

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA STROJNÍHO INŽENÝRSTVÍ
ÚSTAV MATEMATIKY

FACULTY OF MECHANICAL ENGINEERING
INSTITUTE OF MATHEMATICS

A METHOD FOR THE VISUALIZATION OF HIGH PHASE
GRADIENTS IN A MICROSCOPIC IMAGE

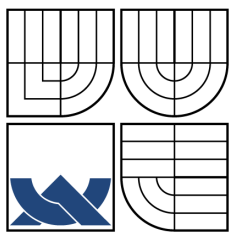
METODA PRO ZVIDITELNĚNÍ VYSOKÝCH FÁZOVÝCH GRADIENTŮ
V MIKROSKOPICKÉM ZOBRAZENÍ

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

HANA DRUCKMÜLLEROVÁ

BRNO 2008



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA STROJNÍHO INŽENÝRSTVÍ
ÚSTAV MATEMATIKY
FACULTY OF MECHANICAL ENGINEERING
INSTITUTE OF MATHEMATICS

A METHOD FOR THE VISUALIZATION OF HIGH PHASE GRADIENTS IN A MICROSCOPIC IMAGE

METODA PRO ZVIDITELNĚNÍ VYSOKÝCH FÁZOVÝCH GRADIENTŮ
V MIKROSKOPICKÉM ZOBRAZENÍ

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

HANA DRUCKMÜLLEROVÁ

VEDOUCÍ PRÁCE
SUPERVISOR

Doc. RNDr. RADIM CHMELÍK, Ph.D.

BRNO 2008

Abstrakt

Holografická mikroskopie je nekonvenční mikroskopická technika, vhodná zvláště pro vzorky s malou optickou hustotou, která umožňuje zviditelnit index lomu pozorovaných objektů. Na Ústavu fyzikálního inženýrství Fakulty strojního inženýrství VUT v Brně byl sestaven unikátní transmisní digitální holografický mikroskop (TDHM). Pořízené snímky (hologramy) jsou zpracovány metodou založenou na Fourierově transformaci, čímž je zrekonstruována intenzita a fáze světelné vlny procházející pozorovaným objektem. Fáze popisuje index lomu a tloušťku pozorovaného objektu. V místech, kde se mění index lomu nebo tloušťka, dochází i ke změně fáze. Úkolem této bakalářské práce bylo najít metodu pro zviditelnění míst s vysokým gradientem fáze. Podařilo se vytvořit metodu, která nevyžaduje navazování fáze, a proto je vhodná pro libovolné obrazy pořízené TDHM. Tato metoda byla implementována do počítačového programu **Gradient3D**, který kromě výpočtu gradientu ve dvou a třech rozměrech umožňuje i vytváření barevných obrazů, jejichž složkami jsou kombinace intenzity, fáze a gradientu. Program též umožňuje odstranění falešných gradientů v místech s nízkou intenzitou, kde je hodnota fáze nespolehlivá. Program byl testován na několika souborech hologramů pořízených TDHM při pozorování biologických vzorků.

Summary

Holographic microscopy is an unconventional microscopy technique suitable especially for transparent samples. It enables to visualize the refractive index of observed objects. A unique transmitted-light digital holographic microscope (TDHM) has been constructed at Institute of Physical Engineering, Faculty of Mechanical Engineering, Brno University of Technology. Holograms captured by the microscope are processed by means of a technique based on the Fourier transform in order to reconstruct the intensity and phase of the light waves passing through the observed object. The phase describes the object refractive index and thickness. In places where the object refractive index or thickness changes, also the phase does. The task of this bachelor thesis was to find a method for visualizing places with high phase gradient. A gradient computation method which was created does not require phase unwrapping and is therefore suitable generally for any images. The method was implemented in a computer software called **Gradient3D**, which enables not only to compute the phase gradient in two and three dimensions, but also to create color images composed from combinations of intensity, phase and gradient. It also contains methods for handling places with low reconstructed intensity where the phase value is unreliable and usually causes false phase gradients. The program has been tested on several image sets from the TDHM capturing biological specimens.

klíčová slova

zpracování obrazu, digitální holografický mikroskop, fázový gradient, optická mikroskopie

key words

image processing, digital holographic microscope, phase gradient, optical microscopy

DRUCKMÜLLEROVÁ, H.: *A method for the visualization of high phase gradients in a microscopic image*, Brno, Brno University of Technology, Faculty of Mechanical Engineering, 2007 (31 pages). Supervisor doc. RNDr. Radim Chmelík, PhD.

I declare that I have written the bachelor thesis *A method for the visualization of high phase gradients in a microscopic image* on my own according to the instructions of my bachelor thesis supervisor doc. RNDr. Radim Chmelík, PhD., and using the sources listed in references.

May 5, 2008

Hana Druckmüllerová

I would like to express thanks to my bachelor thesis supervisor doc. RNDr. Radim Chmelík, PhD., from the Institute of Physical Engineering, Faculty of Mechanical Engineering, Brno University of Technology, for valuable ideas on my work and for the time he spent introducing me to microscopy.

I would like to thank Ing. Hana Janečková from the Institute of Physical Engineering, Faculty of Mechanical Engineering, Brno University of Technology for the hours she spent with me at the microscope.

For biological samples I would like to thank Mgr. Hana Hříbková from the Department of Biology, Faculty of Medicine, Masaryk University.

Hana Druckmüllerová

Contents

1	Introduction	7
2	Problem formulation	7
2.1	Transmitted-light digital holographic microscope	7
2.1.1	Optical set-up of the microscope	7
2.2	Image reconstruction	8
3	Image matrices	11
3.1	Intensity and phase	11
3.2	Phase gradient	12
3.2.1	Numerical methods of 2D gradient computation	12
3.2.2	Numerical methods of 3D gradient computation	13
4	Color spaces	14
4.1	RGB color space	14
4.2	HSL color space	15
4.3	HSL to RGB conversion	15
5	<i>Gradient3D</i> software	17
5.1	Input files	17
5.2	Loading complex amplitude	18
5.3	Gradient computation	18
5.3.1	2D gradient computation	18
5.3.2	3D gradient computation, phase alignment	18
5.4	Pixel information	19
5.5	Displaying intensity in a selected area	21
5.6	Display setting – gamma correction	21
5.7	Handling places with low intensity	21
5.7.1	Gradient masking in dependence on intensity	22
5.7.2	Gradient multiplying with intensity	23
5.8	Output image characteristics	23
5.8.1	Image file names	23
5.8.2	Image resampling	23
5.8.3	Image cut	23
5.9	Output image types	23
5.9.1	Intensity, phase	23
5.9.2	Images without phase gradient	24
5.9.3	Images with phase gradient	24
6	Conclusion	26
	References	27
A	Used symbols	28
B	Enclosed CD	28
C	Figures	29

1 Introduction

Holographic microscopy is an unconventional microscopy technique suitable especially for transparent samples. It enables to visualize the refractive index of observed objects. A unique transmitted-light digital holographic microscope (TDHM) has been constructed at Institute of Physical Engineering, Faculty of Mechanical Engineering, Brno University of Technology. Holograms captured by the microscope are processed by means of a method based on the Fourier transform in order to reconstruct the intensity and phase of the light waves passing through the observed object. The intensity describes the object transparency, the phase its refractive index and thickness. In places where the object refractive index or thickness changes, also the phase does.

The task of this bachelor thesis is to find a method for visualizing places with high phase gradient, deal with the problem of phase unwrapping and the problem of destructive interference where the observed phase value is vague. The task was also to implement this method in a software and test it on experimental data consisting of the complex amplitude matrix captured by means of the digital holographic microscope in various sample depths.

I have created a gradient computation method which does not require phase unwrapping and is therefore suitable generally for any images. I implemented this method in a software called **Gradient3D**, which enables not only to compute the phase gradient in two and three dimensions, but also to create color images composed from combinations of intensity, phase and gradient. It also contains methods for handling places with low reconstructed intensity where the phase value is unreliable and usually causes false phase gradients. The program has been tested on several image sets from the TDHM capturing biological specimens.

2 Problem formulation

2.1 Transmitted-light digital holographic microscope

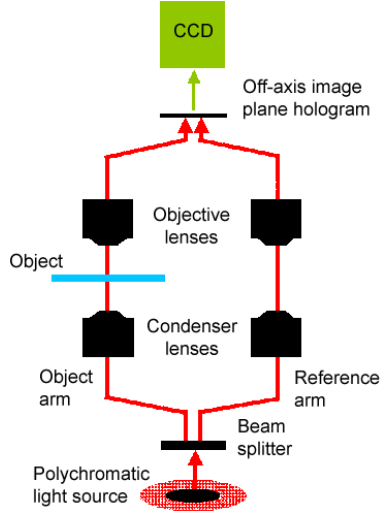
Holographic microscopy enables to observe transparent or semitransparent objects which can be hardly observed with classical optical microscopes because of very low light absorption. The transmitted-light holographic microscope creates hologram according to refractive index of observed objects and therefore it brings qualitatively new approach to visualization of light phase changes in microscopic specimens based on different principles from phase or Nomarski contrast.

The system used in the unique microscope [1], see fig. 1(b) constructed and used in the Laboratory of Optical Microscopy at Institute of Physical Engineering, Faculty of Mechanical Engineering, Brno University of Technology is a holographic microscope working in transmission mode – transmitted-light digital holographic microscope (TDHM).

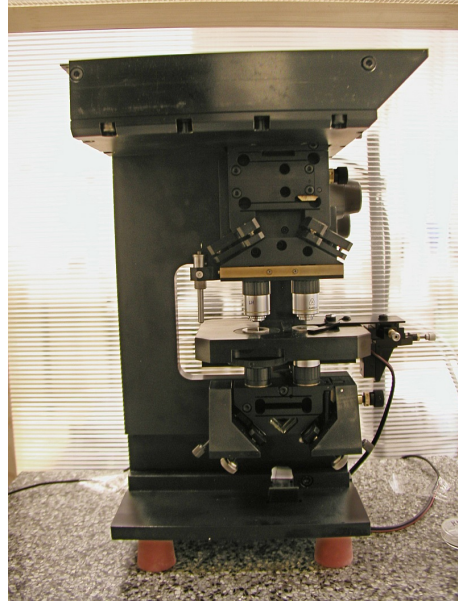
2.1.1 Optical set-up of the microscope

The light source of this microscope is low coherent, which means that polychromatic light source can be used, although better results are obtained when only red light is used (which the microscope was constructed for). The light is split into two beams which pass through optically equivalent condensers. The object arm passes through the observed object whereas the reference arm passes through a reference specimen. This is to simulate the background of the observed object. For a biological specimen with cells in

a water solution on a glass slide covered with a cover glass, similar glasses with water in between are used. Both arms then pass through optically equivalent objective lenses. The beams interfere to form an image-plane hologram which is captured by a CCD camera, see fig. 1(a).



(a) TDHM construction.
(Adapted from [1])



(b) DHM2 microscope in the Laboratory of Optical Engineering, IPE, FME, BUT.

Figure 1: TDHM microscope.

2.2 Image reconstruction

Definition 2.1. Let us denote o and r the *complex amplitude*¹ of the object and reference waves [2], x and y are spatial coordinates in the hologram plane and $\pm k_x, \pm k_y$ are the coordinates of plane-wave wavevector \vec{k} in the direction of the object wave or reference wave respectively. Then the *intensity* of the resultant wave after their interference is

$$i = |oe^{ik_x x + ik_y y} + re^{-ik_x x - ik_y y}|^2. \quad (1)$$

Theorem 2.2. The intensity i can be also written as

$$i = |o|^2 + |r|^2 + or^* e^{2ik_x x + 2ik_y y} + o^* r e^{-2ik_x x - 2ik_y y}, \quad (2)$$

where z^* denotes complex conjugate of z complex number.

Proof. Let us first compute the complex conjugate of $oe^{ik_x x + ik_y y}$. Assuming that for suitable a, α

$$o = a(\cos \alpha + i \sin \alpha), \quad (3)$$

¹The standard term used in physics differs from the mathematical concept of amplitude. I will keep the physical terminology, where complex amplitude is a complex number.

$oe^{ik_x x + ik_y y}$ can be written as

$$\begin{aligned} oe^{ik_x x + ik_y y} &= a(\cos \alpha + i \sin \alpha)(\cos(k_x x + k_y y) + i \sin(k_x x + k_y y)) \\ &= a(\cos(\alpha + k_x x + k_y y) + i \sin(\alpha + k_x x + k_y y)). \end{aligned} \quad (4)$$

Then the complex conjugate of $oe^{ik_x x + ik_y y}$ can be computed as

$$\begin{aligned} (oe^{ik_x x + ik_y y})^* &= a(\cos(\alpha + k_x x + k_y y) + i \sin(\alpha + k_x x + k_y y))^* \\ &= a(\cos(\alpha + k_x x + k_y y) - i \sin(\alpha + k_x x + k_y y)) \\ &= a(\cos(-\alpha - k_x x - k_y y) + i \sin(-\alpha - k_x x - k_y y)) \\ &= a(\cos(-\alpha) + i \sin(-\alpha))(\cos(-k_x x - k_y y) + i \sin(-k_x x - k_y y)) \\ &= o^* e^{-ik_x x - ik_y y}. \end{aligned} \quad (5)$$

Analogically,

$$(re^{-ik_x x - ik_y y})^* = r^* e^{ik_x x + ik_y y}. \quad (6)$$

Then the intensity is

$$\begin{aligned} i &= |oe^{ik_x x + ik_y y} + re^{-ik_x x - ik_y y}|^2 \\ &= (oe^{ik_x x + ik_y y} + re^{-ik_x x - ik_y y})(oe^{ik_x x + ik_y y} + re^{-ik_x x - ik_y y})^* \\ &= (oe^{ik_x x + ik_y y} + re^{-ik_x x - ik_y y})(o^* e^{-ik_x x - ik_y y} + r^* e^{ik_x x + ik_y y}) \\ &= |o|^2 + |r|^2 + o r^* e^{2ik_x x + 2ik_y y} + o^* r e^{-2ik_x x - 2ik_y y}, \end{aligned} \quad (7)$$

which was to be proved. \square

The aim is to compute the o , the complex amplitude of the observed object.

Definition 2.3. The *Fourier transform* [4] of a function $f(x, y) : \mathbb{R}^2 \rightarrow \mathbb{C}^2$ is a function $\mathcal{F}\{f(x, y)\} = F(m, n) : \mathbb{R}^2 \rightarrow \mathbb{C}^2$ defined by the formula

$$F(m, n) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) e^{-imx - iny} dx dy. \quad (8)$$

Theorem 2.4. The *inverse Fourier transform* [4] of a function $F(m, n) : \mathbb{R}^2 \rightarrow \mathbb{C}^2$ is a function $\mathcal{F}^{-1}\{F(m, n)\} = f(x, y) : \mathbb{R}^2 \rightarrow \mathbb{C}^2$

$$f(x, y) = \frac{1}{4\pi^2} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} F(m, n) e^{imx + iny} dm dn. \quad (9)$$

Definition 2.5. The *convolution* [4] of functions $f_1(x, y), f_2(x, y) : \mathbb{R}^2 \rightarrow \mathbb{C}^2$ is

$$f_1(x, y) * f_2(x, y) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f_1(\xi_1, \xi_2) f_2(x - \xi_1, y - \xi_2) d\xi_1 d\xi_2. \quad (10)$$

Theorem 2.6. [4]

$$\mathcal{F}\{f_1(x, y) f_2(x, y)\} = \frac{1}{4\pi^2} F_1(m, n) * F_2(m, n) \quad (11)$$

Definition 2.7. The Dirac impulse $\delta(m, n)$ [5] is a distribution which satisfies the following conditions:

$$\delta(m, n) = 0 \quad \forall [m, n] \in \mathbb{R}^2 - \{[0, 0]\}, \quad (12)$$

$$\int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \delta(m, n) dm dn = 1. \quad (13)$$

Theorem 2.8. The Fourier transform of a complex exponential function [3] is

$$\mathcal{F}\{e^{im_0x+in_0y}\} = 4\pi^2\delta(m - m_0, n - n_0), \quad m_0, n_0 \in \mathbb{R}. \quad (14)$$

Apart from the object hologram, it is necessary to capture a hologram of the background. Applying Fourier transform on the holographic object image, a spectrum with a significant maximum is obtained.

$$\begin{aligned} \mathcal{F}\{or^* e^{2ik_x x + 2ik_y y}\} &= \frac{1}{4\pi^2} \mathcal{F}\{or^*\} * \mathcal{F}\{e^{2ik_x x + 2ik_y y}\} \\ &= \frac{1}{4\pi^2} \mathcal{F}\{or^*\} * 4\pi^2 \delta(m - 2k_x, n - 2k_y) \\ &= \mathcal{F}\{or^*\} * \delta(m - 2k_x, n - 2k_y) \end{aligned} \quad (15)$$

The maximum of the amplitude spectra is shifted by $(2k_x, 2k_y)$ from the zero frequency and represents the carrier frequency. In the Fourier image of the holographic object hologram, the Fourier image of the observed object is mapped around this maximum.

Definition 2.9. The *Hanning window function* [6] with perimeter $r \in \mathbb{R}^+$ is a function $w(m, n) : \mathbb{R}^2 \rightarrow \mathbb{R}$ defined by the formula

$$w(m, n) = \begin{cases} \frac{1}{2} + \frac{1}{2} \cos \frac{\pi \sqrt{m^2 + n^2}}{r}, & \sqrt{m^2 + n^2} \leq r, \\ 0, & \text{otherwise.} \end{cases} \quad (16)$$

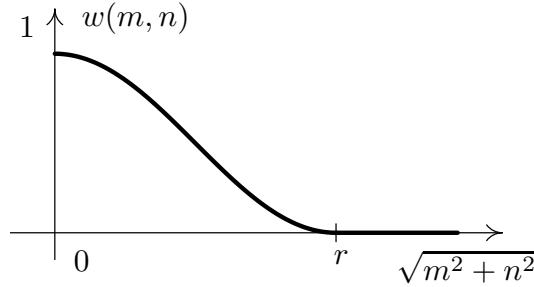


Figure 2: Graph of the Hanning window function.

The neighborhood of the amplitude spectrum maxima in the Fourier images of both the background hologram and the object hologram is cropped and the coordinate system origin is shifted to this maximum. Both images are multiplied with the Hanning window function values, the perimeter of the function is 90 pixels, which was chosen experimentally. Inverse Fourier transform is applied on both images to obtain the complex amplitude of the background ($o_0 r^*$) and the holographic object (or^*). As the hologram of the object also contains the background, this is removed by dividing the object complex amplitude with the background complex amplitude. Let us denote the resulting complex amplitude \tilde{o}' .

$$\tilde{o}' = \frac{or^*}{o_0 r^*} \quad (17)$$

3 Image matrices

In the next chapters, all continuous quantities will be written with a tilde ($\tilde{}$), whereas discrete quantities will be written without any sign.

3.1 Intensity and phase

The complex amplitude $\tilde{o}'(x, y, z)$ is a mapping $\tilde{o}' : \mathbb{R}^3 \rightarrow \mathbb{C}$.

Definition 3.10. The *intensity* $\tilde{\rho}$ and *phase* \tilde{p} of complex amplitude \tilde{o}' in each point $[x, y, z] \in \mathbb{R}^3$ are numbers $\tilde{\rho} \in \mathbb{R}_0^+ = \langle 0, \infty \rangle$, $\tilde{p} \in \langle 0, 2\pi \rangle$ computed according to the following formulae [7]:

$$\tilde{\rho} = \tilde{o}' \tilde{o}'^* = \Re^2(\tilde{o}') + \Im^2(\tilde{o}'), \quad (18)$$

$$\cos \tilde{p} = \frac{\Re(\tilde{o}')}{\sqrt{\tilde{\rho}}}, \quad \sin \tilde{p} = \frac{\Im(\tilde{o}')}{\sqrt{\tilde{\rho}}}. \quad (19)$$

For $\tilde{o}' = 0$, intensity $\tilde{\rho} = 0$ and phase \tilde{p} is undefined.

Theorem 3.11. $\forall [x, y, z] \in \mathbb{R}^3 :$

$$\tilde{o}' = \sqrt{\tilde{\rho}}(\cos \tilde{p} + i \sin \tilde{p}). \quad (20)$$

Proof. $\sqrt{\tilde{\rho}}(\cos \tilde{p} + i \sin \tilde{p}) = \sqrt{\tilde{\rho}} \cos \tilde{p} + i \sqrt{\tilde{\rho}} \sin \tilde{p} = \Re(\tilde{o}') + i \Im(\tilde{o}') = \tilde{o}' \quad \square$

Definition 3.12. Let us have sets $A_1 = \{0, 1, \dots, w-1\}$, $A_2 = \{0, 1, \dots, h-1\}$, $A_3 = \{0, 1, \dots, d-1\}$, $w, h, d \in \mathbb{N}$. Let us denote $o'(x, y, z)$ the mapping $o' : A_1 \times A_2 \times A_3 \rightarrow \mathbb{C}$ which satisfies $o'(x, y, z) = \tilde{o}'(x, y, z) \quad \forall [x, y, z] \in A_1 \times A_2 \times A_3$. Mapping o' is called *discrete approximation of complex amplitude*. If it causes no confusion, it will be called complex amplitude as well.

Convention

o' is represented with *image matrix*, its elements $[x, y, z]$ are called *pixels*, the numbers x, y, z are called *coordinates*, x is the *line coordinate*, y the *column coordinate* and z is the *layer number*. w, h, d are image matrix *width*, *height* and *depth*. For each $z \in A_3$, the set $A_1 \times A_2 \times \{z\}$ is called a *layer*.

Definition 3.13. Similarly to def. 3.12, let us define the *discrete approximation of intensity and phase*. Let us have sets A_1, A_2, A_3 from def. 3.12. Let us denote $\rho(x, y, z)$ and $p(x, y, z)$ mappings $\rho : A_1 \times A_2 \times A_3 \rightarrow \mathbb{R}_0^+$, $p : A_1 \times A_2 \times A_3 \rightarrow \langle 0, 2\pi \rangle$ which satisfy $\rho(x, y, z) = \tilde{\rho}(x, y, z)$, $p(x, y, z) = \tilde{p}(x, y, z) \quad \forall [x, y, z] \in A_1 \times A_2 \times A_3$. If it causes no confusion, ρ will be called intensity and p phase.

Theorem 3.14. $\forall [x, y, z] \in A_1 \times A_2 \times A_3 :$

$$\rho = o' o'^* = \Re^2(o') + \Im^2(o'), \quad (21)$$

$$\cos p = \frac{\Re(o')}{\sqrt{\rho}}, \quad \sin p = \frac{\Im(o')}{\sqrt{\rho}}, \quad (22)$$

$$o' = \sqrt{\rho}(\cos p + i \sin p). \quad (23)$$

Proof. According to def. 3.12, 3.13, $\forall [x, y, z] \in A_1 \times A_2 \times A_3 : o'(x, y, z) = \tilde{o}'(x, y, z)$, $\rho(x, y, z) = \tilde{\rho}(x, y, z)$, $p(x, y, z) = \tilde{p}(x, y, z)$. Therefore formulae (18) – (20) apply for each $[x, y, z] \in A_1 \times A_2 \times A_3$ and p, ρ values in this pixel. \square

3.2 Phase gradient

Definition 3.15. For $a \in \mathbb{R}$ let us define the *integral part* $[a]$ as

$$[a] = \max\{n \in \mathbb{Z} : n \leq a\}. \quad (24)$$

Definition 3.16. Let us define binary operation *addition modulo 2π* , denoted \oplus , by the following formula

$$a \oplus b \equiv a + b \pmod{2\pi} = (a + b) - 2\pi \left\lceil \frac{a + b}{2\pi} \right\rceil, \quad a, b \in \mathbb{R}. \quad (25)$$

Although \tilde{o}' is continuous according to the physical principle, it is obvious that $\tilde{p} = \arg(\tilde{o}')$ is not continuous. It is due to the fact that \arg is the angular coordinate in Gaussian plane represented in Cartesian coordinate system, which is necessary for numerical computation. The aim is to compute the norm of phase gradient. Therefore, a method to cope with \tilde{p} discontinuity points, i.e. 2π skips must be introduced. As a numerically computed approximation of the phase gradient norm consists generally of phase differences, it is not necessary to perform phase unwrapping. The method to cope with the 2π skips I used in the program I have created can be generally written for each point $[x, y, z] \in \mathbb{R}^3$ as

$$\tilde{q}(x, y, z) = \tilde{p}(x, y, z) \oplus \pi, \quad (26)$$

$$\tilde{g}_1(x, y, z) = \|\text{grad } \tilde{p}(x, y, z)\|_2, \quad \tilde{g}_2(x, y, z) = \|\text{grad } \tilde{q}(x, y, z)\|_2, \quad (27)$$

$$\tilde{g}(x, y, z) = \min(\tilde{g}_1(x, y, z), \tilde{g}_2(x, y, z)), \quad (28)$$

where $\|\cdot\|_2$ denotes Euclidean norm in \mathbb{R}^3 .

If a 2π skip occurs at a certain place in the p image, it means that the phase values are slightly under 2π , equal zero or slightly higher than zero. Adding π in 2π modulo arithmetics, these values are around π and no skip occurs. Therefore, if no 2π skip occurs at a certain place, the \tilde{g}_1, \tilde{g}_2 values equal. If a 2π skip occurs, the lower of these gradient values is the correct one.

To simplify the descriptions, the norm of discrete approximation of phase gradient will be called phase gradient only if no confusion may be caused.

3.2.1 Numerical methods of 2D gradient computation

The **Gradient3D** software enables to compute phase gradient in layers. For each layer (i.e. for each $z \in A_3$), the phase p is assumed as a function of only two variables x and y . Let us denote

$$q(x, y) = p(x, y) \oplus \pi. \quad (29)$$

For computing g , the *norm of discrete approximation of 2D phase gradient*, the formulae implemented in the program are described in table 1.

Method name	Formula
XY	$g_1 = \sqrt{\left(\frac{p(x+1,y)-p(x-1,y)}{2}\right)^2 + \left(\frac{p(x,y+1)-p(x,y-1)}{2}\right)^2}$ $g_2 = \sqrt{\left(\frac{q(x+1,y)-q(x-1,y)}{2}\right)^2 + \left(\frac{q(x,y+1)-q(x,y-1)}{2}\right)^2}$ $g = \min(g_1, g_2)$
XY backward	$g_1 = \sqrt{(p(x,y) - p(x-1,y))^2 + (p(x,y) - p(x,y-1))^2}$ $g_2 = \sqrt{(q(x,y) - q(x-1,y))^2 + (q(x,y) - q(x,y-1))^2}$ $g = \min(g_1, g_2)$
XY forward	$g_1 = \sqrt{(p(x,y) - p(x+1,y))^2 + (p(x,y) - p(x,y+1))^2}$ $g_2 = \sqrt{(q(x,y) - q(x+1,y))^2 + (q(x,y) - q(x,y+1))^2}$ $g = \min(g_1, g_2)$

Table 1: Numerical methods of 2D gradient norm computation.

3.2.2 Numerical methods of 3D gradient computation

The program also enables to work with three dimensional matrices of complex amplitude. If it makes sense to compute the gradient of \tilde{p} as a function of three variables x, y, z , the 3D phase gradient procedures should be used to visualize better spatial phase changes. The numerical methods of gradient computation assume that no phase changes were caused by microscope refocusing. If this assumption is not fulfilled, the program makes possible to align the phase in all images. The implemented method of phase alignment computes the arithmetic mean values of phase in a selected rectangle (with W_a width, H_a height and $[L_a, T_a]$ the top left pixel coordinates) and then adds a suitable constant to each layer phase values so that afterwards, the mean value in the rectangle in all layers is a number φ entered by the user.

$$\bar{p}_z = \frac{1}{W_a H_a} \sum_{x=L_a}^{L_a+W_a-1} \sum_{y=T_a}^{T_a+H_a-1} p(x, y, z), \quad z = 0, 1, \dots, d-1 \quad (30)$$

$$s_z = \varphi - \bar{p}_z, \quad z = 0, 1, \dots, d-1 \quad (31)$$

For each pixel with $[x, y, z]$ coordinates

$$\hat{p}(x, y, z) = p(x, y, z) \oplus s_z. \quad (32)$$

Let us denote

$$\hat{q}(x, y, z) = \hat{p}(x, y, z) \oplus \pi. \quad (33)$$

The distance of in-focus planes of the neighboring layers may vary for different input files. Therefore, the *weight ω of the z -axis difference* must be set by the user. As the in-focus plane distance is usually larger than the distance of places captured in neighboring pixels in one layer, ω should be set lower than 1. The formulae implemented in the program are described in table 2.

Method name	Formula
XYZ	$g_1 = \sqrt{\left(\frac{\hat{p}(x+1,y,z)-\hat{p}(x-1,y,z)}{2}\right)^2 + \left(\frac{\hat{p}(x,y+1,z)-\hat{p}(x,y-1,z)}{2}\right)^2 + \omega\left(\frac{\hat{p}(x,y,z+1)-\hat{p}(x,y,z-1)}{2}\right)^2}$ $g_2 = \sqrt{\left(\frac{\hat{q}(x+1,y,z)-\hat{q}(x-1,y,z)}{2}\right)^2 + \left(\frac{\hat{q}(x,y+1,z)-\hat{q}(x,y-1,z)}{2}\right)^2 + \omega\left(\frac{\hat{q}(x,y,z+1)-\hat{q}(x,y,z-1)}{2}\right)^2}$ $g = \min(g_1, g_2)$
XYZ diagonal	$g_1 = \sqrt{\frac{(\hat{p}(x,y,z)-\hat{p}(x-1,y-1,z))^2}{2} + \frac{(\hat{p}(x,y,z)-\hat{p}(x-1,y+1,z))^2}{2} + \omega(\hat{p}(x,y,z)-\hat{p}(x,y,z-1))^2}$ $g_2 = \sqrt{\frac{(\hat{p}(x,y,z)-\hat{p}(x-1,y-1,z))^2}{2} + \frac{(\hat{p}(x,y,z)-\hat{p}(x-1,y+1,z))^2}{2} + \omega(\hat{p}(x,y,z)-\hat{p}(x,y,z-1))^2}$ $g = \min(g_1, g_2)$
XYZ backward	$g_1 = \sqrt{(\hat{p}(x,y,z)-\hat{p}(x-1,y,z))^2 + (\hat{p}(x,y,z)-\hat{p}(x,y-1,z))^2 + \omega(\hat{p}(x,y,z)-\hat{p}(x,y,z-1))^2}$ $g_2 = \sqrt{(\hat{q}(x,y,z)-\hat{q}(x-1,y,z))^2 + (\hat{q}(x,y,z)-\hat{q}(x,y-1,z))^2 + \omega(\hat{q}(x,y,z)-\hat{q}(x,y,z-1))^2}$ $g = \min(g_1, g_2)$
XYZ forward	$g_1 = \sqrt{(\hat{p}(x,y,z)-\hat{p}(x+1,y,z))^2 + (\hat{p}(x,y,z)-\hat{p}(x,y+1,z))^2 + \omega(\hat{p}(x,y,z)-\hat{p}(x,y,z+1))^2}$ $g_2 = \sqrt{(\hat{q}(x,y,z)-\hat{q}(x+1,y,z))^2 + (\hat{q}(x,y,z)-\hat{q}(x,y+1,z))^2 + \omega(\hat{q}(x,y,z)-\hat{q}(x,y,z+1))^2}$ $g = \min(g_1, g_2)$

Table 2: Numerical methods of 3D gradient norm computation.

4 Color spaces

There are three output matrices of the image reconstruction procedure and gradient computation – the intensity matrix $\rho(x, y, z)$, the phase matrix $p(x, y, z)$ and the gradient matrix $g(x, y, z)$. Intensity and gradient are non-negative real numbers, phase are real numbers in $(0, 2\pi)$ interval. For the comprehension of observed objects, the matrices must be displayed graphically. For even better comprehension, some of the matrices can be composed into color images.

4.1 RGB color space

The RGB color space is the basic color space used in computer graphics for the representation of any color on the computer display and it is similar to the way human eye perceives colors. Each color is represented as a combination of the red, green, and blue color component. These colors are called primary colors. The RGB space is an additive one, because secondary colors (magenta, cyan, yellow) are created as a sum of two primary colors. The combination of red, green and blue in full intensities produces white.

Figure 3 shows the cube describing all colors which can be displayed with the RGB color space. The component values are normalized. The red, green and blue component values are at three vertices, cyan, magenta, and yellow are the other vertices, black is at the origin and white is at the vertex farthest from the origin. All gray colors extend from the black to the white vertex on the cube main diagonal. Each color is represented with a point on the walls or inside the cube.

The disadvantage of the RGB color space is that none of the color components is cyclic. This is why the RGB color space is not suitable for the representation of the complex amplitude reconstruction.

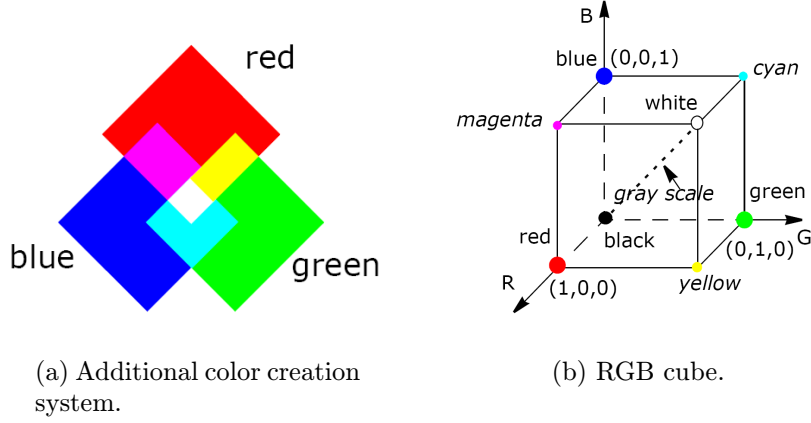


Figure 3: RGB color space. [10]

4.2 HSL color space

The HSL color space (hue, saturation, lightness) enables more intuitive approach to colors. Hue H defines the color shade. It is represented as angle, e.g. from 0° representing red through green, blue and all intermediary colors to 360° which is red again. This periodicity makes the hue component suitable for periodic quantities (e.g. light phase) representation. Saturation S indicates how much the color differs from gray. Saturation values lie within the $\langle 0, 1 \rangle$ interval, 0 means a neutrally gray color, 1 is the most saturated color possible for the given lightness. The lightness component L can be described as illumination intensity. It varies from 0 (black) to 1 (white). The 3D representation of the HSL color space is a double cone, however, it is often drawn as a double hexcone (see fig. 4(a)). All gray shades from black to white lie on the cone axis. Three base vertices represent the primary colors (red, green, blue), the other ones the secondary colors (cyan, magenta, yellow). Apart from the double cone, the HSL color space can be illustrated on the color dialog in MS Windows, fig. 4(b). Moving in the main color panel horizontally, hue is set, vertical movement sets saturation. Lightness is set with the narrow bar on the right.

When stored in image matrices in computers, the H , S , L values are expressed by integer numbers in any interval depending on the image dynamic range.

4.3 HSL to RGB conversion

Definition 4.17. Let us define binary operation *addition modulo 1*, denoted \boxplus , by the following formula

$$a \boxplus b \equiv a + b \pmod{1} = (a + b) - [a + b], \quad a, b \in \mathbb{R}. \quad (34)$$

Let us suppose a $[H, S, L]$ triplet defining the hue, saturation and lightness values is given, $H \in \langle 0, 360 \rangle$, $S, L \in \langle 0, 1 \rangle$. According to [10] and [11], the $[R, G, B] \in \langle 0, 1 \rangle \times \langle 0, 1 \rangle \times \langle 0, 1 \rangle$ triplet consisting of the red, green and blue values is computed as follows. If $S = 0$, the color is gray, i.e. $R = G = B = L$. The hue value is ignored in this case. Even if $S = 0$, the following formulae can be used.

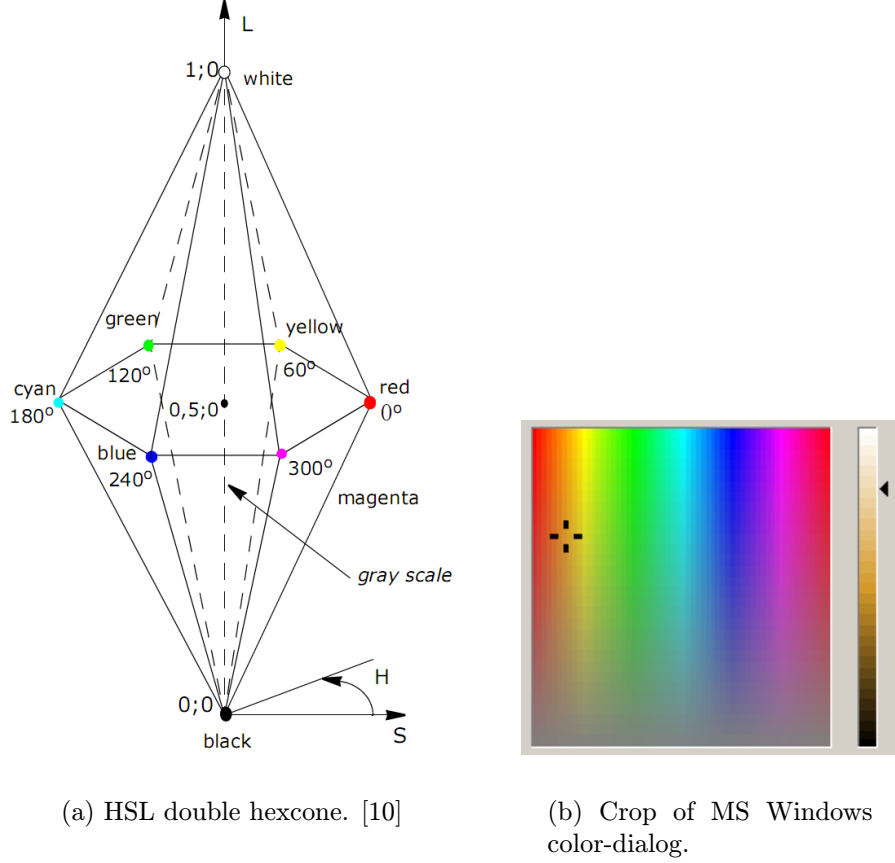


Figure 4: HSL color space representation.

$$E = \begin{cases} L + LS, & L < \frac{1}{2} \\ L + S - LS, & L \geq \frac{1}{2} \end{cases} \quad (35)$$

$$D = 2L - E \quad (36)$$

$$H_1 = \frac{H}{360} \quad (37)$$

$$T_R = H_1 \boxplus \frac{1}{3}, \quad T_G = H_1, \quad T_B = H_1 \boxplus \frac{2}{3} \quad (38)$$

For each color component C , $C \in \{R, G, B\}$

$$C = \begin{cases} D + 6T_C(E - D), & T_C < \frac{1}{6}, \\ E, & \frac{1}{6} \leq T_C < \frac{1}{2}, \\ D + 6\left(\frac{2}{3} - T_C\right)(E - D), & \frac{1}{2} \leq T_C < \frac{2}{3}, \\ D, & \text{otherwise.} \end{cases} \quad (39)$$

However, the described conversion is still simplified. It does not take into consideration that human eyes have different sensitivity in different parts of the spectrum [8], [9]. We perceive most intensively light on 555 nm, which is a yellow-green color. If this sensitivity is denoted 100 %, the sensitivity in red (640 nm) is only 17.5 % and in blue (480 nm)

14%. If the HSL to RGB conversion should preserve the subjective color lightness, the conversion must be performed with respect to human eye sensitivity. This is a much more complicated method, which is implemented as the **HLSToRGB** procedure in the **IIPP** dynamic library [10], which I used for the conversion.

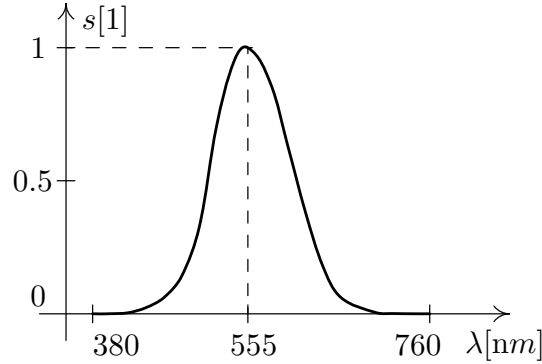


Figure 5: Graph of human eye sensitivity. (Data from [8])

5 *Gradient3D* software

The **Gradient3D** program is intended for numerical phase gradient computation and displaying. It has several additional functions, such as gradient and complex amplitude displaying and composing these matrices into color images. It has a simple graphical interface, the use of which will be described in this section.

5.1 Input files

The input of the program is the complex amplitude in two separate files, one with its real component and one with its imaginary component. The file names must be ***r.txx** for the real component and ***i.txx** for the imaginary one. The MATLAB software, which the image reconstruction is performed with, does not enable to save matrices into files in the way they are stored in computer memory. Therefore, rather inconvenient and large files containing numbers saved as text must be used. The input file structure is as follows:

```
Width    Height    Depth
[0,0,0] value
[1,0,0] value
...
```

Width denotes layer width, **Height** layer height and **Depth** is the image depth, i.e. number of single images stored in one file. The program enables to process these layers separately (2D gradient procedures are used) or as a set of sections of the object one above another (3D gradient procedures are used). Apart from the numbers describing the image size, each number must be written on a separate line. This turned out to be a problem, because MATLAB uses UNIX conventions for line breaks. To convert MATLAB text files containing matrices to the form suitable for the **Gradient3D** program, it is necessary to use a very simple converter which is placed in the **Konvertor** directory on the enclosed CD. It changes the file extension from **asc** to **txx**.

5.2 Loading complex amplitude

After running the program, the real component matrix file is opened with the **Load real component** button. Then the imaginary component is loaded automatically from a file with a corresponding name. The program loads the complex amplitude from these two files, computes the intensity and phase matrices and displays them in the two left images. The phase is displayed in gray-scale with black as 0 and white as 2π . Layers are enumerated starting from zero and the requested layer is displayed after the button with the layer number is pressed.

5.3 Gradient computation

5.3.1 2D gradient computation

If the input files consist of single images, the object moved between particular images or if for any other reason it makes no sense to compute the phase gradient in three dimensions, the 2D gradient computing procedures should be used. There are three available methods, **XY**, **XY backward** and **XY forward**, which differ in the used formula. These formulae are listed in table 2, page 14. The **XY** method creates less contrasty, less sharp images and is therefore suitable for more noisy images. The results of the **XY backward** and the **XY forward** procedures are sharper with enhanced areas with extremely high phase gradients.

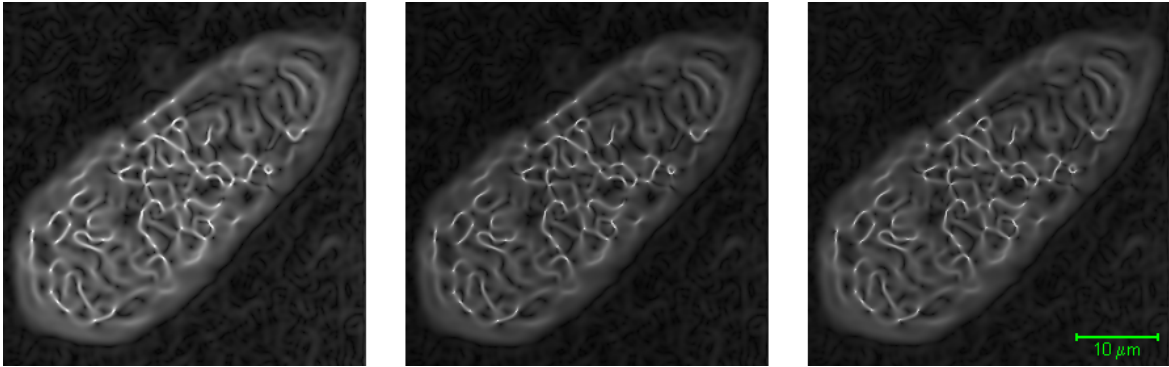


Figure 6: Crops of the same image processed with the XY, XY backward and XY forward procedure (left to right), $\gamma_g = 2.5$ (see def. 5.18).

In the **XY** method, the gradient value is defined for each pixel, even for the edge ones, because the formula changes at the edge pixels so that it uses available phase values only. The **XY backward** and the **XY forward** methods do not change the formula at image edges. Therefore the gradient values in the uppermost ($y = 0$) and most left ($x = 0$) pixels are undefined, i.e. they are equal to zero for the former method and in the lowermost ($y = h - 1$) and most right ($x = w - 1$) for the latter one.

5.3.2 3D gradient computation, phase alignment

If the input files consist of object sections with gradually moving in-focus plane, the three dimensional gradient computation methods should be used to enhance better the spatial phase changes. The 3D gradient procedures assume that no phase changes in the background were caused by microscope refocusing. If this assumption is not fulfilled, the

program makes possible to align the phase in all images. The mathematical principle of phase alignment was described in section 3.2.2.

For performing phase alignment, the **Display area** check-box must be checked and the **phase alignment** item in the **Displayed area is for ...** group must be selected. A green rectangle appears in both intensity and phase images. Its position can be changed by mouse dragging and its size can be changed by dragging its aspects and vertices. Its precise position and size can also be set by entering values in corresponding edits bellow the images. The new average phase value φ in the rectangle in all layers is entered in the **New average phase** edit. The phase alignment is performed after the **Align phase** button is pressed.

This procedure does not bring the required result if a 2π skip appears in the selected area in any of the layers. Furthermore, the rectangle should be placed out of any structures in the image, on the background of the observed object. However, applying phase shift by a constant value for each layer does not deteriorate the image and if the first phase alignment was not performed successfully, it can be repeated once again with or without changing its parameters.

After the phase alignment, it is possible to compute the 3D phase gradient by means of one of the **XYZ**, **XYZ backward**, **XYZ forward** or **XYZ diagonal** procedures. The numerical methods used by these procedures were described in table 2, page 14. Similarly to the **XY** method, the **XYZ** method creates less contrasty, less sharp images, whereas the results of the other procedures are sharper with enhanced areas with extremely high phase gradients. The purpose of the diagonal method was to find out if the gradient computed from the central pixel $p(x, y, z)$, the pixel in the previous layer $p(x, y, z - 1)$ and the pixels to the left $p(x - 1, y, z)$ and up $p(x, y - 1, z)$ is the same as the gradient computed from the central pixel $p(x, y, z)$, the pixel in the previous layer $p(x, y, z - 1)$ and the pixels to the up left $p(x - 1, y - 1, z)$ and down left $p(x - 1, y + 1, z)$. It has turned out that the results of these two methods differ only slightly, especially in areas with extremely low intensity, see fig. 8.

The 3D gradient computation methods can be used not only on image sets with gradually moving in-focus plane, but also on image sets where the z axis is replaced with the time axis. The images can display the observed object in the course of time. It is necessary that the movements between the neighboring images are slight only, otherwise the image will be impossible to interpret.

In the **XYZ** method, the gradient value is defined for each pixel, even for the edge ones, because the formula changes at the edge pixels so that it uses available phase values only. The other methods do not change the formula at image edges. Therefore for the **XYZ diagonal** and **XYZ backward** method, the gradient values in the uppermost ($y = 0$) and most left ($x = 0$) pixels and in the lowest layer ($z = 0$) are undefined, i.e. they are equal to zero. For the **XYZ forward** method, the gradient values are undefined in the lowermost ($y = h - 1$) and most right ($x = w - 1$) pixels and in the uppermost layer ($z = d - 1$).

The name of the gradient-computing method which was used for creating the gradient matrix is displayed in blue on the gradient computation button, the others are left black.

5.4 Pixel information

The program shows additional information about mouse cursor coordinates and the intensity, phase and gradient values in the particular pixel. Moving the mouse cursor on

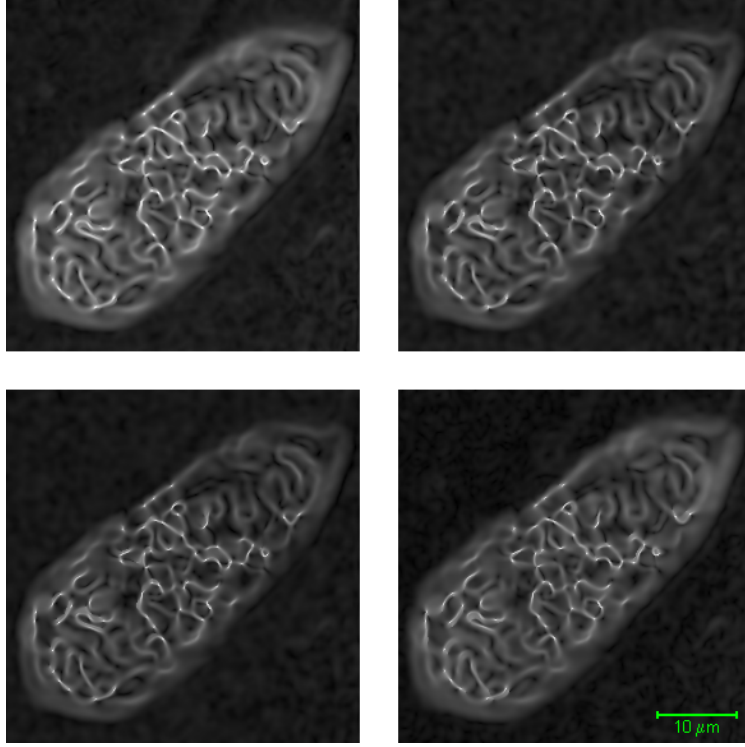


Figure 7: Crops of the same image processed with the XYZ, XYZ diagonal procedures (upper row from left to right), XYZ backward, XYZ forward procedures (lower row from left to right), $\gamma_g = 2.5$, $\omega = 0.1$ (see def. 5.18).



Figure 8: Negative of difference of crops of the same image processed with the XYZ backward and XYZ diagonal procedure, $\gamma_g = 1$, $\omega = 0.1$ (see def. 5.18).

any of the three images (intensity, phase, gradient), panels above each image show the matrix values in the pixel. The magnifier on the right of the program window shows magnified neighborhood of the mouse cursor and the panels next to it show the cursor coordinates in respect to the image. This works also if gradient is not computed and this image is disabled. Phase values are displayed in degrees so that the numbers are easier to understand.

5.5 Displaying intensity in a selected area

As the intensity image often has high pixel value range, some details may be difficult to see, especially in darker places. Therefore the **Gradient3D** software enables to display the intensity of a smaller area only. This area is displayed so that its brightest pixel is displayed white and its darkest pixel is black. This area can be set only if the **Display area** check-box is checked and the **intensity display** item in the **Displayed area is for ...** group is selected. Then a red rectangle is displayed in all computed images. Its position can be changed by mouse dragging and its size can be changed by dragging its aspects and vertices. Its precise position and size can also be set by entering values in corresponding edits below the images. The intensity is displayed in the selected area only, after a layer-selecting button is pressed. It is not necessary to change the current layer, it is also possible to press the button of the current layer to repaint it. The part of the intensity image outside of the rectangle is displayed in uniform green color. This option remains active even when the red rectangle is not visible (as the **Display area** check-box is unchecked), because the area is defined. The option is disabled when the intensity displaying area is the whole image.

5.6 Display setting – gamma correction

Gamma correction can be applied on both intensity and gradient images. It is a non-linear pixel value transformation which requires input image values in $\langle 0, 1 \rangle$ interval and it does not change the pixel value range. Therefore, it does not cause image degradation by pixel overflowing or underflowing. Since the phase image must be displayed linearly, it makes no sense to apply gamma correction on it.

Definition 5.18. Let us denote N the minimum pixel value in the image and M the maximum one. $\gamma \in \langle 0, \infty \rangle$ is called the *gamma value* (set by the user). The *gamma correction* is a mapping $\Gamma_\gamma : \langle N, M \rangle \rightarrow \langle 0, 1 \rangle$ which satisfies the following formula:

$$\Gamma_\gamma(v) = \left(\frac{v - N}{M - N} \right)^{\frac{1}{\gamma}} \quad (40)$$

Pixel values v are transformed into the $\langle 0, 1 \rangle$ interval and then powered to γ^{-1} . The γ^{-1} exponent is used so that the image changes are logical, i.e. $\gamma > 1$ increases overall image brightness, $\gamma < 1$ decreases it and $\gamma = 1$ does not change the image at all.

The gamma value for an image is entered into a corresponding edit. The gamma correction is applied after **Enter** is pressed or the current layer is changed. The gamma corrections are applied not only on the displayed images, but also on the intensity and gradient matrices of the images to be saved (see section 5.9.3).

5.7 Handling places with low intensity

In some places, the object beam intensity is very low, almost equal to zero, therefore its phase is vaguely defined. The hologram contrast is low there and the interference fringes even split at such places. Since the phase of the resulting wave after the interference is a difference of the object and reference ones, this phase is also vaguely defined. The reconstructed phase is therefore unreliable or even meaningless and it often changes very quickly. The computed phase gradient is the highest particularly in these places making the gradient image hard to understand. It is necessary to introduce a method

which removes these false gradient maxima from the images. The following methods are implemented in the **Gradient3D** software.

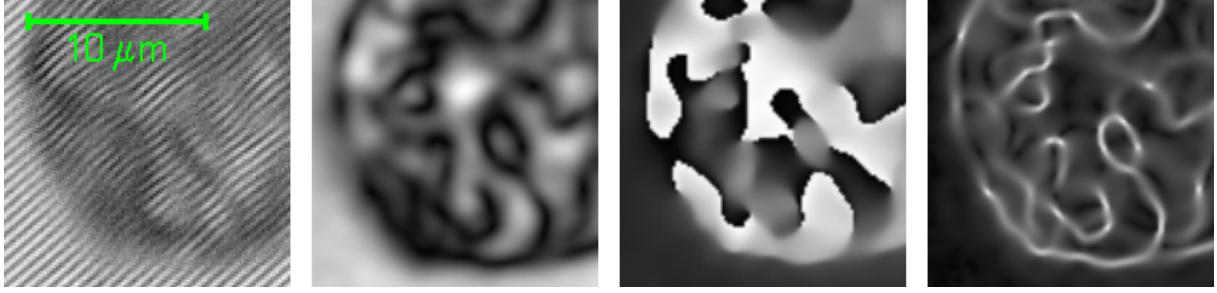


Figure 9: A place with low reconstructed intensity in an original hologram, intensity, phase and gradient XY ($\gamma_g = 2$).

5.7.1 Gradient masking in dependence on intensity

This method is based on applying a non-linear transformation on the gradient values in dependence on intensity. Each raw gradient value is multiplied with a coefficient c which is computed according to the following formula

$$c(\rho) = \begin{cases} 0, & \rho \leq T, \\ \frac{1}{2} \sin \left(\frac{\pi(\rho-T)}{P-T} - \frac{\pi}{2} \right) + \frac{1}{2}, & \rho \in (T, P), \\ 1, & \rho > P. \end{cases} \quad (41)$$

This function makes black all pixels whose intensity is lower than T and diminishes the brightness in all pixels with intensity in the (T, P) interval. All other pixels are left untouched. The T value is set in the **Totally ignored with intensity under** edits and P is set in the **Partially ignored with intensity under** edits. It is possible to set all of these values by entering these edits and pressing **Enter** or the corresponding buttons. It is also possible to use the displayed images to set these values. Clicking the left or right mouse button, the T or P value respectively is set to the intensity in the clicked pixel. Both T and P can be set to the minimum intensity value by pressing the **Reset** button. If the **Display ignored** check-box is checked, all pixels with $\rho \leq T$ are colored red and the $\rho \in (T, P)$ pixels are colored blue.

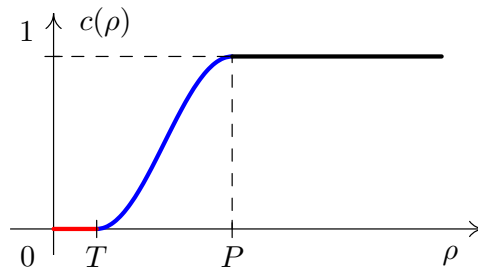


Figure 10: Graph of the $c(\rho)$ function.

This method is applied if the **Apply masking in dependence on intensity** check-box is checked. It is available for all gradient output image types.

5.7.2 Gradient multiplying with intensity

This method enables to multiply the gradient matrix with the intensity one, both with applied gamma correction. It affects the whole image, but the output images are usually more natural, because they are more similar to optical-microscope images. This method is used if the **HSL: H phase, S 1, L gradient x intensity** check box in the **Output image type:** group is checked. It is described in section 5.9.3.

5.8 Output image characteristics

5.8.1 Image file names

The output file names prefix is set in the **File names** edit. The program appends the image type and a three-digit counter starting from 000. The image file format is **bif**. It is always a 16 bit per pixel or per color component BIF. For more information about this format see [12]. The format enables to save additional information about the image in the header of the file. The information is saved as text and can therefore be easily displayed, e.g. by means of the **Lister** in the **Total Commander** or the **View/View BIF header** procedure in the **BIFview** software, which can be found on the enclosed CD. The **Gradient3D** software saves all information about the gradient computation and parameter setting in the BIF file header.

The layer image files are saved in a directory with the same name as the input files. If there have been existing files with same file names, they are overwritten. If this directory does not exist, it is automatically created.

5.8.2 Image resampling

The output images can be enlarged or lessened by setting the **Resample** value. Each image aspect will be multiplied by the set number. Images are enlarged by means of bicubic interpolation and lessened by means of supersampling. To make the computation faster, the program uses the **Resize** procedure in the **IIPP** dynamic library [10].

5.8.3 Image cut

Often, pixels on image edge are quite unreliable due to the numerical methods used for the complex amplitude reconstruction. They appear almost black or even worse, very bright in the gradient image. Therefore, it is useful to remove these pixels from the output image. Numbers determining how much the images will be cropped from each side are entered into the edits on the gray rectangle in the left program panel. These numbers refer to the original image size. If the image is resampled, these numbers are multiplied by the **Resample** value.

5.9 Output image types

5.9.1 Intensity, phase

The **Gradient3D** software enables to save the displayed intensity and phase images, with the current gamma correction applied and with some pixels possibly colored blue or red in places of low intensity (see section 5.7.1). These images are saved in the 24 bit per pixel **bmp** format in the output directory (see section 5.8.1) with file names **Intensity** or

Phase and the layer number. An image is saved after the small button with a floppy disk icon above the phase or intensity image is pressed.

5.9.2 Images without phase gradient

Even if gradient is not computed, the program enables to save some processed images. By pressing the **Save images HSL: H phase, S 1, L intensity** button, it is possible to obtain images which describe the input, i.e. the intensity and phase. The images can be described in the HSL color space as

$$\begin{pmatrix} H \\ S \\ L \end{pmatrix} (x, y, z) = \begin{pmatrix} p \\ 1 \\ \rho \end{pmatrix} (x, y, z), \quad (42)$$

which means that intensity is described with lightness and phase with hue. Saturation is set to the maximum so that phase is easy to distinguish.

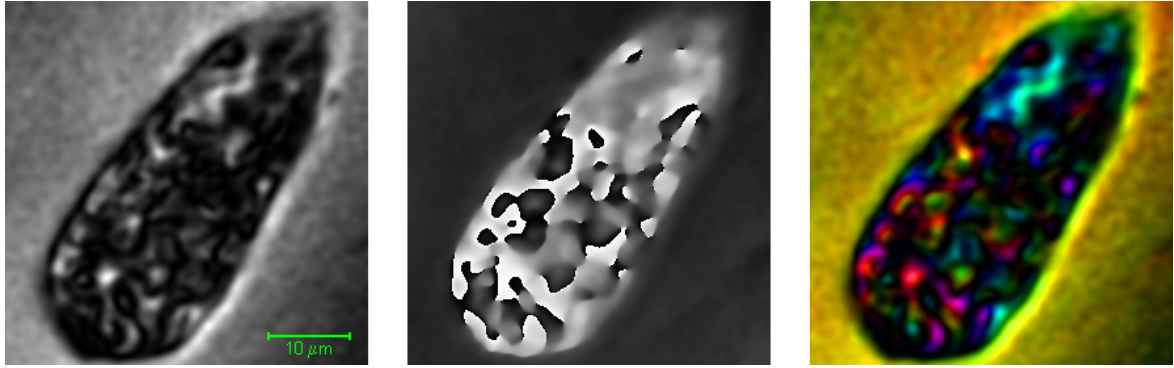


Figure 11: Crops of an intensity image, a phase image and an image with $(H, S, L)^T = (p, 1, \rho)^T$.

5.9.3 Images with phase gradient

After phase gradient is computed, the **Save layers** button is enabled. The image type is set in the **Output image type:** group. The following types are available.

5.9.3.1 Gradient only – BW images

This option saves gray-scale images of phase gradient with applied gamma correction.

5.9.3.2 HSL: H phase, S 1, L gradient

This option saves color images which can be described in the HSL color space as

$$\begin{pmatrix} H \\ S \\ L \end{pmatrix} (x, y, z) = \begin{pmatrix} p \\ 1 \\ \Gamma_{\gamma_g}(g) \end{pmatrix} (x, y, z), \quad (43)$$

where γ_g is the set gradient gamma value. Phase is described with hue, gradient with lightness, saturation is set to the maximum. This method does not create suitable images if there are places with low intensity, see fig. 12. In this case, the **HSL: H phase, S 1, L gradient x intensity** method should be used instead.

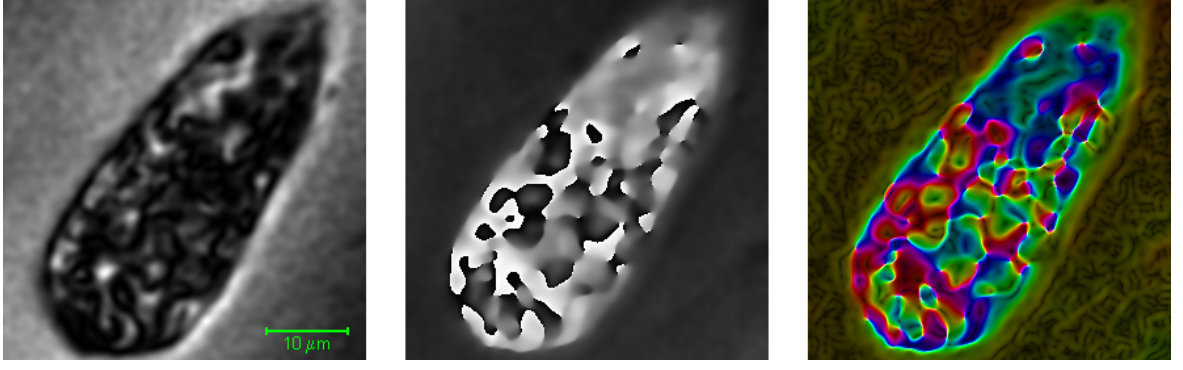


Figure 12: Crops of an intensity image, a phase image, and an image with $(H, S, L)^T = (p, 1, \Gamma_{2.5}(g))^T$, gradient computation method XY backward.

5.9.3.3 HSL: H phase, S 1, L gradient x intensity

This option saves color images which can be described in the HSL color space as

$$\begin{pmatrix} H \\ S \\ L \end{pmatrix} (x, y, z) = \begin{pmatrix} p \\ 1 \\ \Gamma_{\gamma_g}(g) \cdot \Gamma_{\gamma_\rho}(\rho) \end{pmatrix} (x, y, z), \quad (44)$$

where γ_g is the set gradient gamma and γ_ρ is the set intensity gamma. The images of both gamma corrections are mutually multiplied. Phase is described with hue and saturation is set to maximum again.

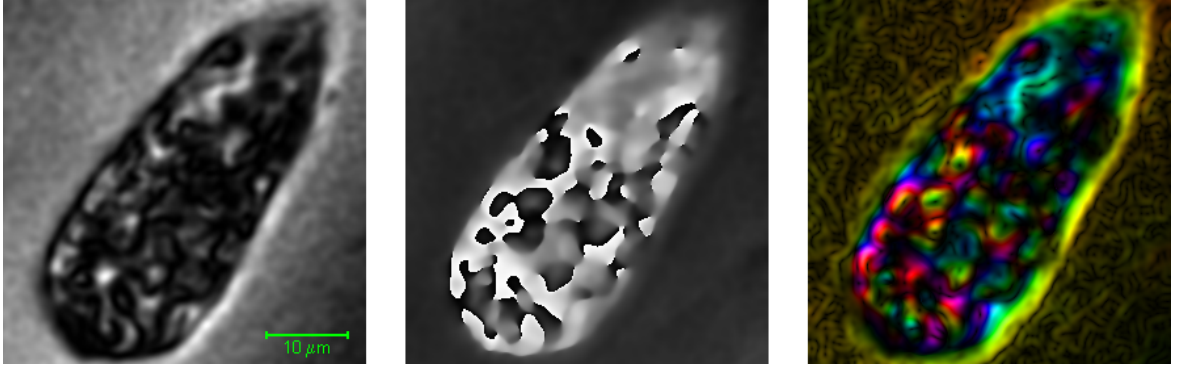


Figure 13: Crops of an intensity image, a phase image, and an image with $(H, S, L)^T = (p, 1, \Gamma_{1.5}(g) \cdot \Gamma_1(\rho))^T$, gradient computation method XY backward.

The disadvantage of this method is that it enhances the noise on the background of the observed object, because the intensity of the observed object is always lower than the background intensity.

The $(H, S, L)^T$ vector in this section does not correspond completely with the HSL definition in section 4.2 where H, L values were in $(0, 1)$ interval. In this section, they can be any non-negative values and will be transformed into this interval before the HSL to RGB conversion. The color scale for the hue component of the color images is shown in fig. 16.

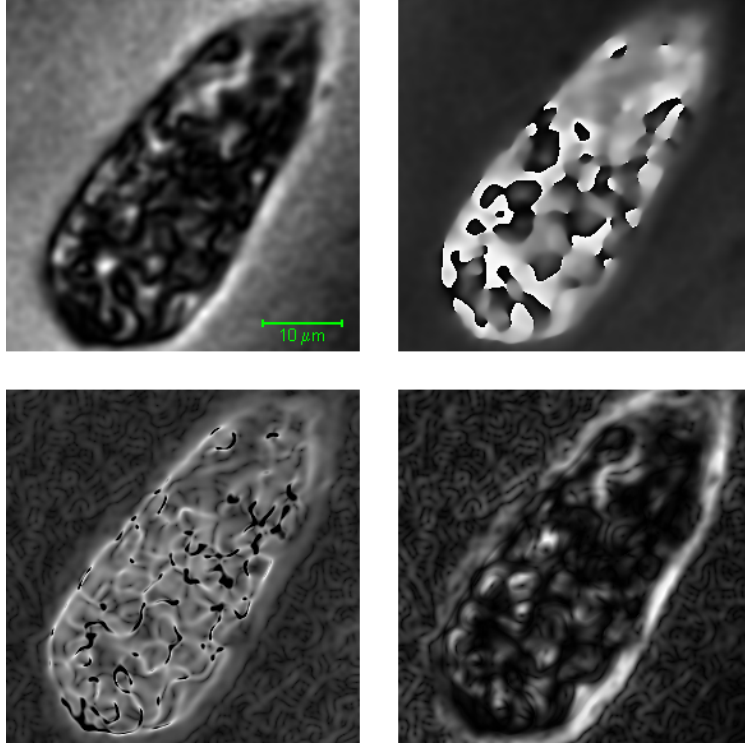


Figure 14: Crops of an intensity image, a phase image, and the lightness component of gradient XY images, the first one masked in dependence on intensity, $T = 0.0050M$, $P = 0.018M$, $(H, S, L)^T = (p, 1, \Gamma_{2.5}(g))^T$, the second one multiplied by intensity, $(H, S, L)^T = (p, 1, \Gamma_{1.5}(g) \cdot \Gamma_1(\rho))^T$, gradient computation method XY backward.

6 Conclusion

I have created the **Gradient3D** software which enables to

- compute phase gradient in two dimensions (in layers) without a necessity of phase unwrapping,
- compute phase gradient in three dimensions without a necessity of phase unwrapping,
- display intensity, phase and gradient graphically,
- create gray-scale images of phase gradient,
- create color images whose components in the HSL color space are combinations of intensity, phase and gradient,
- cut and resize output images, apply gamma-corrections on image color components,
- handle the images so that the false gradients in places with low intensity are removed.

The program has been tested on several hologram sets, some of them are presented in this work.

References

- [1] KOLMAN, P., et al. In vitro Dynamic Observations in a Low-Coherence Holographic Microscope. In *Czech-Polish-Slovak Conference on Wave and Quantum Aspects of Contemporary Optics*. October 11 – 15, 2006, Technical University Liberec, Czech Republic. pp. 66090M – 66096. ISBN 9780819467486. SPIE – The International Society for Optical Engineering.
- [2] CHMELÍK, R. Three dimensional scalar imaging in high-aperture low-coherence interference and holographic microscopes. *Journal of Modern Optics*, December 15, 2006, vol. 53, no. 18, pp. 2 673 – 2 689. ISSN 0950-0340.
- [3] ČÍŽEK, V., *Diskrétní Fourierova transformace a její použití*. 1st edition. Praha: SNTL – Nakladatelství technické literatury, n. p., 1981. 160 pp. Matematický seminář SNTL. ISBN 04-019-81.
- [4] BEZVODA, V., et. al. *Dvojměrná diskrétní Fourierova transformace a její použití – I.: Teorie a obecné užití*. 1st edition. Praha: Státní pedagogické nakladatelství, n. p., 1988. 181 pp. ISBN 17-135-88.
- [5] KOLMOGOROV, A. N., FOMIN, S. V. *Základy teorie funkcí a funkcionální analýzy*. translation Doležal, V., Tichý Z. 1st edition. Praha: SNTL – Nakladatelství technické literatury, n. p., 1975. 584 pp. Teoretická knižnice inženýra. ISBN 04-01575.
- [6] Dirac Delta Consultants Ltd. *Hanning Window* [online]. c2008, last revision January 31, 2008 [cit. March 13, 2008]. <[http://www.diracdelta.co.uk/science/source/h/a/hanning% 20window/source.html](http://www.diracdelta.co.uk/science/source/h/a/hanning%20window/source.html)>.
- [7] RÁB, M. *Komplexní čísla v elementární matematice*. 2nd revised printing, 1996. Vydavatelství Masarykovy univerzity, c1996. ISBN 80-210-1475-X.
- [8] GIANGRANDI, I. *Sensitivity of the human eye* [online]. c2004, last revision December 1, 2004. [cit. March 17, 2008], <<http://www.giangrandi.ch/optics/eye/eye.shtml>>.
- [9] *What Wavelength Goes With a Color?* [online]. c2007, last revision September 28, 2007 [cit. March 17, 2008]. <http://eosweb.larc.nasa.gov/EDDOCS/Wavelengths_for_Colors.html>.
- [10] *Intel Integrated Performance Primitives for Intel Architecture: Reference Manual*. Volume 2: Image and Video Processing. Intel. Document Number: A70805-021US. September 2007.
- [11] Wikipedia contributors. *HSL and HSV* [online]. c2008, last revision February 22, 2008 [cit. February 28, 2008]. <http://en.wikipedia.org/wiki/HSV_color_space>.
- [12] DRUCKMÜLLER M., ŠTARHA P. *Adaptive Contrast Control Image Structure and Object Analyzer version 6.1 User's guide*, SOFO 2008.

A Used symbols

b^*	complex conjugate of b
w, h, d	image width, height, depth, see def. 3.12, page 11
A_1	set $\{0, 1, \dots, w - 1\}$, see def. 3.12, page 11
A_2	set $\{0, 1, \dots, h - 1\}$, see def. 3.12, page 11
A_3	set $\{0, 1, \dots, d - 1\}$, see def. 3.12, page 11
\mathbb{R}_0^+	$\langle 0, \infty \rangle$
\mathbb{R}^+	$(0, \infty)$
$[a]$	integral part of a real number a , see def. 3.15, page 12
\tilde{o}'	object complex amplitude, see formula 17, page 10
$\tilde{\rho}$	intensity, see def. 3.10, page 11
\tilde{p}	phase, see def. 3.10, page 11
o'	discrete approximation of complex amplitude, also complex amplitude only, see def. 3.12, page 11
ρ	discrete approximation of intensity, also intensity only, see def. 3.13, page 11
p	discrete approximation of phase, also phase only, see def. 3.13, page 11
\hat{p}	aligned phase, see sections 3.2.2, 5.3
\tilde{g}	phase gradient norm, see formulae 28, page 12
g	discrete approximation of phase gradient norm, also gradient only, see sections 3.2.1 and 3.2.2
ω	z -axis difference weight, see sections 3.2.2, 5.3
\oplus	addition modulo 2π , see def. 3.16, page 12
\boxplus	addition modulo 1, see def. 4.17, page 15
$[R, G, B]$	a color represented with the red, green, blue triplet, see section 4.1, page 14
$[H, S, L]$	a color represented with the hue, saturation, lightness triplet, see section 4.2, page 15
$c(\rho)$	gradient-masking function in dependence on intensity, see formula 41, page 22
M	the maximal intensity value $\max_{[x,y,z] \in A_1 \times A_2 \times A_3} \rho(x, y, z)$
T	an intensity value under which $c(\rho)=0$, see section 5.7.1, page 22
P	an intensity value starting with $c(\rho)=1$, see section 5.7.1, page 22
Γ_γ	gamma correction function with the γ value, see def. 5.18, page 21
γ_g	gamma value of the gamma correction applied on the gradient matrix, see def. 5.18, page 21
γ_ρ	gamma value of the gamma correction applied on the intensity matrix, see def. 5.18, page 21

B Enclosed CD

The enclosed CD contains:

- this bachelor's thesis in PDF format
- asc to txx converter (Konvertor\Konvertor.exe)

- **Gradient3D** software (Gradient3D\Gradient3D.exe)
- testing data for the **Gradient3D** software (Data directory)
- a viewer of BIF image files (BIFview\BIFview.exe)

C Figures

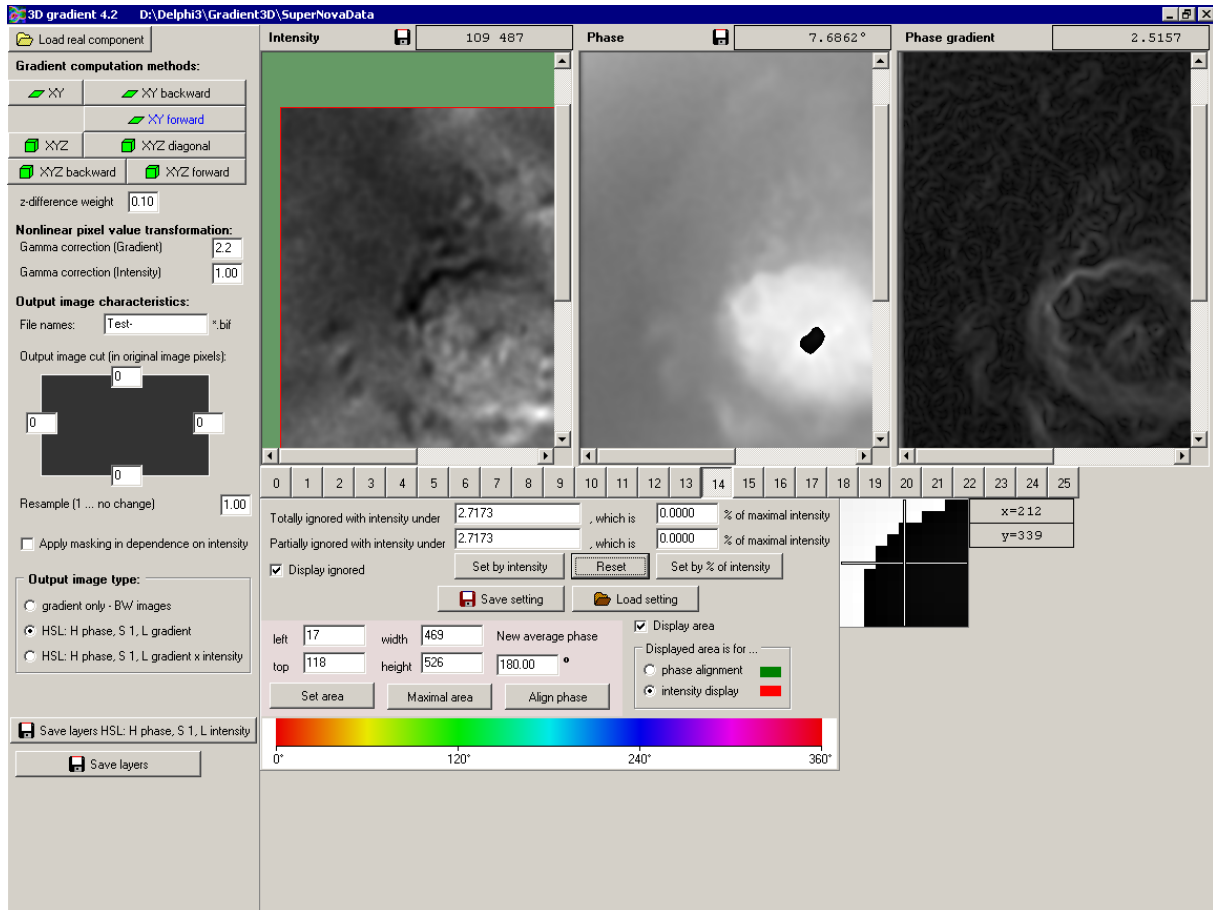


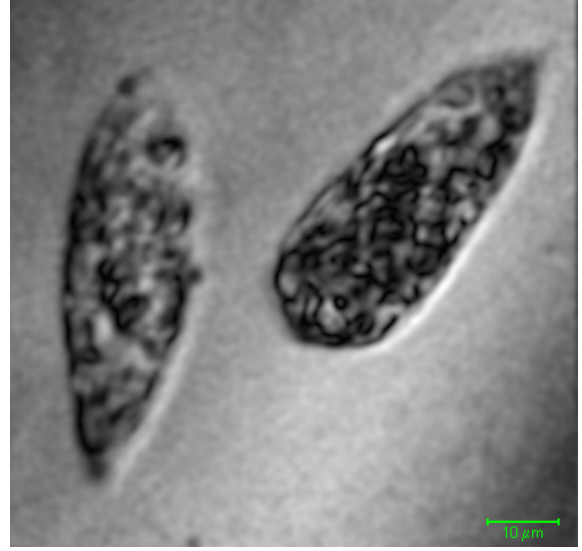
Figure 15: **Gradient3D** software control panel.



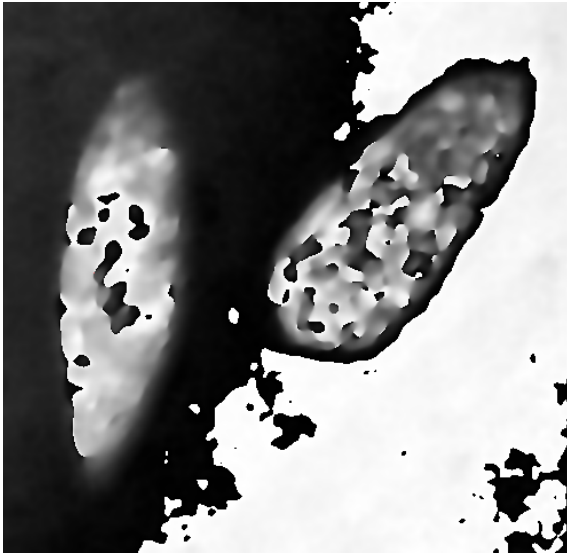
Figure 16: Hue scale used in the HSL color space.



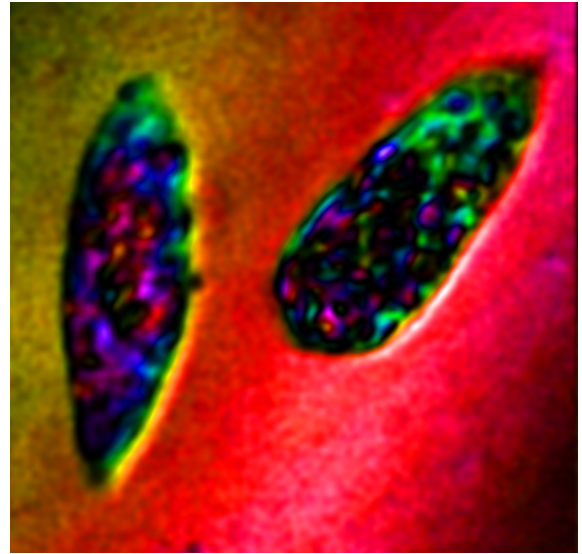
(a) Original hologram.



(b) Intensity, $\gamma_\rho = 1.8$.

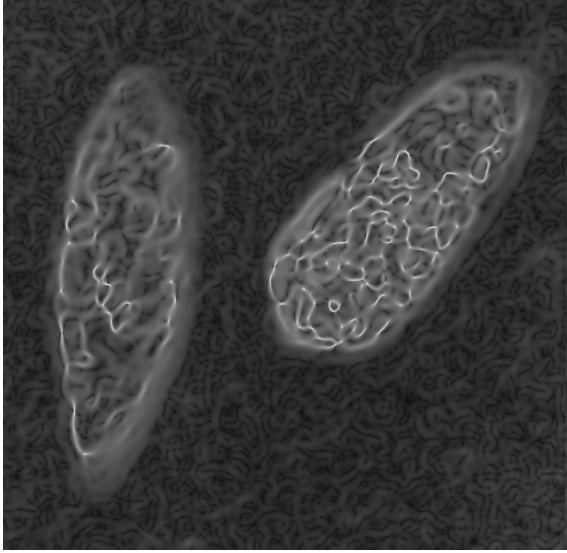


(c) Phase.

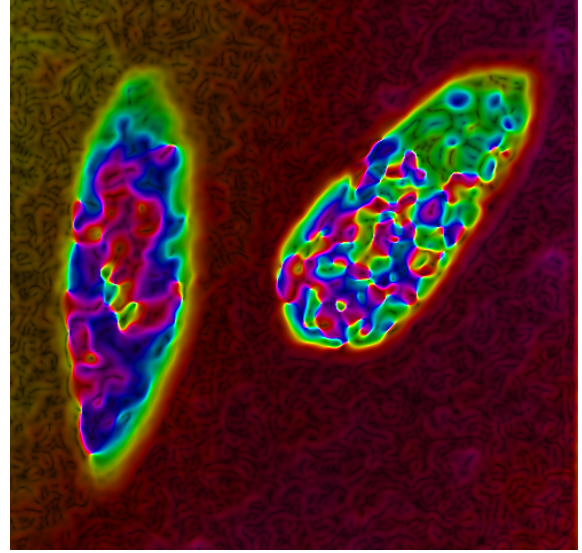


(d) $(H, S, L)^T = (p, 1, \rho)^T$.

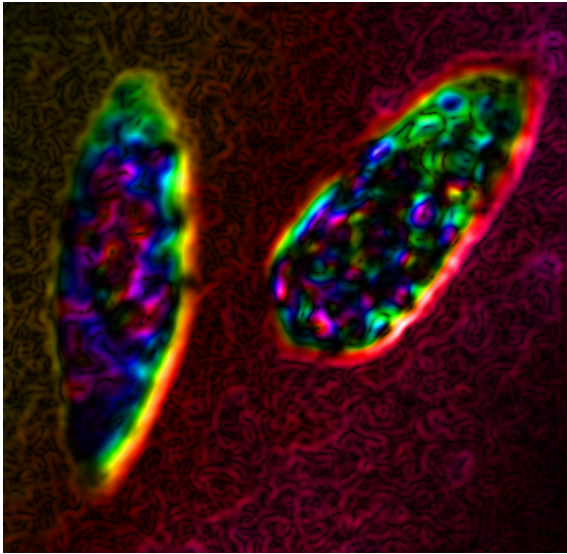
Figure 17: Crops of images of *Tetrahymena thermophila*, November 15, 2007, microscope DHM2, achromatic objectives Lambda 20 \times /NA=0.4 – input image representation.



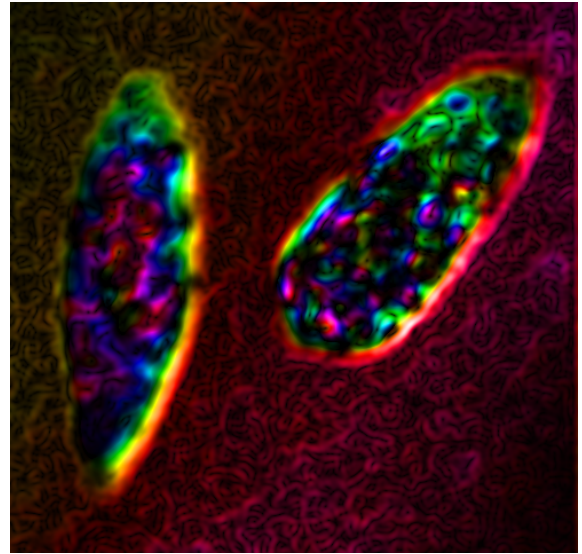
(a) Gradient, $\gamma_g = 3.5$.



(b) $(H, S, L)^T = (p, 1, \Gamma_3(g))^T$.



(c) $(H, S, L)^T = (p, 1, \Gamma_{1.5}(g) \cdot \Gamma_1(\rho))^T$.



(d) $(H, S, L)^T = (p, 1, \Gamma_{1.5}(c(\rho) \cdot g) \cdot \Gamma_1(\rho))^T$,
 $T = 0.00334M$, $P = 0.00656M$.

Figure 18: Crops of images of *Tetrahymena thermophila*, November 15, 2007, microscope DHM2, achromatic objectives Lambda 20 \times /NA=0.4 – processed image representation, gradient computation method XY backward.