude
title('magnitude');
xlabel('fx (cyc/m)');
figure(8)
plot(fx,angle(F)); %plot phase
title('phase');
xlabel('fx (cyc/m)');
```



A) B)

Figure 32 (A) Magnitude and (B) phase plots for centered FFT result.

Running the script generates the plots shown in Fig. 32, where the sampled Fourier magnitude and phase are centered, and the frequency axis is scaled appropriately.

Uniform sampling and square grids are often used in practice. Here, the parameters for the rectangle physical size, the vector side length, the number of samples, and sample interval are defined.

$$L = M\,\Delta x \qquad \text{Eq. 30}$$

The coordinates of the samples along one dimension can be described as

$$x \rightarrow \left[-\frac{L}{2} : \Delta x : \frac{L}{2} - \Delta x\right] \qquad \text{Eq. 31}$$

Assuming a FFT relationship between the spatial and spectral domains, then the following is derived:

$$f_x \rightarrow \left[-\frac{1}{2\Delta x} : \frac{1}{L} : \frac{1}{2\Delta x} - \frac{1}{L}\right] \qquad \text{Eq. 31}$$

which indicates that the spatial frequency coordinates range from $-1/(2\Delta x)$ to $1/(2\Delta x)-1/L$ in steps of $\Delta fx = 1/L$.

In order to compare the simulation and measurements it is necessary to "translate" the frequencies fx into the real space. As described above the spatial frequency f_x scales with the distance to the observation plane z and the wavelengths as

$$f_x = \frac{x}{\lambda z} \qquad \text{Eq. 24}$$

This is valid for free space propagation. A look at our definition in Matlab reveals that the following expression can be used to calculate find the x values on our detector

$$x = f_x \lambda z \qquad \text{Eq. 32}$$

Again z is the propagation distance (distance between grating and detector) and λ is the operating wavelengths

To implement this into the code the following pieces code should be added:

```
figure(9)
lambda=0.635e-6;%wavelength
z=50; %prop distance
lz=lambda*z;
```

```
position_sim = fx*lz;
plot(position_sim ,abs(F)^2); %plot magnitude after propagation
title('magnitude');
xlabel('position (m)');
```