Backward discrete wavefield propagation modeling as an inverse problem: toward perfect reconstruction of wavefield distributions

Vladimir Katkovnik*, Artem Migukin, Jaakko Astola

Signal Processing Institute, University of Technology of Tampere,

P. O. Box 553, Tampere, Finland.

E-mails: {vladimir.katkov, artem.migukin, jaakko.astola}@tut.fi.

*Corresponding author

We consider a reconstruction of a wavefield distribution in an input/object plane from data in an output/diffraction (sensor) plane. A contribution of this paper concerns a digital modeling both for the forward and backward wavefield propagation. A novel algebraic matrix form of the discrete diffraction transform ($\mathcal{DDT}$) originated in [1] is proposed for the forward modeling which is aliasing free and precise for pixel-wise invariant object and sensor plane distributions. This "matrix $\mathcal{DDT}$ " is a base for formalization of the object wavefield reconstruction (backward propagation) as an inverse problem. The transfer matrices of the matrix $\mathcal{DDT}$ are used for calculations as well as for the analysis of conditions when the perfect reconstruction of the object wavefield distribution is possible. We show by simulation that the developed inverse propagation algorithm demonstrates an improved accuracy as compared with the standard convolutional and discrete Fresnel transform algorithms. © 2009 Optical Society of America

OCIS codes: 070.2025, 100.3010, 100.3190

1. Introduction

A wavefield reconstruction from intensity and phase measurements is one of the basic problems in digital wavefield imaging and holography [2]. In a typical holography scenario one distinguishes input/object and output/sensor planes (Fig.1). The object plane is a source of light radiation/reflection propagating along the optical axis. The sensor plane is parallel to the object plane with a distance z between the planes.

A contribution of this paper concerns a digital modeling the forward and backward wavefield propagation. The discrete diffraction transform ($\mathcal{DDT}$) proposed in [1] links discrete pixelated values of the object and sensor distributions. It is the so-called *discrete-to-discrete* modeling which is aliasing free and accurate for a pixel-wise invariant object distribution and pixelated sensor. In this paper we generalize and develop this model in two important aspects. First, the pixels in the object and sensor planes can be of different size. Second, instead of the frequency domain $\mathcal{DDT}$ ($\mathcal{F}-\mathcal{DDT}$) exploited in [1] we introduce an algebraic approach based on the matrix transform of the wavefield distributions. This novel matrix $\mathcal{DDT}$ is also aliasing free and accurate for the pixelated object and diffraction plane distributions. Furthermore, this matrix $\mathcal{DDT}$ does not require double-size image calculations as it is in $\mathcal{F}-\mathcal{DDT}$. The reconstruction of the object distribution from a distribution given in the sensor plane is formulated as an inverse problem. Depending on the pixel size and the distance between the object and sensor planes $\mathcal{DDT}$ matrices can become very ill-conditioned what makes the reconstruction of the object distribution difficult or even impossible. The ability of the perfect/good quality reconstruction is well characterized by the rank and conditioning number of the transform matrices of the introduced matrix $\mathcal{DDT}$. The forward matrix transform modeling is a natural and very productive tool to study limitations of the wavefield reconstruction and to develop novel effective algorithms.

2. Discretization of backward wavefield propagation

In this section we consider a wavefield reconstruction based on a discretization of the backward propagation integral operator. In this way we arrive to the well known standard techniques as well as introduced some novel ones.

2.A. *Integral models*

Let $u_d(y,x)$ be a complex-valued $2D$ wavefield defined in the sensor plane $z=d$ of the $3D$ space (x,y,z) as a function of the lateral coordinates x and y (see Fig. 1). According to the scalar diffraction theory there is a linear operator which links this sensor wavefield distribution with the object wavefield $u_0(y,x)$ in the plane $z=0$ as $u_d(y,x)=\mathcal{D}_d\{u_0\}$, where $\mathcal{D}_d$ stays for a diffraction operator $\mathcal{D}_z$ with a distance parameter $z=d$. The thorough theory of this operator representation can be found in [3], [4].

The diffraction operator can be given as the convolution

$$\begin{aligned} u_z(y,x) &= \mathcal{D}_z\{u_0\} \triangleq \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} g_z(y-\eta,x-\xi)u_0(\eta,\xi)d\eta d\xi, \qquad (1)\\ (y,x) &\in R^2, \end{aligned}$$

where the kernel g_z is shift invariant defined by the first Raylegh-Sommerfeld solution of the Maxwell-Helmholtz equation [4]

$$g_z(y,x)=z\frac{\exp(j2\pi r/\lambda)}{j\lambda\cdot r^2},\ r=\sqrt{x^2+y^2+z^2},\ z\gg\lambda, \qquad (2)$$

where λ is a wavelength.

It is shown in [3] that the operator $\mathcal{D}_z$ is invertible, and the inverse operator also can be presented as the convolution with a shift-invariant kernel. If the diffraction wavefield $u_z(y,x)$ is given the wavefield in the object plane $z=0$ can be reconstructed using the inverse operator $\mathcal{D}_z^{-1}$ as

$$u_0(\eta,\xi)=\mathcal{D}_z^{-1}\{u_z\}\triangleq\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}g_{-z}(\eta-y,\xi-x)u_z(y,x)dxdy. \tag{3}$$

The term *diffraction transform* is used in [3] for the forward $\mathcal{D}_z$ and backward $\mathcal{D}_z^{-1}$ operators. Thus, the reconstruction problem is reduced to calculation of the inverse transform and the corresponding wavefield.

2.B. Convolutional discrete models

Discretization of the *integral diffraction transforms* defined by the formulas (1) and (3) is a straightforward idea in order to derive digital models for the forward and backward wavefield propagation. This discrete modeling is a subject of many publications and the discrete Fourier transform ($\mathcal{DFT}$) based techniques are very popular. While the review of this area is beyond the scope of this paper, we wish to mention that the discrete space domain modeling for holography is discussed in details in [2] and the accuracy of the frequency domain approach can be found in [5]. The $\mathcal{DFT}$ based algorithm with a detailed accuracy analysis is presented in [6]. A number of recent developments concern continuous and discrete Fresnel transforms and their multiresolution versions (e.g. [7]- [11]).

Two spatial domain approaches are the most popular ones for derivation of discrete models of the wavefield propagation: the $\mathcal{DFT}$ convolution and Fresnel transform.

The discrete convolutional models as sampled versions of the formulas (1) and (3) are of the form

$$u_z(\Delta_{y,z}s, \Delta_{x,z}t) = \tag{4}$$

$$\sum_{k,l} u_0(\Delta_{y,0}k, \Delta_{x,0}l) g_z(\Delta_{y,z}s - \Delta_{y,0}k, \Delta_{x,z}t - \Delta_{x,0}l) \cdot \Delta_{y,0}\Delta_{x,0}.$$

$$\hat{u}_0(\Delta_{y,0}k, \Delta_{x,0}l) = \tag{5}$$

$$\sum_{s,t} u_z(\Delta_{y,z}s, \Delta_{x,z}t) g_{-z}(\Delta_{y,0}k - \Delta_{y,z}s, \Delta_{x,0}l - \Delta_{x,z}t) \cdot \Delta_{y,z}\Delta_{x,z},$$

where the hat in $\hat{u}_0$ means an estimate of u_0, and $\Delta_{y,0} \times \Delta_{x,0}$, $\Delta_{y,z} \times \Delta_{x,z}$ are pixel's sizes (sampling intervals) in the object and sensor planes, respectively.

Assuming that the pixel sizes are equal both for the object and sensor planes, $\Delta_{y,z} = \Delta_{y,0} = \Delta_y$ and $\Delta_{x,z} = \Delta_{x,0} = \Delta_x$ we can treat this equation as a discrete shift-invariant convolution and apply $\mathcal{DFT}$ to the both parts of these equations (e.g. [2], [12]):

$$U_z(f_y, f_x) = G_z(f_y, f_x) U_0(f_y, f_x), \tag{6}$$

$$\hat{U}_0(f_y, f_x) = G_{-z}(f_y, f_x) U_z(f_y, f_x). \tag{7}$$

where $U_z = \mathcal{DFT}\{u_z\}$, $U_0 = \mathcal{DFT}\{u_0\}$, $\hat{U}_0 = \mathcal{DFT}\{\hat{u}_0\}$ and $G_z = \mathcal{DFT}\{g_z\}$ are calculated over the $N_y \times N_x$ pixels.

It is shown, in particular in [6], that the model (6)-(7) can be essentially improved if the kernel g_z in (4) is zero-padded to the double size and all calculations are produced for respectively double-sized U_z, U_0, $\hat{U}_0$, G_z. This zero-padding of the kernel g_z is applied in ([1], Section 3b) as a tool to obtain the accurate forward propagation modeling implemented in the frequency domain.

2.C. Fresnel discrete transform

If $z \gg \sqrt{x^2+y^2}$ the following approximation of the kernel g_z is valid

$$g_z(y,x) \simeq \frac{\exp(j2\pi z/\lambda)}{j\lambda z}\exp[j\frac{\pi}{\lambda z}(x^2+y^2)]. \qquad (8)$$

The approximation (8) is used for calculation of the integrals (1) and (3), further for discrete modeling by discretization of these integrals and derivation of the Fresnel transform [2].

These discretization results are summarized in the following proposition.

Proposition 1. I. *Provided (8), the discrete wavefield modeling of the* *forward propagation can be presented in the form:*

$$\begin{aligned}
&u_z(\Delta_{y,z}s, \Delta_{x,z}t) \triangleq \qquad (9)\\
&\frac{-j\exp[j(2\pi z/\lambda + \pi((\Delta_{y,z}s)^2 + (\Delta_{x,z}t)^2)/(\lambda z))]}{\lambda \cdot z} \times \\
&\sum_{k,l} \{\exp[j\frac{\pi}{\lambda z}((\Delta_{y,0}k)^2 + (\Delta_{x,0}l)^2)] u_0(\Delta_{y,0}k, \Delta_{x,0}l)\} \times \\
&\exp[-j\frac{2\pi\Delta_{y,0}\Delta_{y,z}sk + \Delta_{x,0}\Delta_{x,z}tl}{\lambda z}] \cdot \Delta_{y,0}\Delta_{x,0}.
\end{aligned}$$

and the modeling of the backward propagation as:

$$\begin{aligned}
&\hat{u}_0(\Delta_{y,0}k, \Delta_{x,0}l) \triangleq \qquad (10)\\
&\frac{j\exp[-j(2\pi z/\lambda + \pi((\Delta_{y,0}k)^2 + (\Delta_{x,0}l)^2)/(\lambda z))]}{\lambda \cdot z} \times \\
&\sum_{s,t} \exp[-j\frac{\pi}{\lambda z}((\Delta_{y,z}s)^2 + (\Delta_{x,z}t)^2)] u_z(\Delta_{y,z}s, \Delta_{x,z}t) \times \\
&\exp[j\frac{2\pi(\Delta_{y,0}\Delta_{y,z}sk + \Delta_{x,0}\Delta_{x,z}tl)}{\lambda z}] \cdot \Delta_{x,z}\Delta_{y,z}.
\end{aligned}$$

II. *Let the object and sensor planes be of the same size* $N_y \times N_x$ *and the following conditions be fulfilled*

$$N_y = \frac{\lambda \cdot z}{\Delta_{y,0}\Delta_{y,z}},\ N_x = \frac{\lambda \cdot z}{\Delta_{x,0}\Delta_{x,z}}, \qquad (11)$$

then forward and backward propagations (9)-(10) can be calculated using

$\mathcal{DFT}$:

$$u_z(\Delta_{y,z}s, \Delta_{x,z}t) \triangleq \tag{12}$$
$$\frac{-j}{\lambda \cdot z} \exp[j(2\pi z/\lambda + \pi((\Delta_{y,z}s)^2 + (\Delta_{x,z}t)^2)/(\lambda z))] \times$$
$$\mathcal{DFT}_{k,l}\{\exp[j\frac{\pi}{\lambda z}((\Delta_{y,0}k)^2 + (\Delta_{x,0}l)^2)]u_0(\Delta_{y,0}k, \Delta_{x,0}l)\}[s,t] \cdot \Delta_{y,0}\Delta_{x,0}.$$

and

$$\hat{u}_0(\Delta_{y,0}k, \Delta_{x,0}l) \triangleq \tag{13}$$
$$\frac{j}{\lambda \cdot z} \exp[-j(2\pi z/\lambda + \pi((\Delta_{y,0}k)^2 + (\Delta_{x,0}l)^2)/(\lambda z))] \times$$
$$\mathcal{DFT}^{-1}_{s,t}\{\exp[-j\frac{\pi}{\lambda z}((\Delta_{y,z}s)^2 + (\Delta_{x,z}t)^2)]u_z(\Delta_{y,z}s, \Delta_{x,z}t)\}[k,l] \times$$
$$N_y N_x \Delta_{x,z}\Delta_{y,z},$$

where

$$\hat{u}_0 = u_0. \tag{14}$$

The subscripts in $\mathcal{DFT}_{k,l}$ and $\mathcal{DFT}^{-1}_{s,t}$ show with respect to which variables these transforms are calculated.

While the formulas (9)-(13) are well known, often they are given in shorten forms where some of the factors (in particular phase factors) are omitted. For convenience of the reference and comparison with novel algorithms we present here these transforms in the complete forms.

Conventionally, the term *discrete Fresnel transform* is applied to the transforms defined by the formulas (12) and (13) valid provided the sampling conditions (11). This transform enables the perfect reconstruction of u_0, $\hat{u}_0 = u_0$.

More general formulas (9) and (10) are valid as discrete approximations of the integrals (1)-(3) for arbitrary parameter values, i.e. for any size of the object and diffraction planes and the pixels in these planes. However, in general, if the conditions (11) are not satisfied $\hat{u}_0$ defined by (10) is not equal to u_0, $\hat{u}_0 \neq u_0$.

The sums in (9) and (10) have a form

$$\sum_{s,t} \exp[j\frac{2\pi(\Delta_{y,0}\Delta_{y,z} \cdot sk + \Delta_{x,0}\Delta_{x,z} \cdot tl)}{\lambda z}] \cdot w(s,t),$$

where $w(s,t)$ is a function of integers s and t. These sums are called the *discrete-time/space Fourier transforms* and treated as the integral Fourier transform of the discrete sequences (e.g. [13]). These sums are not $\mathcal{DFT}$ because the factors of sk and tl in the argument of the exponent are not integer. However, these sums become $\mathcal{DFT}$ provided the conditions (11) and the fast Fourier transform algorithms can be used for calculations.

Let us illustrate the fact that (10) does not give the perfect reconstruction if the conditions (11) are not fulfilled.

Assume that $N_y = N_x = 512$, $\Delta_{y,0} = \Delta_{y,z} = \Delta_{x,0} = \Delta_{x,z} = 5\mu m$ and $z = d_f$ where the "*in-focus*" distance d_f is defined accordingly to (11) as $d_f \triangleq N_y \Delta_{y,0}^2/\lambda = 0.0202\ m$. In our numerical experiments through the paper we assume the amplitude modulation of the object distribution, i.e. u_0 is non-negative real-valued, $0 \leq u_0 \leq 1$, and the phase of u_0 is equal to zero. This amplitude distribution of u_0 is defined by the Baboon test-image. Fig. 2a demonstrates the "*perfect*" reconstruction of the amplitude distribution for the "*in-focus*" distance $z = d_f$. Now let us assume that the distance is different from d_f for instance equal to $z = 3d_f$. The corresponding reconstruction shown in Fig. 2b is far from being accurate. It confirms that the backward Fresnel transform is not inverse for the forward Fresnel transform. Fig. 2c shows the reconstruction of u_0 for $z = 3d_f$ using the regularized inverse technique $\mathcal{M} - \mathcal{DDT}$ introduced later. We can see a significant improvement in this reconstruction in comparison with what is achieved in Fig. 2b.

Proposition 2. *The discrete Fresnel transform formulas (9) and (10) can be represented in the following matrix forms*

$$\mathbf{u}_z = \mu_z \cdot \mathbf{C}_{y,z} \cdot \mathbf{u}_0 \cdot \mathbf{C}_{x,z}^T, \tag{15}$$

$$\hat{\mathbf{u}}_0 = \mu_0 \cdot \mathbf{C}_{y,-z} \cdot \mathbf{u}_z \cdot \mathbf{C}_{x,-z}^T, \tag{16}$$

where $\mathbf{C}_{y,z} = (\mathbf{C}_{y,z}[s,s'])_{N_{y,z}\times N_{y,0}}$, $\mathbf{C}_{x,z} = (\mathbf{C}_{x,z}[s,s'])_{N_{x,z}\times N_{x,0}}$, $\mathbf{C}_{y,-z} = (\mathbf{C}_{y,-z}[s,s'])_{N_{y,0}\times N_{y,z}}$, $\mathbf{C}_{x,-z} = (\mathbf{C}_{x,-z}[s,s'])_{N_{x,0}\times N_{x,z}}$ *are the matrices with the elements calculated according to the formulas*:

$$\begin{aligned}
&\mathbf{C}_{y,z}[s,s'] = \exp(j\frac{\pi}{\lambda z}(s\Delta_{y,z} - s'\Delta_{y,0})^2), \qquad (17)\\
&\mathbf{C}_{x,z}[s,s'] = \exp(j\frac{\pi}{\lambda z}(s\Delta_{x,z} - s'\Delta_{x,0})^2),\\
&\mu_z = \mu \cdot \Delta_{x,0}\Delta_{y,0},\ \mu_0 = \mu^* \cdot \Delta_{x,z}\Delta_{y,z},\\
&\mu = \frac{\exp(j2\pi z/\lambda)}{j\lambda \cdot z}.
\end{aligned}$$

Accordingly, $\mathbf{u}_z = (\mathbf{u}_z[s,t])_{N_{y,z}\times N_{x,z}}$, $\hat{\mathbf{u}}_0 = (\hat{\mathbf{u}}_0[k,l])_{N_{y,0}\times N_{x,0}}$ are the matrices of the sizes $N_{y,z}\times N_{x,z}$ *and* $N_{y,0}\times N_{x,0}$, respectively.

The proof of this proposition is elementary by inserting (17) into (15) and (16). Note that the derived formulas are valid for the non-square matrices of arbitrary sizes.

There is a serious difference between standard Fresnel transforms (12)-(13) valid provided the sampling conditions (11) and the formulas (9)-(10) valid in much more general case when these sampling conditions are not satisfied. Despite this difference the term discrete Fresnel transform is used addressing to both kind of the transforms. However, we use the term *matrix* (*forward* and *inverse*) *discrete Fresnel transform* ($\mathcal{M}-\mathcal{DF}r\mathcal{T}$ and $\mathcal{M}-\mathcal{IDF}r\mathcal{T}$) for the transforms defined by the matrix formulas (15) and (16), respectively. In our simulation experiments we use this matrix version of the discrete Fresnel transform.

It follows from (15)-(16) that the reconstruction of $\mathbf{u}_0$ from $\mathbf{u}_z$ can be obtained also as a solution of the equation (15). In this way we arrive to the backward propagation modeling quite different from the standard equation (16). For instance, for the square nonsingular $\mathbf{C}_{y,z}$ and $\mathbf{C}_{x,z}$ this backward propagation is of the form

$$\hat{\mathbf{u}}_0 = \frac{1}{\mu_z} \cdot \mathbf{C}_{y,z}^{-1} \cdot \mathbf{u}_z \cdot \mathbf{C}_{x,z}^{-T}, \tag{18}$$

and it is different from (16) because in general $\mathbf{C}_{y,z}^{-1} \neq \mathbf{C}_{y,-z}$ and $\mathbf{C}_{x,z}^{-1} \neq \mathbf{C}_{x,-z}$. Concerning the discussed above invertability of $\mathcal{M} - \mathcal{DF}r\mathcal{T}$ we can see that it has a place if and only if $\mathbf{C}_{y,z}\mathbf{C}_{y,-z}$ and $\mathbf{C}_{x,z}\mathbf{C}_{x,-z}$ are the identity matrices. This condition can be guaranteed under the assumptions (11) and has no place in the general case.

2.D. Principal limitations of the backward propagation integral

In order the back propagation integral (3) enables the perfect restoration of the object u_0 the distribution u_z should be given for all x and y, i.e. the sensor in the diffraction plane should of the infinite size. For the finite-size sensor the perfect restoration of any u_0 is not possible. Thus, there is a sort of non-symmetry for the forward and backward propagation integral operators.

In this discussion and in what follows the term the *perfect restoration* is a property of the algorithm to give the accurate (precise) reconstruction for any applicable object distribution. Thus, the backward diffraction transform is always non-perfect for the finite-size sensor as it is not able to give the perfect reconstruction for any object distribution while it can be achieved for some particular distributions. The perfect restoration property should guarantee the precise result for any distribution.

The property of the inverse integral diffraction transform to be non-perfect for a finite sensor is inherited by all discrete restoration algorithms based on approximations of the kernel g_{-z} in (3).

3. Matrix discrete diffraction transform

The standard techniques discussed in the previous section consider discrete models as approximations for forward and backward wavefield propagation integrals. In the approach originated in [1] and further developed in this paper we follow different ideas. First, we develop the accurate forward discrete modeling which is precise for a class of pixel-wise invariant distributions. Second, we reconstruct the object distribution by inverting this precise forward model. We show that under some assumptions the forward model can be inverted and then we can achieve the perfect reconstruction of any object distribution, of course from the class of the pixel-wise invariant distributions. The essential point is that for reconstruction we do not exploit the backward propagation integral (3), which is valid for the infinite sensor only. Instead we use the accurate discrete forward propagation model.

Let us assume that the input of our model is discrete defined by a pixel-wise constant object distribution and the output is also discrete as defined by the outputs of the sensor pixels. For these pixel-wise invariant distributions $\mathcal{DDT}$ gives an accurate *discrete-to-discrete* modeling because this model is obtained by the accurate integration of (1). In this development the standard assumptions concerning sampling and bandlimitedness can be omitted as not relevant and replaced by hypotheses of piece-wise constant distributions.

In [1] the $\mathcal{DDT}$ transform is proposed in two different complementary forms: spatial algebraic and frequency domains. The frequency form is exploited for calculations, while the algebraic form is used mainly for interpretations and illustrations.

In this paper we present a novel algebraic form of $\mathcal{DDT}$ based on two (line- and column-transform) matrices enabling the following goals: the forward propagation modeling, calculation of the inverse wavefield propagation and the analysis of the conditioning of the $\mathcal{DDT}$ forward propagation operator. We name this novel transform as **M**atrix **D**iscrete **D**iffraction **T**ransform ($\mathcal{M}-\mathcal{DDT}$). In order to make a difference between this novel $\mathcal{DDT}$ and the old one we name $\mathcal{DDT}$ introduced originally in [1] as **F**requency domain **D**iscrete **D**iffraction **T**ransform ($\mathcal{F}-\mathcal{DDT}$). The latter model has been presented assuming the same size square pixels in the input and output planes and the square matrices of the input and output data.

3.A. Forward propagation modeling

Let the pixels in the object and sensor planes can be rectangular of the sizes $(\Delta_{y,0}\times\Delta_{x,0})$ and $(\Delta_{y,z}\times\Delta_{x,z})$, respectively. The sizes of the images in the object and sensor planes measured in pixels can be also different $N_{y,0}\times N_{x,0}$ and $N_{y,z}\times N_{x,z}$. Then the formulas (10)-(13) from [1] are generalized to the following form

$$u_z(y,x) = \tag{19}$$
$$\sum_{k=-N_{y,0}/2}^{N_{y,0}/2-1} \sum_{l=-N_{x,0}/2}^{N_{x,0}/2-1} u_0[k,l] \int_{-\Delta_{x,0}/2}^{\Delta_{x,0}/2} d\xi \int_{-\Delta_{y,0}/2}^{\Delta_{y,0}/2} g_z(y - k\Delta_{y,0} + \eta, x - l\Delta_{x,0} + \xi) d\eta,$$

$$u_0[k,t] = u_0(k\Delta_{y,0} + \eta, l\Delta_{x,0} + \xi),\ |\eta| \leq \Delta_{y,0}/2,\ |\xi| \leq \Delta_{x,0}/2,$$

where the sum is calculated over $N_{y,0} \times N_{x,0}$ pixels of the input rectangular array.

Let the output signal of a sensor's pixel be the average value of the distribution impinging on this pixel:

$$u_z[s,t] = \frac{1}{\Delta_{y,z}\Delta_{x,z}} \int_{-\Delta_{y,z}/2}^{\Delta_{y,z}/2} \int_{-\Delta_{x,z}/2}^{\Delta_{x,z}/2} u_z(s\Delta_{y,z} + \eta', t\Delta_{x,z} + \xi') d\xi' d\eta'. \tag{20}$$

Inserting (19) into (20) we arrive to the *space domain* $\mathcal{DDT}$ introduced as a discrete convolution with a shift-varying kernel a_z:

$$u_z[s,t] = \sum_{k=-N_{y,0}/2}^{N_{y,0}/2-1} \sum_{l=-N_{x,0}/2}^{N_{x,0}/2-1} a_z[s,k;t,l] \cdot u_0[k,l], \tag{21}$$
$$s = -N_{y,z}/2, ..., N_{y,z}/2 - 1,$$
$$t = -N_{x,z}/2, ..., N_{x,z}/2 - 1,$$

where

$$a_z[s,k;t,l] = \frac{1}{\Delta_{y,z}\Delta_{x,z}} \int_{-\Delta_{x,z}/2}^{\Delta_{x,z}/2} \int_{-\Delta_{x,0}/2}^{\Delta_{x,0}/2} d\xi d\xi' \times \tag{22}$$

$$\int_{-\Delta_{y,z}/2}^{\Delta_{y,z}/2} \int_{-\Delta_{y,0}/2}^{\Delta_{y,0}/2} g_z(s\Delta_{y,z} - k\Delta_{y,0} + \eta' + \eta, t\Delta_{x,z} - l\Delta_{x,0} + \xi' + \xi) d\eta d\eta',$$

$$s = -N_{y,z}/2, ..., N_{y,z}/2 - 1,\ k = -N_{y,0}/2, ..., N_{y,0}/2 - 1,$$

$$t = -N_{x,z}/2, ..., N_{x,z}/2 - 1,\ l = -N_{x,0}/2, ..., N_{x,0}/2 - 1.$$

The kernel a_z in (22) is an averaged (pixel-wise double-averaged) version of the original kernel g_z in (1).

For the Fresnel approximation of g_z (8) the kernel a_z in (22) allows a factorization

$$a_z[s,k;t,l] = \mu \mathbf{A}_y[s,k] \mathbf{A}_x[t,l],\ \mu = \frac{\exp(j2\pi z/\lambda)}{j\lambda \cdot z} \tag{23}$$

where

$$\mathbf{A}_y[s,k] = \tag{24}$$

$$\frac{1}{\Delta_{y,z}} \int_{-\Delta_{y,z}/2}^{\Delta_{y,z}/2} \int_{-\Delta_{y,0}/2}^{\Delta_{y,0}/2} \exp(j\frac{\pi}{\lambda z}(s\Delta_{y,z} - k\Delta_{y,0} + \eta' + \eta)^2) d\eta d\eta',$$

$$\mathbf{A}_x[t,l] =$$

$$\frac{1}{\Delta_{x,z}} \int_{-\Delta_{x,z}/2}^{\Delta_{x,z}/2} \int_{-\Delta_{x,0}/2}^{\Delta_{x,0}/2} \exp(j\frac{\pi}{\lambda z}(t\Delta_{x,z} - l\Delta_{x,0} + \xi' + \xi)^2) d\xi d\xi'.$$

Inserting (23) into (21) we arrive to the formula

$$\mathbf{u}_z[s,t] = \mu \sum_{k=-N_{y,0}/2}^{N_{y,0}/2-1} \sum_{l=-N_{x,0}/2}^{N_{x,0}/2-1} \mathbf{A}_y[s,k]\mathbf{u}_0[k,l]\mathbf{A}_x[t,l], \tag{25}$$

what defines the matrix form of the discrete input-output forward propagation model

$$\mathbf{u}_z = \mu \cdot \mathbf{A}_y \cdot \mathbf{u}_0 \cdot \mathbf{A}_x^T. \tag{26}$$

Note, that calculations of the double integrals in (24) can be reduced to integration on a single variable using the formulas from Appendix 1.

The kernel $a_z[s,k;t,l]$ in (21) becomes shift-invariant depending on the differences of the arguments $s-k$ and $t-l$ as soon as the pixels at the object and sensor planes take equal sizes, $\Delta_{x,z} = \Delta_{x,0} = \Delta_x$ and $\Delta_{y,z} = \Delta_{y,0} = \Delta_y$. In this case the $\mathcal{DDT}$ becomes essentially simpler because the matrices in (26) are symmetrical with the elements depending on the differences of the indexes only, $\mathbf{A}_y[s,k] = \mathbf{A}_y[s-k] = \mathbf{A}_y[k-s]$, $\mathbf{A}_x[t,l] = \mathbf{A}_x[t-l] = \mathbf{A}_x[l-t]$. If $\Delta_y = \Delta_x$ and $N_y = N_x$ then $\mathbf{A}_x = \mathbf{A}_y$.

The formula (26) defines what we call the *matrix discrete diffraction transform* ($\mathcal{M}-\mathcal{DDT}$) with the averaged matrices. The matrices $\mathbf{A}_y$ and $\mathbf{A}_x^T$ manipulate respectively by the rows and columns of the matrix $\mathbf{u}_0$. The link between the object and diffraction plane distributions can be written in the operator form $\mathbf{u}_z = \mathcal{M}-\mathcal{DDT}\{\mathbf{u}_0\}$.

It follows from the definition formulas (19)-(20) as well as from the results presented in [1] that $\mathcal{M}-\mathcal{DDT}$ is aliasing free and gives the accurate distribution in the sensor plane provided that the object distribution is pixel-wise invariant.

3.B. Backward (inverse) modeling and perfect reconstruction

An inverse of $\mathcal{M}-\mathcal{DDT}$ can be defined as an operator mapping $\mathbf{u}_z$ into $\hat{\mathbf{u}}_0$. This inverse of $\mathcal{M}-\mathcal{DDT}$ gives the perfect reconstruction of the pixel-wise object distribution if the forward operator $\mathcal{M}-\mathcal{DDT}$ is non-singular. In this section, we clarify the assumptions when the $\mathcal{M}-\mathcal{DDT}$ is non-singular, analyze the conditioning of this transform and introduce the regularized inverse algorithm for the ill-conditioned $\mathcal{M}-\mathcal{DDT}$.

Using the matrix vectorization technique (e.g. [21]) the equation (26) can be rewritten in the standard matrix form explicitly resolved with respect to the variables in the equation as

$$\mathbf{u}_z^{\mathrm{col}} = \mu \cdot \mathbf{G} \cdot \mathbf{u}_0^{\mathrm{col}},\ \mathbf{G} = \mathbf{A}_x \otimes \mathbf{A}_y \tag{27}$$

where the index ($^{\mathrm{col}}$) means vector-variables built from columns of the corresponding matrix and the symbol $\otimes$ stands for the Kronecker product of matrices. The dimension of the matrix $\mathbf{G}$ is equal to $N_{y,z}N_{x,z} \times N_{y,0}N_{x,0}$, and the vectors $\mathbf{u}_z^{\mathrm{col}}$ and $\mathbf{u}_0^{\mathrm{col}}$ have the lengths $N_{y,z}N_{x,z}$, $N_{y,0}N_{x,0}$, respectively.

Let the size $N_{y,z}N_{x,z}$ of the sensor-vector is equal or larger than the size $N_{y,0}N_{x,0}$ of the object-vector, then the least square estimate $\hat{\mathbf{u}}_0^{\mathrm{col}}$ of $\mathbf{u}_0^{\mathrm{col}}$ is a solution of the normal equation

$$\mathbf{G}^H \mathbf{u}_z^{\mathrm{col}} = \mu \cdot \mathbf{G}^H \mathbf{G} \cdot \hat{\mathbf{u}}_0^{\mathrm{col}}. \tag{28}$$

For $\mathbf{G} = \mathbf{A}_x \otimes \mathbf{A}_y$ we obtain [21]

$$\mathbf{G}^H \cdot \mathbf{G} = (\mathbf{A}_x \otimes \mathbf{A}_y)^H \cdot (\mathbf{A}_x \otimes \mathbf{A}_y) =$$

$$(\mathbf{A}_x^H \otimes \mathbf{A}_y^H) \cdot \mathbf{A}_x \otimes \mathbf{A}_y = \mathbf{A}_x^H \mathbf{A}_x \otimes \mathbf{A}_y^H \mathbf{A}_y.$$

The matrices $\mathbf{A}_x^H \mathbf{A}_x$ and $\mathbf{A}_y^H \mathbf{A}_y$ are Hermitian and have real-valued non-negative eigenvalues. Let ρ_i, $i = 1, ..., N_{x,0}$, and λ_j, $j = 1, ..., N_{y,0}$, be these eigenvalues respectively for $\mathbf{A}_x^H \mathbf{A}_x$ and $\mathbf{A}_y^H \mathbf{A}_y$. Then, the $N_{y,0}N_{x,0}$ eigenvalues of $\mathbf{G}^H \cdot \mathbf{G}$ are defined by the products $\rho_i \lambda_j$, $i = 1, ..., N_{x,0}$, $j = 1, ..., N_{y,0}$ [21]. It follows that the conditioning number as well as the rank of the matrix $\mathbf{G}^H \cdot \mathbf{G}$ are the products of the conditioning numbers and ranks calculated separately for the matrices $\mathbf{A}_x^H \mathbf{A}_x$ and $\mathbf{A}_y^H \mathbf{A}_y$:

$$cond_{\mathbf{G}} = cond_{\mathbf{A}_x^H \mathbf{A}_x} \cdot cond_{\mathbf{A}_y^H \mathbf{A}_y},\ rank_{\mathbf{G}} = rank_{\mathbf{A}_x^H \mathbf{A}_x} \cdot rank_{\mathbf{A}_y^H \mathbf{A}_y} \tag{29}$$

The conditioning numbers and ranks of $\mathbf{G}^H \cdot \mathbf{G}$ conventionally serve as a measure of the complexity of the numerical solution of the set of the normal equations (28). Despite the fact that the representation in the form (27)-(28) is straightforward for use the standard linear algebra techniques we prefer to base our algorithms on the matrix representation (26), what allows to build numerical algorithms using the matrices $\mathbf{A}_y^H \mathbf{A}_y$ and $\mathbf{A}_x^H \mathbf{A}_x$ of much smaller dimensions than the dimension of the matrix $\mathbf{G}$. In a similar way, we will characterize the complexity of these problems using the ranks of the matrices $\mathbf{A}_y^H \mathbf{A}_y$ and $\mathbf{A}_x^H \mathbf{A}_x$ separately.

If the matrices $\mathbf{A}_y$ and $\mathbf{A}_x$ in (26) are square and non-singular then the perfect reconstruction of $\mathbf{u}_0$ from $\mathbf{u}_z$ is achieved with the solution in the obvious form

$$\hat{\mathbf{u}}_0 = \frac{1}{\mu} \mathbf{A}_y^{-1} \mathbf{u}_z \mathbf{A}_x^{-T}, \tag{30}$$

where $\hat{\mathbf{u}}_0 = \mathbf{u}_0$.

For rectangular object and sensor planes the perfect reconstruction is achieved provided the following assumptions:

1. The support of the sensor plane distribution is equal or larger than the

support of the object plane distribution

$$N_{x,z} \geq N_{x,0} \text{ and } N_{y,z} \geq N_{y,0}; \tag{31}$$

2. $\mathbf{A}_y$ and $\mathbf{A}_x$ are full rank matrices, i.e.

$$rank(\mathbf{A}_y) = N_{y,0},\ rank(\mathbf{A}_x) = N_{x,0}. \tag{32}$$

If the conditions 1 and 2 are held then the perfect reconstruction is given by the formula

$$\hat{\mathbf{u}}_0 = \frac{1}{\mu}(\mathbf{A}_y^H \mathbf{A}_y)^{-1} \mathbf{A}_y^H \mathbf{u}_z \mathbf{A}_x^* (\mathbf{A}_x^T \mathbf{A}_x^*)^{-1},\ \hat{\mathbf{u}}_0 = \mathbf{u}_0, \tag{33}$$

where $(^H)$ stands for the Hermitian conjugate, $\mathbf{A}_y^H = (\mathbf{A}_y^*)^T$.

The numerical study shows that depending on the parameters, in particular, on the distance z, the pixel's sizes, and object and sensor sizes the matrices $\mathbf{A}_y$ and $\mathbf{A}_x$ can be extremely ill-conditioned. This sort of ill-conditioning means that these matrices are *numerically singular* and the formulas (30) and (33) being formally correct practically are useless. The corresponding calculations cannot be fulfilled and gives unstable results highly sensitive with respect to the parameter variations, round off errors of calculations and observation noise.

To deal with this situation we replace the concept of the rank in (32) by the *numerical rank* calculated based on the singular value spectrum of the matrices and the perfect reconstruction by the *numerical perfect* (practically high accuracy) reconstruction. The algorithm (33) is also should be modified. One of the efficient ways to deal with the ill-conditioning is to involve a prior information on the distribution in question. The regularization is one of the conventional techniques for this goal.

Let us consider the standard Tikhonov's regularization [14], [15]. Instead of the solution of the equation (26) we are looking for the regularized estimate of $\mathbf{u}_0$ defined by minimization of the quadratic criterion

$$\begin{aligned} \hat{\mathbf{u}}_0 &= \arg\min_{\mathbf{u}_0} L, \qquad (34) \\ L &= ||\mathbf{u}_z - \mu\mathbf{A}_y\mathbf{u}_0\mathbf{A}_x^T||_F^2 + \alpha^2||\mathbf{u}_0||_F^2, \qquad (35) \end{aligned}$$

where the quadratic Frobenius matrix norm as defined by the formula $||\mathbf{u}_0||_F^2 = \sum_{k,l}|u_0(k,l)|^2$.

The regularization penalty $||\mathbf{u}_0||_F^2$ in L serves in order to enable a bounded and smooth estimate $\hat{\mathbf{u}}_0$ of the object distribution $\mathbf{u}_0$. The regularization parameter α^2 controls the level of the smoothness of the estimate $\hat{\mathbf{u}}_0$ [14].

Proposition 3*. The regularized inverse $\hat{\mathbf{u}}_0$ defined as a minimizer of (34)* *is a solution of the linear equation*

$$(|\mu|\mathbf{A}_y^H\mathbf{A}_y+\alpha\mathbf{I})\hat{\mathbf{u}}_\mathbf{0}(|\mu|\mathbf{A}_x^T\mathbf{A}_x^*+\alpha\mathbf{I})- \tag{36}$$

$$\alpha|\mu|\hat{\mathbf{u}}_\mathbf{0}\mathbf{A}_x^T\mathbf{A}_x^*-\alpha|\mu|\mathbf{A}_y^H\mathbf{A}_y\hat{\mathbf{u}}_0=\mu^*\mathbf{A}_y^H\mathbf{u}_z\mathbf{A}_x^*,\ \alpha\geq 0.$$

This proposition is proved in Appendix 2.

It is checked numerically that the second and the third components in the left-hand side of this equation are not essential and can be dropped with an approximate solution given in the explicit non-recursive form

$$\hat{\mathbf{u}}_0{=}\frac{1}{\mu}(\mathbf{A}_y^H\mathbf{A}_y+\frac{\alpha}{|\mu|}\mathbf{I})^{-1}\mathbf{A}_y^H\mathbf{u}_z\mathbf{A}_x^*(\mathbf{A}_x^T\mathbf{A}_x^*+\frac{\alpha}{|\mu|}\mathbf{I})^{-1}. \tag{37}$$

Comparing the latter equation with (33) we note that the inverse of $\mathbf{A}_y^H\mathbf{A}_y$ and $\mathbf{A}_x^T\mathbf{A}_x^*$ is replaced by the inverse of their regularized versions $\mathbf{A}_y^H\mathbf{A}_y+\frac{\alpha}{|\mu|}\mathbf{I}$ and $\mathbf{A}_x^T\mathbf{A}_x^*+\frac{\alpha}{|\mu|}\mathbf{I}$.

The square complex-valued matrices $\mathbf{A}_y^H\mathbf{A}_y$ ($N_{y,0}\times N_{y,0}$) and $\mathbf{A}_x^T\mathbf{A}_x^*$ ($N_{x,0}\times N_{x,0}$) are Hermitian with the real-valued non-negative spectrums (eigenvalues): $\{\rho_1\geq\rho_2\geq...\geq\rho_J\}$, where J is equal to $N_{y,0}$ for $\mathbf{A}_y^H\mathbf{A}_y$ and equal to $N_{x,0}$ for $\mathbf{A}_y^H\mathbf{A}_y$. The spectrums can be different for these matrices.

The numerical ranks of the matrices $\mathbf{A}_y^H\mathbf{A}_y$ and $\mathbf{A}_x^T\mathbf{A}_x^*$ are calculated as a number of eigenvalue elements of the corresponding matrix which are larger than a given threshold value. In our experiments we use the threshold equal to 10^{-12},

$$rank = \max_j (j : \rho_j/\rho_1 > 10^{-12}). \tag{38}$$

For $\alpha > 0$ the regularized inverse (37) gives a biased estimate of the true distribution. Smaller α means a smaller bias. However, too small α are not applicable as the estimate $\hat{\mathbf{u}}_0$ can be completely destroyed.

It follows from our numerical experiments that if the numerical rank is defined according to (38), and the value of this rank is equal to $N_{y,0}$ for $\mathbf{A}_y^H \mathbf{A}_y$ and $N_{x,0}$ for $\mathbf{A}_x^H \mathbf{A}_x$ then there is a small α enabling the high accuracy (numerically perfect) reconstruction of the object distribution.

Let the pixel's sizes be so small that the averaging in the forward propagation integrals (22) is not essential. Denote the corresponding $\mathbf{A}_y$ and $\mathbf{A}_x$

matrices with no averaging as $\mathbf{B}_y$ and $\mathbf{B}_x$, respectively, then

$$\begin{gathered}
\mathbf{B}_y[s,k] \triangleq \lim_{\Delta_{y,z},\Delta_{y,0}\to 0} \mathbf{A}_y[s,k] = \\
\Delta_{y,0}\cdot \exp(j\frac{\pi}{\lambda z}(s\Delta_{y,z} - k\Delta_{y,0})^2) = \Delta_{y,0}\cdot \mathbf{C}_{y,z}[s,k], \\
\mathbf{B}_x[t,l] \triangleq \lim_{\Delta_{x,z},\Delta_{x,z}\to 0} \mathbf{A}_x[l,t] = \\
\Delta_{x,0}\cdot \exp(j\frac{\pi}{\lambda z}(t\Delta_{x,z} - l\Delta_{x,0})^2) = \Delta_{x,0}\cdot \mathbf{C}_{x,z}[t,l], \\
s = -N_{y,z}/2, ..., N_{y,z}/2-1,\ k = -N_{y,0}/2, ..., N_{y,0}/2-1, \\
t = -N_{x,z}/2, ..., N_{x,z}/2-1,\ l = -N_{x,0}/2, ..., N_{x,0}/2-1,
\end{gathered} \tag{39}$$

where $\mathbf{C}_{y,z}$ and $\mathbf{C}_{x,z}$ are defined in (17).

Thus,

$$\mathbf{B}_y = \Delta_{y,0}\cdot \mathbf{C}_{y,z} \text{ and } \mathbf{B}_x = \Delta_{x,0}\cdot \mathbf{C}_{x,z}, \tag{40}$$

i.e. within the scalar factors $\Delta_{y,0}$, $\Delta_{x,0}$ the matrices $\mathbf{B}_y$ and $\mathbf{B}_x$ are identical to the matrices of the matrix Fresnel transform. We will call these $\mathbf{B}_y$ and $\mathbf{B}_x$ matrices *"non-averaged"*, in order to emphasize that the averaging effect is omitted.

Proposition 4.

Let the object and sensor planes be of the equal pixel-size $N_y \times N_x$ and the *conditions (11) be fulfilled. Then for the non-averaged matrices*

$$\frac{1}{\Delta_{y,0}^2 N_y}\mathbf{B}_y^H\mathbf{B}_y = \mathbf{I}_{N_y\times N_y},\ \frac{1}{\Delta_{x,0}^2 N_x}\mathbf{B}_x^T\mathbf{B}_x^* = \mathbf{I}_{N_x\times N_x}. \tag{41}$$

or

$$\frac{1}{N_y}\mathbf{C}_{y,z}^H\mathbf{C}_{y,z} = \mathbf{I}_{N_y\times N_y},\ \frac{1}{N_x}\mathbf{C}_{x,z}^T\mathbf{C}_{x,z}^* = \mathbf{I}_{N_x\times N_x}. \tag{42}$$

The proof of (41) is given in Appendix 3, and (42) follows from (40).

Proposition 4 proves that provided the conditions (11) the matrices $\mathbf{A}_y^H\mathbf{A}_y$ and $\mathbf{A}_x^T\mathbf{A}_x^*$ to be inverted in (37) become well conditioned, does not require any regularization, $\alpha = 0$, and can be inverted analytically with

$$\begin{gathered}\hat{\mathbf{u}}_0 {=} \frac{1}{\mu}(\mathbf{B}_y^H\mathbf{B}_y)^{-1}\mathbf{B}_y^H\mathbf{u}_z\mathbf{B}_x^*(\mathbf{B}_x^T\mathbf{B}_x^*)^{-1} = \\ \frac{1}{\mu\Delta_{y,0}^2\Delta_{x,0}^2 N_x N_y}\mathbf{B}_y^H\mathbf{u}_z\mathbf{B}_x^* = \frac{1}{\mu\Delta_{y,0}\Delta_{x,0}N_x N_y}\mathbf{C}_{y,z}^H\mathbf{u}_z\mathbf{C}_{x,z}^*.\end{gathered} \tag{43}$$

It can be checked using (11) that (43) is equivalent to the discrete Fresnel backward propagation (16). Thus, in these conditions $\mathcal{M}-\mathcal{DDT}$ with non-averaged matrices becomes equivalent to the inverse discrete Fresnel transform.

Using the result (41) we may rewrite the regularized inverse solution (37) as

$$\hat{\mathbf{u}}_0 = \frac{\mathbf{1}}{\mu \cdot \gamma_y \cdot \gamma_x} (\widetilde{\mathbf{B}_y^H \mathbf{B}_y} + \tilde{\alpha}_y \cdot \mathbf{I})^{-1} \mathbf{B}_y^H \mathbf{u}_z \mathbf{B}_x^* (\widetilde{\mathbf{B}_x^T \mathbf{B}_x^*} + \tilde{\alpha}_x \cdot \mathbf{I})^{-1}, \tag{44}$$

$$\widetilde{\mathbf{B}_y^H \mathbf{B}_y} = \mathbf{B}_y^H \mathbf{B}_y / \gamma_y,\ \widetilde{\mathbf{B}_x^T \mathbf{B}_x^*} = \mathbf{B}_x^T \mathbf{B}_x^* / \gamma_x,$$

$$\tilde{\alpha}_y = \alpha / (|\mu| \cdot \boldsymbol{\gamma}_y),\ \tilde{\alpha}_x = \alpha / (|\mu| \cdot \boldsymbol{\gamma}_x),\ \gamma_y = \Delta_{y,0}^2 N_y,\ \gamma_x = \Delta_{x,0}^2 N_x,$$

where $\tilde{\alpha}_y$ and $\tilde{\alpha}_x$ are normalized regularization parameters. Under assumptions of Proposition 4 $\widetilde{\mathbf{B}_y^H \mathbf{B}_y}$ and $\widetilde{\mathbf{B}_x^T \mathbf{B}_x^*}$ are the identity matrices.

With the averaged matrices the solution for $\hat{\mathbf{u}}_0$ has a form (44) where the matrices $\mathbf{B}_y$ and $\mathbf{B}_x$ are replaced by $\mathbf{A}_y$ and $\mathbf{A}_x$, respectively. In this case, the normalized matrices are different from the identity matrices. However, as a rule they are quasidiagonal with diagonal items close to 1 and very small absolute values of non-diagonal items. Then the standard requirements to the regularization parameters can be done in the form $\tilde{\alpha}_y \ll 1$, $\tilde{\alpha}_x \ll 1$, i.e.

$$0 < \alpha_0 \leq \alpha \ll |\mu| \min(\boldsymbol{\gamma}_y, \gamma_x). \tag{45}$$

This upper bound guarantees that the regularization does not change the main diagonal terms of the matrices, and the lower bound α_0 guarantees the regularization effects.

Let as evaluate the order of this upper bound for α. Note that $|\mu| = \dfrac{1}{\lambda d}$, then for the pixels of the equal size Δ and square image $N \times N$ we obtain

$|\mu|\min(\boldsymbol{\gamma}_y, \gamma_x) = \dfrac{\Delta^2 N}{\lambda d} = \dfrac{\Delta^2 N d_f}{\lambda d d_f} = \dfrac{d_f}{d}$. It means that roughly $\alpha \ll d_f/d$. Thus, for the distances close to the in-focus ones, $\alpha \ll 1$. Evaluation of the lower bound α_0 is a difficult problem. It has been found experimentally, that a values from the interval $[10^{-7}, 10^{-1}]$ on many occasions are able to give a good accuracy reconstruction of the object distribution.

A generalized Tikhonov's regularization uses various quadratic norms of linear functionals of $\mathbf{u}_0$ mainly such as gradients or the Laplacian [14], [15]. Further developments of the regularization techniques results in nonlinear penalty functions such as the total variation [16] and l_0 and l_1 norms. The latter leads to the recent compressed sensing estimators are able to built lower complexity estimates with strong denoising properties [17], [18]. All that sort of approaches are applicable to the considered inverse wavefront restoration problems.

4. Experiments on wavefield reconstruction

4.A. Simulations

Simulation is used as an universal tool for accurate study of the reconstruction accuracy and qualitative effects.

4.A.1. Sensor redundancy and reconstruction accuracy

It is assumed that images in object and sensor planes are square but can be of different size, $N_{y,0} = N_{x,0} = N_0$ and $N_{y,z} = N_{x,z} = N_z = qN_0$, $q \geq 1$, the pixels are square of the equal size for the sensor and object planes, $\Delta_{y,0} = \Delta_{x,0} = \Delta_0$, $\Delta_{y,z} = \Delta_{x,z} = \Delta_z$, $\Delta_0 = \Delta_z = \Delta$. The size of the image in the sensor plane is always larger than or equal to the size of the image in the object plane.

Let us denote the ratio of these sizes as $q = N_z/N_0$. The parameter q defines a redundancy of the sensor with respect to the object distribution. The larger sensor image the better accuracy of reconstruction for the object plane. While this statement sounds as quite obvious, one of the problem addressed is the analysis of the influence of the parameter q on the rank of the $\mathcal{M}-\mathcal{DDT}$ matrices and on the accuracy of reconstruction. Note that the $\mathcal{M}-\mathcal{DDT}$ algorithm always uses the averaged matrices $\mathbf{A}_y$ and $\mathbf{A}_x$.

The size of $\mathbf{A}_y$ and $\mathbf{A}_x$ is $N_z \times N_0$, while the size of $\mathbf{A}_y^H\mathbf{A}_y$ and $\mathbf{A}_x^H\mathbf{A}_x$ is always $N_0 \times N_0$ for any $N_z = qN_0$, $q \geq 1$. The rank and conditioning of $\mathbf{A}_y^H\mathbf{A}_y$ and $\mathbf{A}_x^H\mathbf{A}_x$ are seriously dependent on the sensor size N_z and the parameter q.

The distance $d = d_f$ corresponding to the conditions (11) is calculated as follows

$$d_f = N_z \Delta^2/\lambda = N_0 q \Delta^2/\lambda. \tag{46}$$

We call it "*in-focus*" distance with notation d_f because for the non-averaged matrices $\mathbf{B}_y^H \mathbf{B}_y$ and $\mathbf{B}_x^H \mathbf{B}_x$ it enables the perfect reconstruction (see *Proposition 3* for accurate statements). It has been observed from multiple simulation experiments that for the averaged matrices the distances $d = d_f$ and d close to d_f are also *in-focus* distances with the best reconstruction accuracy. This interesting fact cannot be proved analytically as there are no analytical expressions for the averaged matrices $\mathbf{A}_y$ and $\mathbf{A}_x$.

The numerical ranks of the matrix $\mathbf{A}_y^H \mathbf{A}_y$ (512×512) versus the distance d for different values of $q = \{1, 2, 4\}$ are shown in Fig. 3. In order to emphasize the dependence of the *"in-focus"* distance on the parameter q we will use notation $d_f|_q$ where the values of q is shown. Let us start from the curves corresponding to $q = 1$. The peaks of these curves for the averaged $\mathbf{A}_y$ and non-averaged $\mathbf{B}_y$ (as it is specified in (40)) cases are achieved exactly at the "*in-focus*" distance $d_f|_{q=1}$ and equal to the maximal rank 512. These ranks are quickly going down as $d > d_f$, and the ranks for the averaged case mainly lower than those for the non-averaged case. As a result the reconstruction of the object distribution becomes more and more questionable for the distances larger than d_f.

For instance, for $d = 3d_f|_{q=1} = 0.06\ m$ the rank for the averaged case is about 260, i.e. only 260 lines and columns from 512 can be treated as independent ones. We can expect that only this number of columns and lines can be reconstructed and there is no chance to reach a good quality imaging. For smaller distances $d < d_f$, the ranks are erratic, first decreasing rapidly, then increasing and again going down. It results in the correspondingly varying accuracy of reconstructions.

For the redundant sensors of the larger size with $q = 2$ and $q = 4$ the ranks are also decreasing functions for $d > d_f|_{q=2}$ and $d > d_f|_{q=4}$, respectively, but the rank values are much higher that it is for $q = 1$. Note also that nearly all cases the ranks for the averaged matrices are lower than those for the non-averaged ones.

The very interesting behavior of the ranks of the averaged matrices is observed for $d \leq d_f|_{q=2}$ and for $d \leq d_f|_{q=4}$ because these ranks take the maximum value 512 up to the very small values of d. It means that for these distances $\mathcal{M} - \mathcal{DDT}$ with the averaged matrices is able to give reconstructions of the very high accuracy.

For the non-averaged matrices the behavior of the rank curves is different. The ranks take the maximum values for $d_f|_{q=1} \leq d \leq d_f|_{q=2}$ if $q = 2$ and $d_f|_{q=1} \leq d \leq d_f|_{q=4}$ if $q = 4$ and become erratical for $d < d_f|_{q=1}$.

$RMSE$ for $\mathcal{M} - \mathcal{DDT}$ algorithm (with averaged matrices) versus the distance d for $q = 1$, $q = 2$ and $q = 4$ is shown in Fig. 4. We can see that if $q = 2$ we obtain the practical perfect reconstruction for all $d \leq d_f|_{q=2}$. If $q = 4$ this interval of the perfect reconstruction becomes wider, we have the perfect reconstruction for all $d \leq d_f|_{q=4}$. In all these calculations we use regularization parameters from the interval $[10^{-7}, 10^{-1}]$. For $q = 1$ the accuracy is much worse than it is for $q = \{2, 4\}$. Fig. 4 is complementary to Fig. 3. It demonstrates that for all intervals of d where the ranks take the maximum values the perfect (or high accuracy) reconstruction is achieved.

4.A.2. Comparative accuracy analysis

The comparative quantitative accuracy analysis of the different algorithms for various distances d is shown in Fig. 5. The best performance nearly for all d is demonstrated by the regularized inverse $\mathcal{M} - \mathcal{DDT}$ algorithm (37). The recursive regularized inverse frequency domain $\mathcal{F} - \mathcal{DDT}$ presented in [1] demonstrates the accuracy close to the one obtained by the regularized inverse $\mathcal{M}-\mathcal{DDT}$ algorithm. The results of $\mathcal{F}-\mathcal{DDT}$ are given for 10 iterations. It is emphasized that $\mathcal{M}-\mathcal{DDT}$ enables the perfect reconstruction for $d=d_f$. If the square pixels are of the equal size in the object and sensor planes and the corresponding images are also of the equal size the standard double-size zero-padding convolutional algorithm (*conv-2*) and $\mathcal{M}-\mathcal{IDF}r\mathcal{T}$ give the results nearly identical within the accuracy of calculations. The accuracy of these algorithms are much better than it can be obtained by the ordinary convolutional algorithm (*conv-1*). The curves corresponding to these algorithms in Fig. 5 are overlapping. Overall the $\mathcal{M}-\mathcal{DDT}$ algorithm demonstrates the essentially better accuracy than the standard ones. This algorithm differs from $\mathcal{M}-\mathcal{IDF}r\mathcal{T}$ only by use of the averaged matrices $\mathbf{A}_y$, $\mathbf{A}_x$, while the non-averaged $\mathbf{B}_y$, $\mathbf{B}_x$ are used in $\mathcal{M}-\mathcal{IDF}r\mathcal{T}$.

Fig. 6 and Fig. 7 illustrate a visual performance of the $\mathcal{M}-\mathcal{DDT}$ algorithm versus the $\mathcal{M}-\mathcal{IDF}r\mathcal{T}$ algorithm. In this simulation we use the $\mathcal{M}-\mathcal{DDT}$ with the averaged matrices for the forward propagation which gives the precise prediction of the wavefield distribution at the sensor plane. Thus, it is a good imitation of the optical experiments. The backward $\mathcal{M}-\mathcal{DDT}$ propagation (inverse distribution reconstruction) is performed according to the formula (37) and by $\mathcal{M}-\mathcal{IDF}r\mathcal{T}$ according to the formulas (15)-(16). The advantage of the regularized inverse algorithm is clear for all cases.

In Fig. 7 we use the redundant sensor of the double size with $q = 2$. The accuracy is essentially improved as compared with the case $q = 1$ shown in Fig. 6. The accuracy results similar to discussed above have been confirmed for various test images.

Fig. 8 demonstrates the ability of the $\mathcal{M}-\mathcal{DDT}$ algorithm with the averaging to give the aliasing free results. The left image is the $\mathcal{M}-\mathcal{IDF}r\mathcal{T}$ reconstruction with the distance $d = 0.5 \cdot d_f$. This reconstruction is completely destroyed by the aliasing effects. In the same time the $\mathcal{M}-\mathcal{DDT}$ reconstruction (right image) is nearly perfect.

When $d \neq d_f$ the matrices $\mathbf{A}_y^H\mathbf{A}_y$and $\mathbf{A}_x^H\mathbf{A}_x$ can be extremely ill-conditioned with the conditioning number of the order $10^{18} \div 10^{20}$. It is interesting that very small regularization parameters α are able to change the accuracy of the reconstructions in a dramatic way. In particular, the image Fig. 6a is obtained with very small $\alpha = 10^{-12}$. Increasing these regularization parameters upto the value 10^{-4} practically does not change the reconstruction accuracy. Thus, the algorithm is quite robust with respect to the selection of α. Nevertheless, non-zero positive regularization parameters are of importance as it is in particular illustrated in Fig. 9. These images are obtained for $d = 1.01 \times d_f$, i.e. the distance d is different from the in-focus one by one percent only. The image Fig. 9a obtained for $\alpha = 0$ is completely destroyed while the image in Fig. 9b obtained for $\alpha = 10^{-7}$ shows the high-accuracy imaging. The regularization results in a lower sensitivity of reconstruction to variations of the distance values. Note, that the zero regularization parameters can be used for $d = d_f$.

4.A.3. Non-square pixels and non-equal size images

In the above simulation produced for the square pixels of the equal size for the image and sensor planes the in-focus distance is selected using the formulas (46) obtained from (11). For the general case when the pixels are rectangular and of different size in the image and sensor planes and the images in these planes can be also of different size the sampling conditions (11) derived from (10) take a well known form [2]

$$N_{y,0}=\frac{\lambda \cdot d}{\Delta_{y,0}\Delta_{y,z}},\ N_{y,z}=\frac{\lambda \cdot d}{\Delta_{y,0}\Delta_{y,z}},\ N_{x,0}=\frac{\lambda \cdot d}{\Delta_{x,0}\Delta_{x,z}},N_{x,z}=\frac{\lambda \cdot d}{\Delta_{x,0}\Delta_{x,z}}. \qquad (47)$$

Of course these condition cannot be fulfilled simultaneously with the same value of the distance parameter d.

Nevertheless, the in-focus distance for $\mathcal{M}-\mathcal{DDT}$ can be found using the ranks of the matrices $\mathbf{A}_y$ and $\mathbf{A}_x$ analyzed as functions of d. The matrices $\mathbf{A}_y$ and $\mathbf{A}_x$ in (26) define transformations of the rows and columns of the object distribution $\mathbf{u}_0$ in order to map this distribution in the observed sensor plane distribution $\mathbf{u}_z$. The conditions (47) can be considered separately for these matrices as two groups of equations, respectively for the matrix $\mathbf{A}_y$

$$N_{y,0}=\frac{\lambda\cdot d}{\Delta_{y,0}\Delta_{y,z}},\ N_{y,z}=\frac{\lambda\cdot d}{\Delta_{y,0}\Delta_{y,z}} \tag{48}$$

and for the matrix $\mathbf{A}_x$

$$N_{x,0}=\frac{\lambda\cdot d}{\Delta_{x,0}\Delta_{x,z}},N_{x,z}=\frac{\lambda\cdot d}{\Delta_{x,0}\Delta_{x,z}}. \tag{49}$$

Then the in-focus distances for the matrices $\mathbf{A}_y$ ("focussing in-rows") $d_{f,y}$ and $\mathbf{A}_x$ ("focussing in-lines") $d_{f,x}$ are calculated as

$$\begin{aligned} d_{f,y} &= \Delta_{y,0}\Delta_{y,z}N_{y,z}/\lambda = q\Delta_{y,0}\Delta_{y,z}N_{y,0}/\lambda, \\ d_{f,x} &= \Delta_{x,0}\Delta_{x,z}N_{x,z}/\lambda = q\Delta_{x,0}\Delta_{x,z}N_{x,0}/\lambda \end{aligned} \tag{50}$$

In these formulas we use that $N_{y,z} \geq N_{y,0}$ and $N_{x,z} \geq N_{x,0}$. When $N_{y,z} = N_{y,0}$ and $N_{x,z} = N_{x,0}$ these matrices become the identical, respectively for $d = d_{f,y}$ and $d = d_{f,x}$. When $N_{y,z} > N_{y,0}$ and $N_{x,z} > N_{x,0}$ the sensor is redundant ($q > 1$) and the dependence of the rank with respect to d has a form shown in Fig. 3. Thus the rank takes the maximum values for the distances equal or smaller than defined in (50).

Now we select a single in-focus distance d_f from two $d_{f,y}$ and $d_{f,x}$ for the $\mathcal{M}-\mathcal{DDT}$ algorithm (with averaged matrices) as follows

$$d_f = \min(d_{f,y},\, d_{f,x}). \tag{51}$$

The ranks of the averaged $\mathbf{A}_y$ and $\mathbf{A}_x$ takes the maximal or close to maximal values for all $d \leq d_{f,y}$ and $d \leq d_{f,x}$. Thus selection of d_f guarantees than simultaneously the ranks of the both matrices $\mathbf{A}_y$ and $\mathbf{A}_x$ takes values close to the maximum values.

These results are valid only for the averaged matrices $\mathbf{A}_y$ and $\mathbf{A}_x$ and, in general, are not valid for the non-averaged $\mathbf{B}_y$ and $\mathbf{B}_x$. Note also that (51) is valid provided that a difference between $d_{f,y}$ and $d_{f,x}$ is not too large. Thus, the recommendation (51) is not universal and should be used cautiously.

Illustrations of using the focussing distance calculated according to the rule (51) are shown in Figures 10 and 11. It is assumed that the pixels are rectangular of the size (5×8) μm and the same for the object and sensor planes, the both images of the size 512×512. In Fig. 10 we can see the numerical ranks of the matrices $\mathbf{A}_y^H\mathbf{A}_y$ and $\mathbf{A}_x^H\mathbf{A}_x$. These ranks take their maximum value 512 for the corresponding in-focus distances $d_{f,y} = 0.0202$ m and $d_{f,y} = 0.0518$ m. It is seen that for $d_f = \min(d_{f,y}, d_{f,x}) = 0.0202$ m the rank of the matrix $\mathbf{A}_x^H\mathbf{A}_x$ is close to 512. It confirm the motivation behind selection of d_f according to the formula (51). Fig. 11 demonstrates that the object wavefield reconstruction according to $\mathcal{M}-\mathcal{DDT}$ and the best applicable standard technique $\mathcal{M}-\mathcal{IDF}r\mathcal{T}$. It is seen that the $\mathcal{M}-\mathcal{DDT}$ results (Fig. 11a) for $d_{f,y}$ is better than that (Fig. 11c) for the distance $d_{f,x}$ with lower $RMSE$ value 0.0115 versus 0.0559. Comparison with respect to $\mathcal{M}-\mathcal{IDF}r\mathcal{T}$ (Fig. 11b and Fig. 11d) is in favor $\mathcal{M}-\mathcal{DDT}$ both visually and numerically with the best $RMSE$ value 0.0809 achieved for the distance $d_{f,x}$.

4.B. Phase-shifting wavefield registration

The phase-shifting holography is one of the popular techniques used for measurement of the complex-valued wavefield distributions. In the basic setup for the in-line phase-shifting holography, the object beam and a special reference beam reflected at the piezoelectric transducer mirror controlled by a computer are combined on the digital sensor and form interference pattern [2], [12]. Following to [22] we show that $\mathcal{M}-\mathcal{DDT}$ techniques is applicable for the object wavefield reconstruction. The wavefield impinging on the digital sensor can be presented in the form

$$I_\phi(y,x) = |u_z(y,x) + u_{ref}(y,x)e^{j\phi(y,x)}|^2 =$$
$$|u_z(y,x)|^2 + |u_{ref}(y,x)|^2 +$$
$$u_z(y,x)u^*_{ref}(y,x)e^{-j\phi(y,x)} + u^*_z(y,x)u_{ref}(y,x)e^{j\phi(y,x)}.$$

Assume that the amplitude u_{ref} and the phase shift $\phi(y,x) = \phi$ of the reference beam are invariant then the averaged outputs of the sensor pixels are calculated for $I_\phi(y,x)$ according to (20) as follows:

$$\mathbf{I}_\phi[k,l] = \mathbf{I}_z[k,l] + |u_{ref}|^2 + \tag{52}$$
$$\mathbf{u}_z[k,l]u^*_{ref}e^{-j\phi} + \mathbf{u}^*_z[k,l]u_{ref}e^{j\phi},$$

where $\mathbf{I}_\phi[k,l] = \frac{1}{\Delta_{y,z}\Delta_{x,z}}\int_{-\Delta_{y,z}/2}^{\Delta_{y,z}/2}\int_{-\Delta_x,z/2}^{\Delta_{x,z}/2} I_\phi(k\Delta_{y,z}+\eta', l\Delta_{x,z}+\xi')d\xi' d\eta'$, $\mathbf{I}_z[k,l] = \frac{1}{\Delta_{y,z}\Delta_{x,z}}\int_{-\Delta_{y,z}/2}^{\Delta_{y,z}/2}\int_{-\Delta_x,z/2}^{\Delta_{x,z}/2} |u_z(k\Delta_{y,z}+\eta', l\Delta_{x,z}+\xi')|^2 d\xi' d\eta'$, and $\mathbf{u}_z[k,l]$ are given by (20).

Let us use for the three phase shifts $\phi = 0$, $\pi/2$, π then it follows from (52) that the complex-valued wavefield $\mathbf{u}_z[k,l]$ can be calculated as

$$\mathbf{u}_z[k,l] = \frac{1}{4u^*_{ref}}(\mathbf{I}_0[k,l] - \mathbf{I}_\pi[k,l] - j(2\mathbf{I}_{\pi/2}[k,l] - \mathbf{I}_0[k,l] - \mathbf{I}_\pi[k,l])) \tag{53}$$

and the wavefield reconstruction is reduced to reconstruction of the discrete object distribution $\mathbf{u}_0$ from the wavefield distribution $\mathbf{u}_z$ given in the sensor plane.

In our preliminary physical tests we use this phase-shifting setup with an object distribution generated by SLM. Visually $\mathcal{M}-\mathcal{DDT}$ gives quite a good reconstruction. An accuracy evaluation is planned as a further special experimental study, in particular including a precise calibration of the equipment and corrections of the wavefield propagation modeling.

4.C. Implementation

Computation of the integrals defining the matrix $\mathbf{A}_y$ is the most demanding element of the computational complexity of $\mathcal{M} - \mathcal{DDT}$. This matrix depends on the sizes of images and pixels in the object and sensor planes as well as on the distance between these planes. Thus, $\mathbf{A}_y$ can be calculated for a set of these parameters in advance, saved and used for processing of various wavefield distribution with the corresponding size and distance parameters. Respectively, in evaluation of the computational complexity we separate the time required for calculations of the matrices $\mathbf{A}_y$ (and $\mathbf{B}_y$) from the time required for calculation of wavefield reconstructions themselves. This separation allows to make a fare comparison of the computational complexity versus the standard methods where there are no calculations of the integrals used $\mathbf{A}_y$.

We represent the estimate reconstruction (37) into the form of the matrix product

$$\hat{\mathbf{u}}_0 = \frac{1}{\mu}\mathbf{Q_y}\mathbf{u}_z\mathbf{Q}_x, \tag{54}$$

where $\mathbf{Q}_y = (\mathbf{A}_y^H\mathbf{A}_y + \alpha/|\mu|\cdot\mathbf{I})^{-1}\mathbf{A}_y^H$, $\mathbf{Q}_x = \mathbf{A}_x^*(\mathbf{A}_x^T\mathbf{A}_x^* + \alpha/|\mu|\cdot\mathbf{I})^{-1}$. In Table 1 and Table 2 we show the computational time for calculations of the matrices $\mathbf{A}_y$ and $\mathbf{Q}_y$ (time for calculation of $\mathbf{A}_x$ and $\mathbf{Q}_x$ are equal to computational time for $\mathbf{A}_y$ and $\mathbf{Q}_y$, respectively).

All further results are given for square matrices of the size $N \times N$ with the pixel's sizes in the object and sensor planes $\Delta_{y,0} \times \Delta_{x,0}$. The results in Table 1 for matrix $\mathbf{A}_y$ essentially depend on the ratio of pixel sizes in object and sensor planes. If $\Delta_{y,z} = \Delta_{y,0}$, then $\mathbf{A}_y$ is the Toeplitz matrix defined by elements of a single column (row). It enables a fast calculation of the matrix with a minimal required time. If the pixels are not square and/or equal in the object and sensor planes, e.g. $\Delta_{y,z} \neq \Delta_{y,0}$, the computational time is much higher. In general, it requires calculation of all N^2 elements of the matrix with the maximal time shown in the last row of Table 1. Note that in this table $\Delta_{y,z} = 5\mu m$.

Table 1. Computational time (in sec) for matrices $\mathbf{A}_y$ and $\mathbf{B}_y$.

Matrix \ N^2	128^2	256^2	512^2	1024^2
$\mathbf{B}_y$	0.0046	0.02	0.11	0.43
$\mathbf{A}_y$, $\Delta_{y,0} = \Delta_{y,z}$	1.9	3.9	7.1	16
$\mathbf{A}_y$, $\Delta_{y,0} = 2 \times \Delta_{y,z}$	5.3	14.2	33	110
$\mathbf{A}_y$, $\Delta_{y,0} = 1.1 \times \Delta_{y,z}$	28.5	73.4	186	768

However, if for instance $\Delta_{y,0} = 2 \times \Delta_{y,z}$, as it in the third row of Table 1 or $\Delta_{y,0}$ is multiple to $\Delta_{y,z}$ with a factor different from 2 then a more economical algorithm is used looking for equal elements of the matrix $\mathbf{A}_y$ and calculating these equal elements only once. If these are equal elements in $\mathbf{A}_y$ and they are properly identified then the required time is much shorter than it is for calculation of the all items of $\mathbf{A}_y$. Of course, this acceleration depends on the ratio $\Delta_{y,z}/\Delta_{y,0}$.

In Table 2 we show the calculation time of the matrix $\mathbf{Q}_y$, which depends on the size of the matrix only. These results are obtained assuming that the matrix $\mathbf{A}_y$ is already known. In Table 3 we show the time used for calculation of the estimates (54) and compare it with the time required for the standard convolutional methods *conv-1* and *conv-2*. Naturally, that *conv-1* is fastest and the time required for *conv-2* estimates is much higher and comparable with the time required by $\mathcal{M} - \mathcal{DDT}$. Note, that in these calculations for $\mathcal{M} - \mathcal{DDT}$ we assume that the matrices $\mathbf{Q}_y$ and $\mathbf{Q}_x$ are given. Under this assumption the computational complexity of $\mathcal{M} - \mathcal{DDT}$ is of the order on the standard *conv-2* algorithm using the zero-padding.

Table 2. Computational time (in sec) for matrices $\mathbf{Q}_y$.

Matrix \ N^2	128^2	256^2	512^2	1024^2
$\mathbf{Q}_y$	0.02	0.15	1.1	10.1

Table 3. Computational time (in sec) for calculation of estimates .

Method \ N^2	128^2	256^2	512^2	1024^2
$\mathcal{M} - \mathcal{DDT}$	0.011	0.046	0.74	3.75
conv-1	0.0045	0.023	0.13	0.55
conv-2	0.023	0.13	0.55	2.3

The computational time results given in the tables are obtain by Monte-Karlo simulation with averaging over 100 samples. The computer used for experiments is Intel Pentium 4 @ 3.2GHz, RAM: 4GB, Windows Xp SP2, Matlab 7.7.0 (R2008b).

Memory requirements depend mainly on the size of the matrices in (54) and calculated as $N^2 \times 8 \times 2 \times$ 3 Bytes, where N^2 is a number elements in the matrices, 8 Bytes for double-precision floating point calculations, the factor 2 appears because of all variables are complex-valued, and 3 is a number of matrices in the product (54).

A demo-version of the $\mathcal{M}-\mathcal{DDT}$ and $\mathcal{F}-\mathcal{DDT}$ algorithms implemented in MATLAB are available at the website *http://www.cs.tut.fi/~lasip*.

5. Conclusion

This paper concerns two aspects of discrete holography: discrete modeling of the forward wavefield propagation and the inverse. The important property of the proposed $\mathcal{M}-\mathcal{DDT}$ is that it gives the accurate prediction of the forward propagation for a pixelwise invariant object distribution. In particular, this prediction is free of the aliasing effects typical for all standard discrete models. The forward $\mathcal{M}-\mathcal{DDT}$ can be treated as a generalization of the standard Fresnel transform to the averaged version of this transform, where the averaging is applied both in the sensor and object domains. The matrix form of the proposed $\mathcal{M}-\mathcal{DDT}$ is exploited in order to develop the inverse reconstruction techniques and formulate the conditions when the numerical perfect (high accuracy) reconstruction of the object plane wavefield distribution is possible.

Numerically and visually the $\mathcal{M}-\mathcal{DDT}$ based algorithms demonstrate a better performance as compared with the standard Fresnel transform and convolutional techniques. The inverse $\mathcal{M}-\mathcal{DDT}$ algorithms gives also better results as compared with the recursive and non-recursive $\mathcal{F}-\mathcal{DDT}$ reconstruction algorithms.

6. Acknowledgement

We highly appreciate very fruitful comments of three anonymous referees. This work was supported by the Academy of Finland, project No. 213462 (Finnish Centre of Excellence program 2006 – 2011).

References

1. V. Katkovnik, J. Astola, and K. Egiazarian , "Discrete diffraction transform for propagation, reconstruction, and design of wavefield distributions," *Applied Optics*, vol. 47,

no. 19, pp. 3481-3493, 2008.

2. Th.Kreis, *Handbook of Holographic Interferometry (Optical and Digital Methods)* (Wiley-VCH Gmbh&Co.KGaA, Weinheim, 2005).
3. G. S. Sherman, "Integral-transform formulation of diffraction theory," *J. Opt. Soc. Am*, vol. 57, no. 12, 1490-1498 (1967).
4. J. W. Goodman, *Introduction to Fourier Optics* (McGraw-Hill, Inc, New York, Second Edition, 1996).
5. L. Onural, "Exact analysis of the effects of sampling of the scalar diffraction field," *J. Opt. Soc. Am. A,* vol. 24, no. 2, 359-367 (2007).
6. F. Shen and A. Wang, "Fast-Fourier-transform based numerical integration method for the Rayleigh–Sommerfeld diffraction formula," *Appl. Opt.*, vol. 45, no. 6, 1102-1110 (2006).
7. L. Onural, "Some mathematical properties of the uniformly sampled quadratic phase function and associated issues in digital Fresnel diffraction simulations," *Opt. Eng.* vol. 43, no. 11, pp. 2557–2563 (2004).
8. I. Aizenberg and J. Astola, "Discrete generalized Fresnel functions and transforms in an arbitrary discrete basis, " *IEEE Trans on Signal Processing*, vol. 54, no.11, pp. 4261-4270 (2006).
9. M. Liebling, Th. Blu, and M. Unser, "Fresnelets: new multiresolution wavelet bases for digital holography," *IEEE Trans on Image Processing*, vol. 12, no. 1, pp. 29-43 (2003).
10. B. M. Hennelly and J.T. Sheridan,"Generalizing, optimizing, and inventing numerical algorithms for the fractional Fourier, Fresnel, and linear canonical transforms," *J. Opt.*

Soc. Am, vol. 22, no. 5 pp. 917-927 (2005).

11. L. Yaroslavsky, "Discrete transforms, fast algorithms and point spread functions of numerical reconstruction of digitally recorded holograms," pp. 93-141, in J. Astola and L. Yaroslavsky (eds.), *Advances in Signal Transforms: Theory and Applications*, (EURASIP Book Series on Signal Processing and Communications, vol. 7, Hindawi Publishing Corporation, 2007).
12. I. Yamaguchi, J. Kato, S. Ohta, and J. Mizuno, "Image formation in phase-shifting digital holography and applications to microscopy," *Appl. Opt.*, vol. 40, no. 34, 6177–6186 (2001).
13. A. V. Oppenheim and R.W. Schafer. *Discrete-Time Signal Processing*, 2nd Edition, (Prentice Hall Signal Processing Series, 1999).
14. Tikhonov A.N. and V.Y. Arsenin. *Solution of ill-posed problems* (Wiley, New York, 1977).
15. M. Bertero and P. Boccacci. *Introduction to inverse problems in imaging* . (IOP Publishing Ltd, 1998).
16. L. Rudin, S. Osher, and E. Fatemi, “Nonlinear total variation based noise removal algorithms”, *Physica D*, vol. 60, pp. 259–268 (1992).“
17. Donoho, D.L., “Compressed sensing”, *IEEE Trans. Inf. Theory*, vol. 52, no. 4, pp. 1289-1306, (2006).
18. E. Candès, J. Romberg, and T. Tao, “Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information”, *IEEE Trans. Inf. Theory*, vol. 52, no. 2, pp. 489-509, (2006).

19. V. Katkovnik, K. Egiazarian, J. Astola, *Local Approximation Techniques in Signal and Image Processing* (SPIE PRESS, Bellingham, Washington, 2006).

20. L.L.Scharf. *Statistical signal processing* (Prentice Hall, 1991).

21. Henry Theil, *Linear algebra and matrix methods in econometrics*, Handbook of Econometrics, Volume I, Edited by Z. Griliches and M.D. Intriligator (North- Holland Publishing Company, 1983).

22. V. Katkovnik, J. Astola, K. Egiazarian, "Numerical wavefield reconstruction in phase-shifting holography as inverse discrete problem," *EUSIPCO 2008*, 2008.

Appendix 1. Replacement of double integration by integration on single variable.

Consider the integral $J = \int_{-1/2}^{1/2} \int_{-1/2}^{1/2} f(x + \Delta_1 \xi_1 + \Delta_2 \xi_2) d\xi_1 d\xi_2$, where f is a function of two variables. Further, change the variables of integration according to the following formula

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} \cos\varphi & \sin\varphi \\ -\sin\varphi & \cos\varphi \end{pmatrix} \begin{pmatrix} \Delta_1 \xi_1 \\ \Delta_2 \xi_2 \end{pmatrix}. \tag{55}$$

With $\varphi = \pi/4$ this orthogonal transformation means a rotation of the integration area on the angle $\pi/4$ and multiplication on Δ_1 and Δ_2 results in the corresponding scalling of the obtained area. For $\varphi = \pi/4$ the new variables are defined as

$$\begin{pmatrix} u \\ v \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} \Delta_1 & \Delta_2 \\ -\Delta_1 & \Delta_2 \end{pmatrix} \begin{pmatrix} \xi_1 \\ \xi_2 \end{pmatrix} = \frac{1}{\sqrt{2}} \begin{pmatrix} \Delta_1 \xi_1 + \Delta_2 \xi_2 \\ -\Delta_1 \xi_1 + \Delta_2 \xi_2 \end{pmatrix}. \tag{56}$$

Let us assume that $\Delta_1 \geq \Delta_2$ then the integration area is a rectangular rotated on 45^0 as it is shown in Fig. 12. According to (56) the coordinates of the corners of this rectangular are as follows

$$\begin{aligned} u_1 &= \frac{1}{2\sqrt{2}}(\Delta_1 + \Delta_2),\ v_1 = \frac{1}{2\sqrt{2}}(-\Delta_1 + \Delta_2), \\ u_2 &= \frac{1}{2\sqrt{2}}(\Delta_1 - \Delta_2),\ v_2 = \frac{1}{2\sqrt{2}}(-\Delta_1 - \Delta_2). \end{aligned} \tag{57}$$

The integral J over this rectangular area is represented as

$$\begin{aligned} &J = \int_{-1/2}^{1/2}\int_{-1/2}^{1/2} f(x + \Delta_1\xi_1 + \Delta_2\xi_2)d\xi_1 d\xi_2 = \\ &\frac{1}{\Delta_1\Delta_2}\int\int_{u,v\in S} f(x+\sqrt{2}u)dudv, \end{aligned} \tag{58}$$

where S is an area of the rectangular. The integrand in (58) depends on the variable u and does not depends on v. It allows to represent this integral as a sum of the three integrals with integration over the areas I, II, and III shown in Figure 12. Using simple geometrical speculations we can calculate these integrals as integrals over a single variable u as follows

$$\begin{aligned} J &= J_1 + J_2 + J_3, \\ J_1 &= \frac{1}{\Delta_1\Delta_2}\int_{-u_2}^{u_2} f(x+\sqrt{2}u)(\Delta_2\sqrt{2})du, \\ J_2 &= \frac{1}{\Delta_1\Delta_2}\int_{u_2}^{u_1} f(x+\sqrt{2}u)2(u_1-u)du, \\ J_3 &= \frac{1}{\Delta_1\Delta_2}\int_{-u_1}^{-u_2} f(x+\sqrt{2}u)2(u_1+u)du. \end{aligned}$$

Inserting in these formulas the values (57) we arrive to

$$\begin{aligned} J_1 &= \frac{1}{\Delta_1}\int_{-(\Delta_1-\Delta_2)/2}^{(\Delta_1-\Delta_2)/2} f(x+u)du, \\ J_2 &= \frac{1}{\Delta_1\Delta_2}\int_{(\Delta_1-\Delta_2)/2}^{(\Delta_1+\Delta_2)/2} f(x+u)(\frac{\Delta_1+\Delta_2}{2}-u)du, \\ J_3 &= \frac{1}{\Delta_1\Delta_2}\int_{(\Delta_1-\Delta_2)/2}^{(\Delta_1+\Delta_2)/2} f(x-u)(\frac{\Delta_1+\Delta_2}{2}-u)du. \end{aligned} \tag{59}$$

If $\Delta_2 > \Delta_1$ the formulas (59) are used replacing in them Δ_1 by Δ_2 and Δ_2 by Δ_1. For the general case with the arbitrary Δ_1 and Δ_2 these formulas can be rewritten as

$$\begin{aligned} J_1 &= \frac{1}{\max(\Delta_1,\Delta_2)}\int_{-D_2}^{D_2} f(x+u)du, \\ J_2 &= \frac{1}{\Delta_1\Delta_2}\int_{D_2}^{D_1} f(x+u)(D_1-u)du, \\ J_3 &= \frac{1}{\Delta_1\Delta_2}\int_{D_2}^{D_1} f(x-u)(D_1-u)du, \end{aligned} \tag{60}$$

where $D_1 = (\Delta_1+\Delta_2)/2$, $D_2 = |\Delta_1-\Delta_2|/2$.

Let us derive the formulas valid for the integrals in (24). For $\mathbf{A}_y$ we have:

$$\mathbf{A}_y(k,s) = \frac{1}{\Delta_{y,z}}\int_{-\Delta_{y,z}/2}^{\Delta_{y,z}/2}\int_{-\Delta_{y,0}/2}^{\Delta_{y,0}/2} \exp(j\frac{\pi}{\lambda d}(k\Delta_{y,z} - s\Delta_{y,0} + \xi' + \xi)^2)d\xi d\xi' =$$

$$\Delta_{y,0}\int_{-1/2}^{1/2}\int_{-1/2}^{1/2} \exp(j\frac{\pi}{\lambda d}(k\Delta_{y,z} - s\Delta_{y,0} + \xi_1\Delta_{y,0} + \xi_2\Delta_{y,z})^2)d\xi_1 d\xi_2.$$

Then using (60) we obtain

$$\mathbf{A}_y(k,s) = \Delta_{y,0}\cdot(J_1(k,s) + \tilde{J}_2(k,s)), \tag{61}$$

$$J_1(k,s) = \frac{1}{\max(\Delta_{y,0},\Delta_{y,z})}\int_{-D_2}^{D_2} \exp(j\frac{\pi}{\lambda d}(k\Delta_{y,z} - s\Delta_{y,0} + u)^2)du, \tag{62}$$

$$J_2 \quad (k,s) + J_3(k,s) \triangleq \tilde{J}_2(k,s) = \frac{1}{\Delta_{y,0}\Delta_{y,d}}\int_{D_2}^{D_1} [\exp(j\frac{\pi}{\lambda d}(k\Delta_{y,d} - s\Delta_{y,0} + u)^2) +$$
$$\exp(j\frac{\pi}{\lambda d}(k\Delta_{y,d} - s\Delta_{y,0} - u)^2)](D_1 - u)du.$$

In a similar way we obtain for $\mathbf{A}_x$:

$$\mathbf{A}_x(l,t) = \Delta_{x,0} \cdot (J_1(l,t) + \tilde{J}_2(l,t)), \tag{63}$$

$$J_1(l,t) = \frac{1}{\max(\Delta_{x,0},\Delta_{x,z})}\int_{-D_2}^{D_2} \exp(j\frac{\pi}{\lambda d}(l\Delta_{x,z} - t\Delta_{x,0} + u)^2)du, \tag{64}$$
$$\tilde{J}_2(l,t) = \frac{1}{\Delta_{x,0}\Delta_{x,z}}\int_{D_2}^{D_1} [\exp(j\frac{\pi}{\lambda d}(l\Delta_{x,z} - t\Delta_{x,0} + u)^2) +$$
$$\exp(j\frac{\pi}{\lambda d}(l\Delta_{x,z} - t\Delta_{x,0} - u)^2)](D_1 - u)du.$$

For $\Delta_1 = \Delta_2 = \Delta$ the integral $J_1 = 0$, $J = \tilde{J}_2$ and it can be transformed to

$$J = \int_{-1}^{1} f(x + \Delta u)(1 - |u|)du. \tag{65}$$

In the derivation of this formula in [1] the erroneous extra factor 2 is appeared which should be omitted.

Appendix 2. Proof of Proposition 3

Using the matrix trace the criterion (35) can be expressed as follows

$$L = tr((\mathbf{u}_z - \mu \mathbf{A}_y \mathbf{u}_0 \mathbf{A}_x^T)^H (\mathbf{u}_z - \mu \mathbf{A}_y \mathbf{u}_0 \mathbf{A}_x^T) + \alpha^2 \mathbf{u}_0^H \mathbf{u}_0) =$$

$$tr(\mathbf{u}_z^H \mathbf{u}_z - \mu \mathbf{u}_z^H \mathbf{A}_y \mathbf{u}_0 \mathbf{A}_x^T - \mu^* \mathbf{A}_x^* \mathbf{u}_0^H \mathbf{A}_y^H \mathbf{u}_z + |\mu|^2 \mathbf{A}_x^* \mathbf{u}_0^H \mathbf{A}_y^H \mathbf{A}_y \mathbf{u}_0 \mathbf{A}_x^T + \alpha^2 \mathbf{u}_0^H \mathbf{u}_0).$$

Minimization in (34) concerns both the real and imaginary parts of $\mathbf{u}_0$. These minimum conditions can be given in the following form (e.g. Appendixes 14.6 and 14.7 in [19]):

$$\frac{\partial L}{\partial \mathbf{u}_0^H} = 0. \tag{66}$$

Using the formulas for the trace matrix differentiation (e.g. [20], pp. 275-276),

$$\frac{\partial}{\partial \mathbf{R}} tr(\mathbf{Q}\mathbf{R}) = \mathbf{Q}^T, \ \frac{\partial}{\partial \mathbf{R}} tr(\mathbf{Q}\mathbf{R}^T) = \mathbf{Q},$$

we arrive to the equation

$$-\mu^* \mathbf{A}_x^H \mathbf{u}_z^T \mathbf{A}_y^* + |\mu|^2 \mathbf{A}_x^H \mathbf{A}_x \mathbf{u}_0^T \mathbf{A}_y^T \mathbf{A}_y^* + \alpha^2 \mathbf{u}_0^T = 0,$$

what can be rewritten in the equivalent form

$$\mu^* \mathbf{A}_y^H \mathbf{u}_z \mathbf{A}_x^* = (|\mu| \mathbf{A}_y^H \mathbf{A}_y + \alpha \mathbf{I}) \mathbf{u}_0 (|\mu| \mathbf{A}_x^T \mathbf{A}_x^* + \alpha \mathbf{I}) - \alpha |\mu| \mathbf{u}_0 \mathbf{A}_x^T \mathbf{A}_x^* - \alpha |\mu| \mathbf{A}_y^H \mathbf{A}_y \mathbf{u}_0, \tag{67}$$

where $\mathbf{I}$ is the identity matrix of the corresponding size. For simplicity we do not show the size of the identity matrix. It proves Proposition 3.

The following iterative algorithm has been used to solve the matrix equation (67)

$$\begin{aligned}
&\mathbf{u}_0^{(k+1)} = (|\mu|\mathbf{A}_y^H\mathbf{A}_y + \alpha\mathbf{I})^{-1}\mu^*\mathbf{A}_y^H\mathbf{u}_z\mathbf{A}_x^*(|\mu|\mathbf{A}_x^T\mathbf{A}_x^* + \alpha\mathbf{I})^{-1} + \\
&|\mu|\boldsymbol{\alpha}(\mathbf{A}_y^H\mathbf{A}_y|\mu| + \alpha\mathbf{I})^{-1}\mathbf{u}_0^{(k)}\mathbf{A}_x^T\mathbf{A}_x^*(\mathbf{A}_x^T\mathbf{A}_x^*|\mu| + \alpha\mathbf{I})^{-1} + \\
&|\mu|\alpha(|\mu|\mathbf{A}_y^H\mathbf{A}_y + \alpha\mathbf{I})^{-1}\mathbf{A}_y^H\mathbf{A}_y\mathbf{u}_0^{(k)}(|\mu|\mathbf{A}_x^T\mathbf{A}_x^* + \alpha\mathbf{I})^{-1}, \\
&k = 0, 1, ..., \ \mathbf{u}_0^{(1)} = \mathbf{0}.
\end{aligned} \tag{68}$$

Appendix 3. Proof of Proposition 4

First, consider the product of the matrices $\mathbf{A}_y^H \mathbf{A}_y$. Some transformation shows that

$$
\begin{aligned}
&\mathbf{A}_y^H \mathbf{A}_y|_{k,l} = \Delta_{y,0}^2 \sum_s \rho_y^*[s-k]\rho_y[s-l] = \qquad (69)\\
&\Delta_{y,0}^2 \sum_s \exp(-j\frac{\pi}{\lambda z}(s\Delta_{y,z} - k\Delta_{y,0})^2)\exp(j\frac{\pi}{\lambda z}(s\Delta_{y,z} - l\Delta_{y,0})^2) = \\
&\Delta_{y,0}^2 \exp(-j\frac{\pi}{\lambda z}[(k\Delta_{y,0})^2 - (l\Delta_{y,0})^2]) \times \\
&\sum_s \exp(-j\frac{2\pi s\Delta_{y,z}\Delta_{y,0}}{\lambda z}(-k+l)) = \\
&\Delta_{y,0}^2 \exp(-j\frac{\pi}{\lambda z}[(k\Delta_{y,0})^2 - (l\Delta_{y,0})^2]) \times \\
&\sum_{s=-N_y/2}^{N_y/2-1} \exp(-j\frac{2\pi s}{N_y}(-k+l)) = \Delta_{y,0}^2 N_y \delta_{k,l}.
\end{aligned}
$$

It proves that $\mathbf{A}_y^H \mathbf{A}_y$ within the scalar factor is the identity matrix, $\mathbf{A}_y^H \mathbf{A}_y = \Delta_{y,0}^2 N_y \cdot \mathbf{I}_{N_y \times N_y}$. In a similar way, $\mathbf{A}_x^T \mathbf{A}_x^* = \Delta_{x,0}^2 N_x \cdot \mathbf{I}_{N_x \times N_x}$.

List of Figure Captions

Fig. 1. Principal setup of wavefield propagation and reconstruction.

Fig. 2 Image reconstruction: (a) by the backward Fresnel transform for $z = d_f$ (perfect quality), (b) by the backward Fresnel transform for $z = 3d_f$ (average quality), (c) by the regularized inverse $\mathcal{M} - \mathcal{DDT}$ algorithm (quality is improved with respect to (b)).

Fig. 3. The rank of the matrix $\mathbf{A}_y^H \mathbf{A}_y$ versus the distance $z = d$ for averaged (av) $\mathbf{A}_y$ and non-averaged (non-av) $\mathbf{B}_y$ matrices for different sizes of the square sensor defined by the parameter q.

Fig. 4. $RMSE$ versus the distance $z = d$ for $q = 1$, $q = 2$ and $q = 4$, averaged matrices. A nearly perfect reconstruction is obtained: for all $d \leq d_f|_{q=2}$ if $q = 2$ and for all $d \leq d_f|_{q=4}$ if $q = 4$.

Fig. 5. The accuracy of the image restoration ($RMSE$) versus the distance d for different algorithms: $\mathcal{M} - \mathcal{DDT}$, the convolutional inverse using the transfer function of the image size (conv-1) and of the double size with zero-padding image (conv-2), $\mathcal{M} - \mathcal{IDFrT}$, and the recursive regularized inverse $\mathcal{F} - \mathcal{DDT}$, $d_f = 0.02\ m$, $q = 1$.

Fig. 6. Comparative imaging by $\mathcal{M}-\mathcal{DDT}$ and $\mathcal{M}-\mathcal{IDFrT}$ algorithms, for various distances between the object and sensor planes: a) $d = d_f$, b) $d = 3d_f$, c) $d = 6d_f$, where d_f is the in-focus distance for $q = 1$, $d_f = d_f|_{q=1}$. The images in the object and sensor planes are of the equal size, $q = 1$.

Fig. 7. Comparative imaging by $\mathcal{M}-\mathcal{DDT}$ and $\mathcal{M}-\mathcal{IDFrT}$ algorithms, for various distances between the object and sensor planes: a) $d = d_f$, b) $d = 3d_f$, c) $d = 6d_f$, where d_f is the in-focus distance for $q = 1$, $d_f = d_f|_{q=1}$. The image in the sensor plane is of the double size of the image in the object plane, $q = 2$.

Fig. 8. Object wavefield reconstruction (amplitude distribution) with the distance $z = .5d_f$: (left) standard $\mathcal{M}-\mathcal{IDFrT}$ fails with a pattern of clear aliasing effects, (right) $\mathcal{M}-\mathcal{DDT}$ with the averaged matrices gives a good quality aliasing free reconstruction.

Fig. 9. $\mathcal{M}-\mathcal{DDT}$ reconstructions, the distance $d = 1.01 \times d_f$: image (a) obtained with $\alpha = 0$ is completely destroyed, image (b) obtained with $\alpha = 10^{-7}$ gives a very good accuracy $RMSE = 0.0049$.

Fig. 10. The ranks of the matrices $\mathbf{A}_y^H \mathbf{A}_y$ and $\mathbf{A}_x^H \mathbf{A}_x$ for the rectangular pixels (5×8) μm and $RMSE$ for the Baboon image reconstruction by

$\mathcal{M}-\mathcal{DDT}$ and $\mathcal{M}-\mathcal{IF}r\mathcal{DT}$ algorithms versus the distance d.

Fig. 11. Comparative imaging by the $\mathcal{M}-\mathcal{DDT}$ and $\mathcal{M}-\mathcal{IDF}r\mathcal{T}$algorithms for the distances $d = d_{f,y} = 0.0202$ m (first row) and $d = d_{f,x} = 0.0518$ m (second row): (a) $\mathcal{M}-\mathcal{DDT}$ reconstruction, $RMSE = 0.0115$; (b) $\mathcal{M}-\mathcal{IDF}r\mathcal{T}$ reconstruction, $RMSE = 0.874$; (c) $\mathcal{M}-\mathcal{DDT}$ reconstruction, $RMSE = 0.0115$; (d) $\mathcal{M}-\mathcal{IDF}r\mathcal{T}$ reconstruction, $RMSE = 0.0809$.

Fig. 12. Integration areas for calculation of the integral (58).

List of Table Captions

Table 1 Computational time (in sec) for matrices $\mathbf{A}_y$ and $\mathbf{B}_y$.

Table 2 Computational time (in sec) for matrices $\mathbf{Q}_y$.

Table 3 Computational time (in sec) for calculation of estimates.

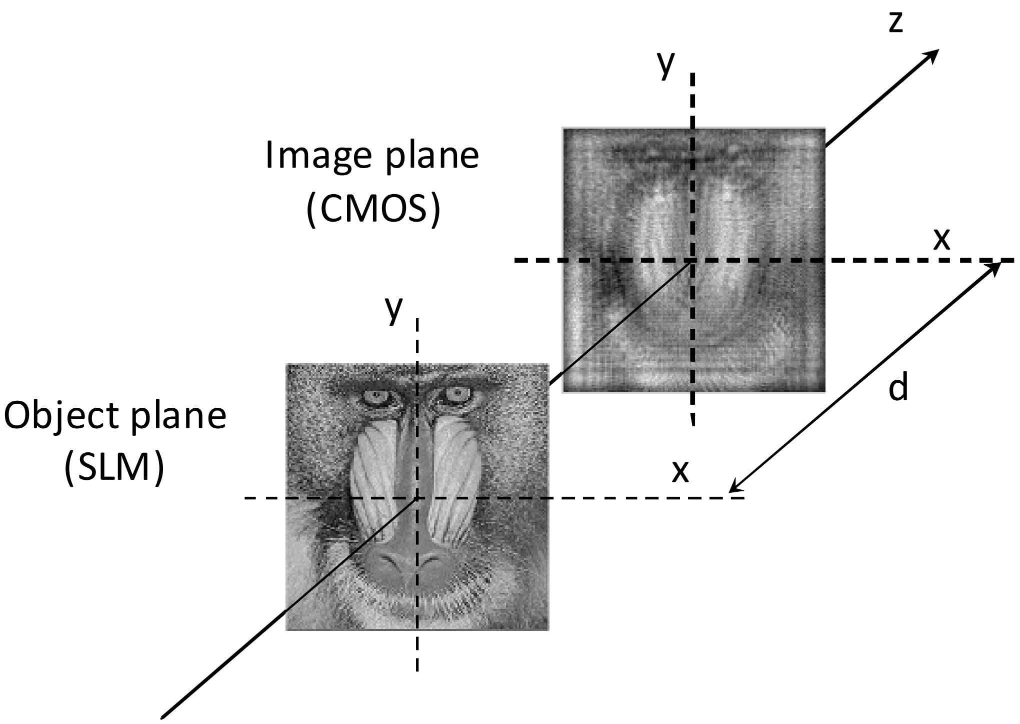
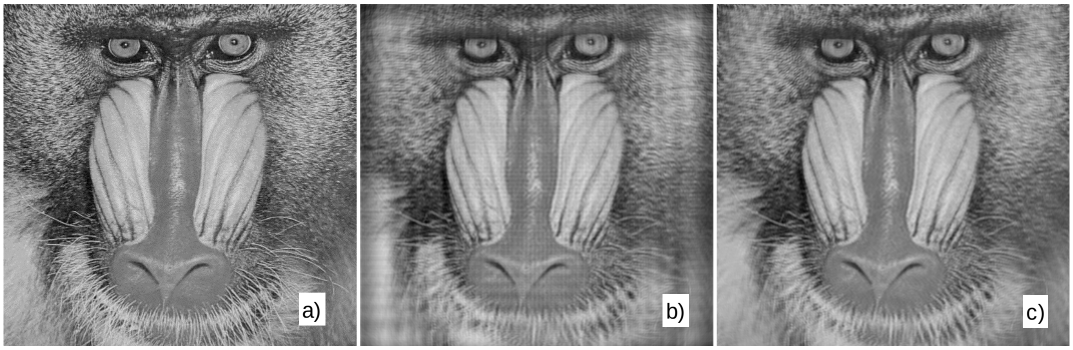


Fig. 1. Principal setup of wavefield propagation and reconstruction.



$RMSE = 2 \times 10^{-14}$, $z = d_f$; $RMSE = 0.100$, $z = 3d_f$; $RMSE = 0.074$, $z = 3d_f$

Fig. 2. Image reconstruction: (a) by the backward Fresnel transform for $z = d_f$ (perfect quality), (b) by the backward Fresnel transform for $z = 3d_f$ (average quality), (c) by the regularized inverse $\mathcal{M} - \mathcal{DDT}$ algorithm (quality is improved with respect to (b)).

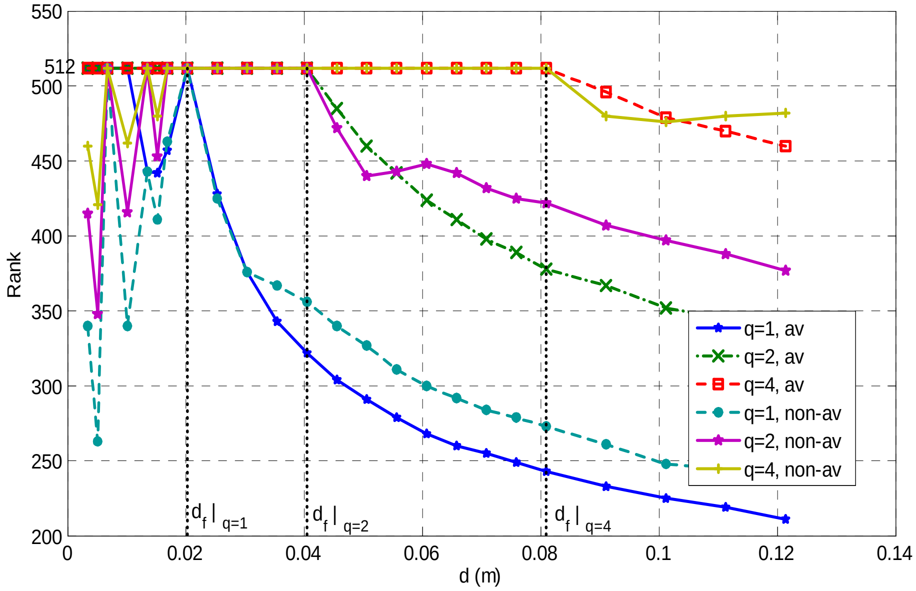


Fig. 3. The rank of the matrix $\mathbf{A}_y^H \mathbf{A}_y$ versus the distance $z = d$ for averaged (av) $\mathbf{A}_y$ and non-averaged (non-av) $\mathbf{B}_y$ matrices for different sizes of the square sensor defined by the parameter q.

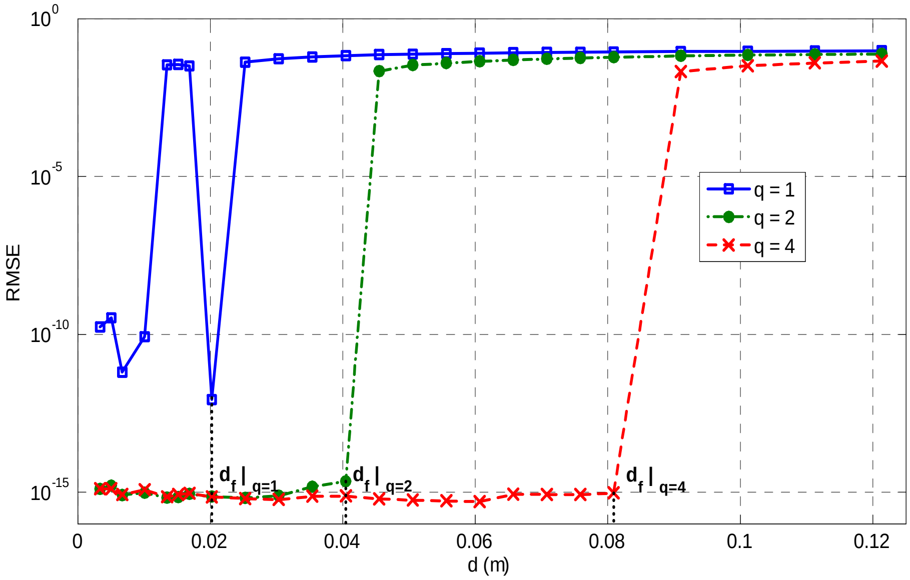


Fig. 4. $RMSE$ versus the distance $z = d$ for $q = 1$, $q = 2$ and $q = 4$, averaged matrices. A nearly perfect reconstruction is obtained: for all $d \leq d_f|_{q=2}$ if $q = 2$ and for all $d \leq d_f|_{q=4}$ if $q = 4$.

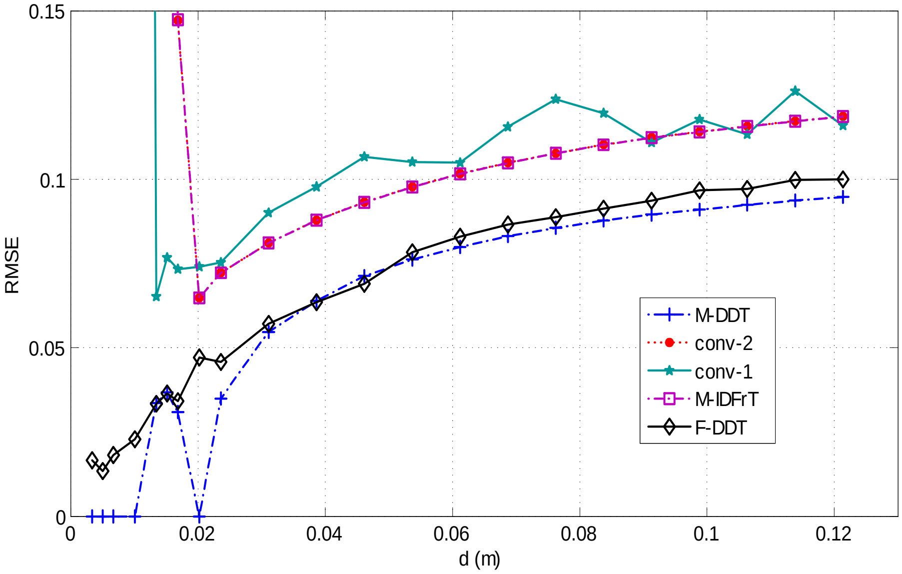
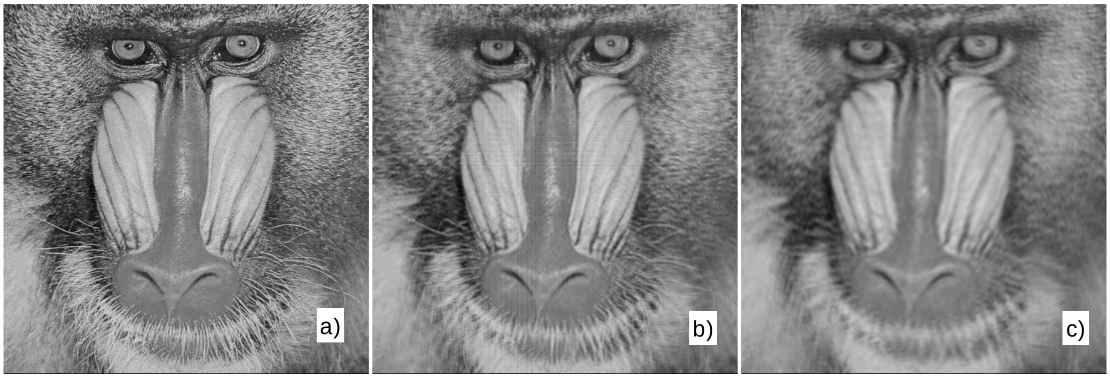


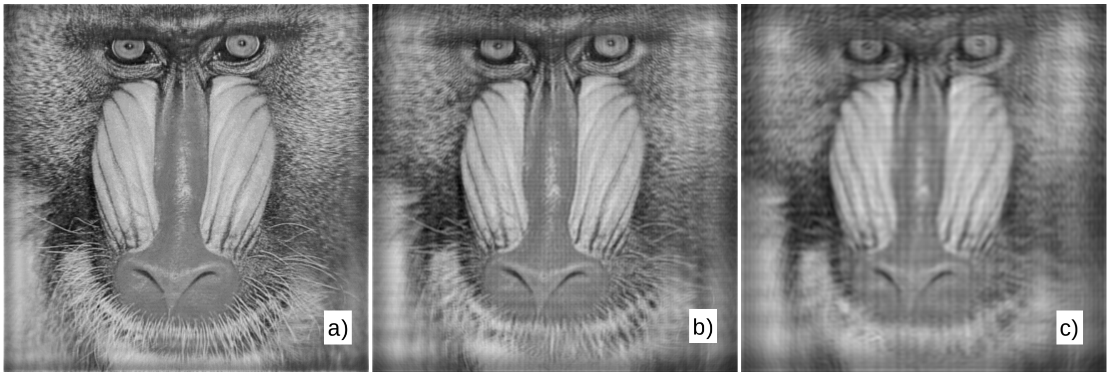
Fig. 5. The accuracy of the image restoration $(RMSE)$ versus the distance d for different algorithms: $\mathcal{M}-\mathcal{DDT}$, the convolutional inverse using the transfer function of the image size (conv-1) and of the double size with zero-padding image (conv-2), $\mathcal{M}-\mathcal{IDF}r\mathcal{T}$, and the recursive regularized inverse $\mathcal{F}-\mathcal{DDT}$, $d_f = 0.02\ m$, $q = 1$.

Imaging by $\mathcal{M}-\mathcal{DDT}$, sensor size parameter $q=1$



$RMSE = 7.7\times10^{-13}$, $z=d_f$; $RMSE = 0.074$, $z=3d_f$; $RMSE = 0.090$, $z=6d_f$

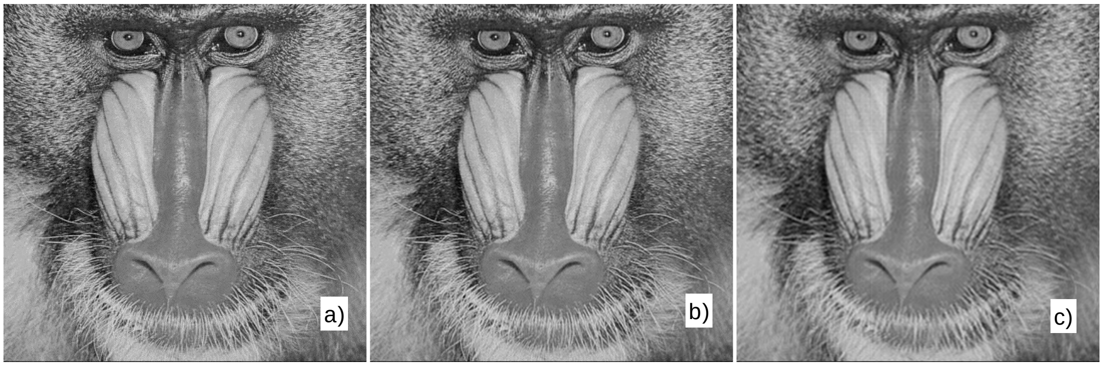
Imaging by $\mathcal{M}-\mathcal{IDF}r\mathcal{T}$, sensor size parameter $q=1$



$RMSE = 0.065$, $d=d_f$; $RMSE = 0.101$, $d=3d_f$; $RMSE = 0.119$, $d=6d_f$

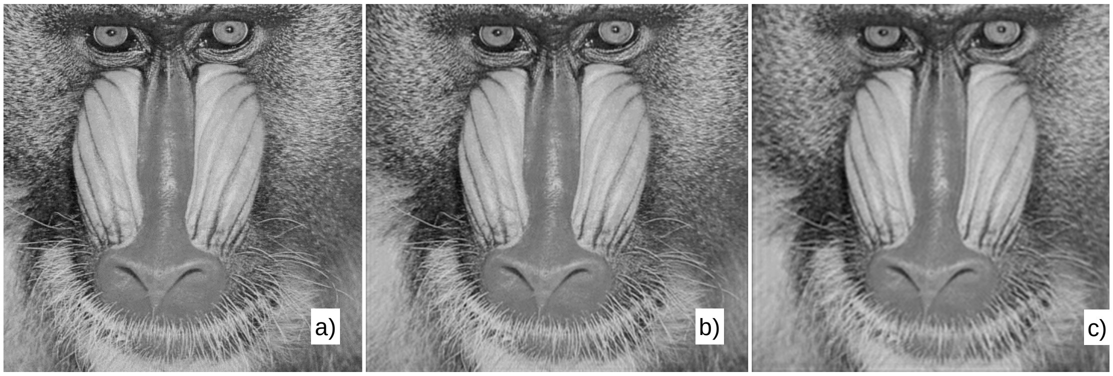
Fig. 6. Comparative imaging by $\mathcal{M}-\mathcal{DDT}$ and $\mathcal{M}-\mathcal{IDF}r\mathcal{T}$ algorithms, for various distances between the object and sensor planes: a) $d=d_f$, b) $d=3d_f$, c) $d=6d_f$, where d_f is the in-focus distance for $q=1$, $d_f=d_f|_{q=1}$. The images in the object and sensor planes are of the equal size, $q=1$

Imaging by $\mathcal{M}-\mathcal{DDT}$, sensor size parameter $q=2$



$RMSE = 7.148\times 10^{-16}$, $z=d_f$; $RMSE = 0.0444$, $z=3d_f$; $RMSE = 0.0763$, $z=6d_f$

Imaging by $\mathcal{M}-\mathcal{IDFrT}$, sensor size parameter $q=2$



$RMSE = 0.0165$, $z=d_f$; $RMSE = 0.0548$, $z=3d_f$; $RMSE = 0.0819$, $z=6d_f$

Fig. 7. Comparative imaging by $\mathcal{M}-\mathcal{DDT}$ and $\mathcal{M}-\mathcal{IDFrT}$ algorithms, for different distances between the object and sensor planes: a) $z=d_f$, b) $z=3d_f$, c) $z=6d_f$, where d_f is the in-focus distance for $q=1$, $d_f=d_f|_{q=1}$. The image in the sensor plane is of the double size of the image in the object plane, $q=2$.

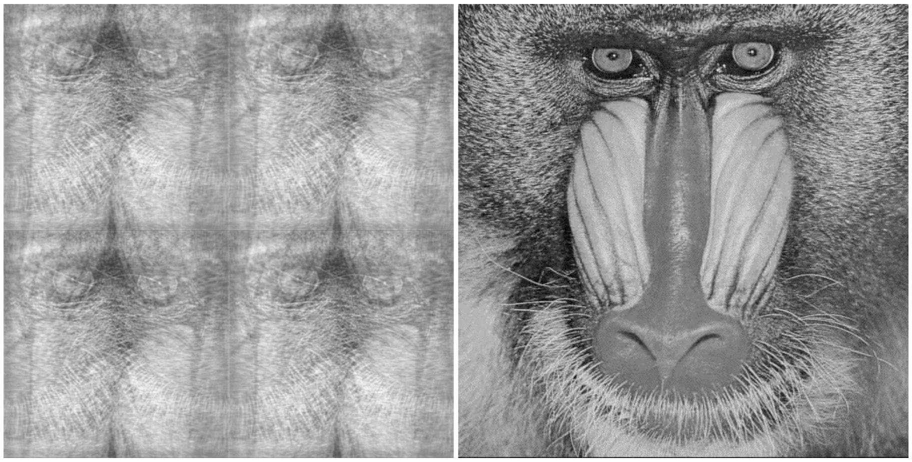


Fig. 8. Object wavefield reconstruction (amplitude distribution) the distance $z = 0.5d_f$: (left) $\mathcal{M} - \mathcal{IDF}r\mathcal{T}$ fails with a patern of clear aliasing effects, (right) inverse $\mathcal{M} - \mathcal{DDT}$ with the averaged matrices gives a good quality aliasing free reconstruction.

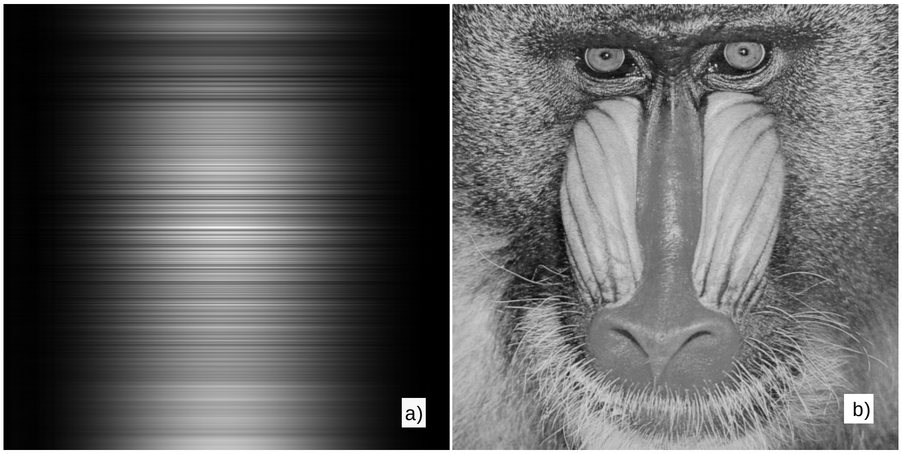


Fig. 9. $\mathcal{M} - \mathcal{DDT}$ reconstructions, the distance $d = 1.01 \times d_f$: image (a) obtained with $\alpha = 0$ is completely destroyed, image (b) obtained with $\alpha = 10^{-7}$ gives a very good accuracy $RMSE = 0.0049$.

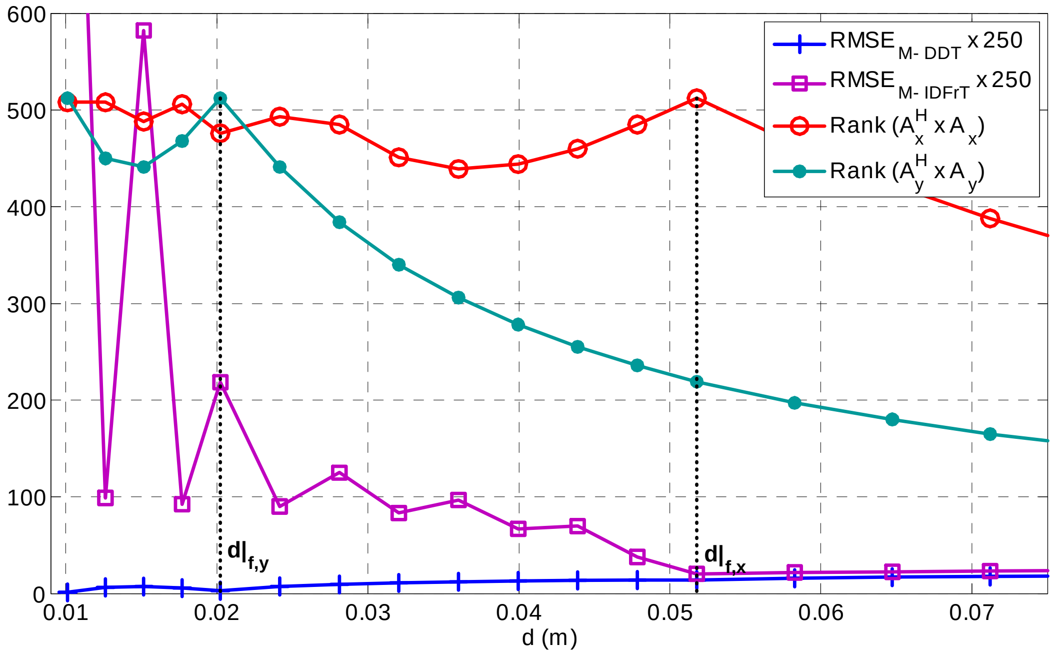


Fig. 10. The ranks of the matrices $\mathbf{A}_y^H \mathbf{A}_y$ and $\mathbf{A}_x^H \mathbf{A}_x$ for the rectangular pixels (5×8) μm and $RMSE$ for the Baboon image reconstruction by $\mathcal{M}-\mathcal{DDT}$ and $\mathcal{M}-\mathcal{IDF}r\mathcal{T}$ algorithms versus the distance d.

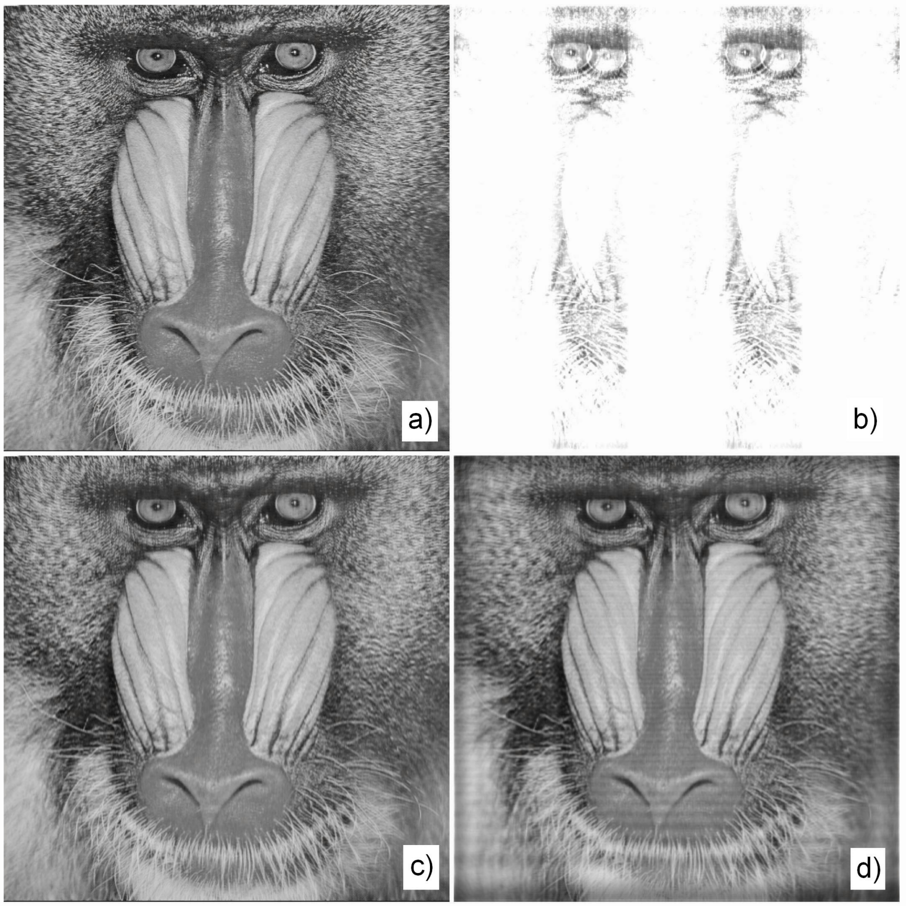


Fig. 11. Comparative imaging by the $\mathcal{M}-\mathcal{DDT}$ and $\mathcal{M}-\mathcal{IDF}r\mathcal{T}$algorithms for the distances $d = d_{f,y} = 0.0202\ m$ (first row) and $d = d_{f,x} = 0.0518\ m$ (second row): (a) $\mathcal{M}-\mathcal{DDT}$ reconstruction, $RMSE = 0.0115$; (b) $\mathcal{M}-\mathcal{IDF}r\mathcal{T}$ reconstruction, $RMSE = 0.874$; (c) $\mathcal{M}-\mathcal{DDT}$ reconstruction, $RMSE = 0.0115$; (d) $\mathcal{M}-\mathcal{IDF}r\mathcal{T}$ reconstruction, $RMSE = 0.0809$.

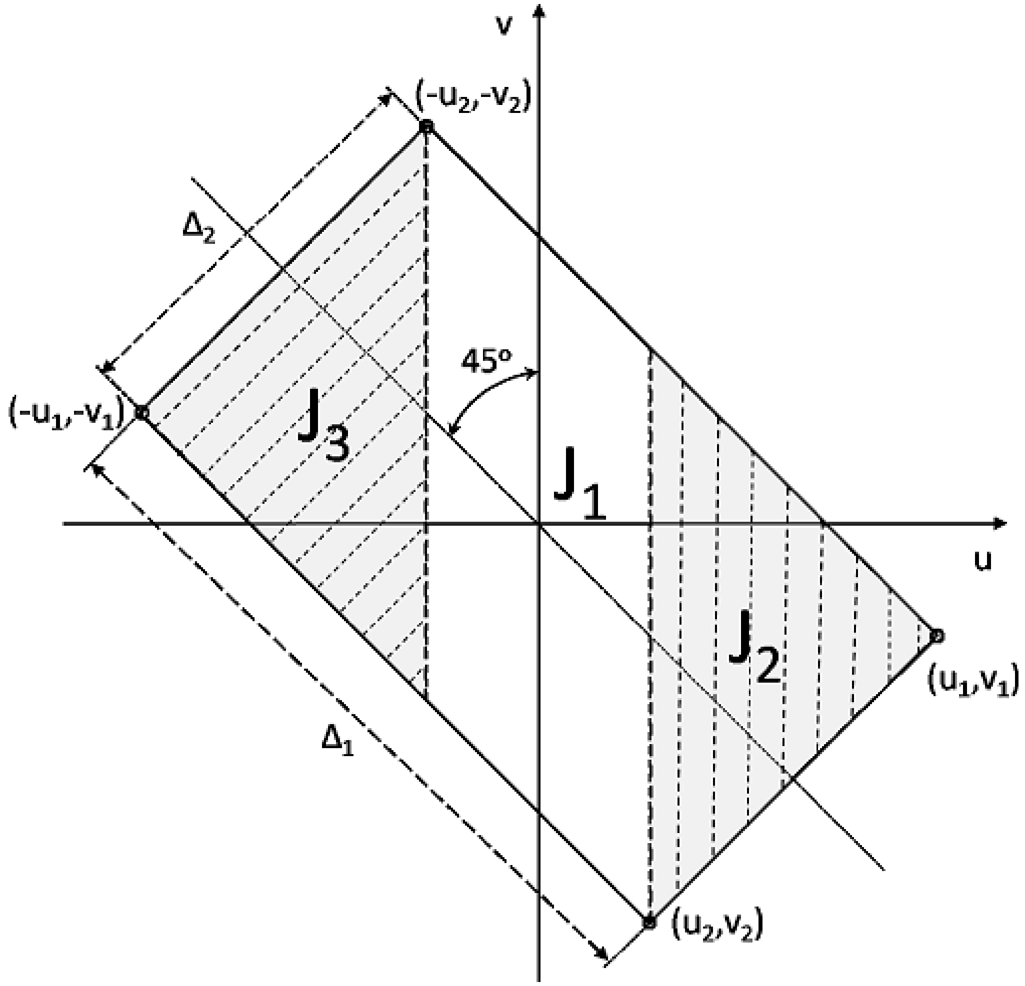


Fig. 12. Integration areas for calculation of the integral (58).