

Article

ZernikeViewer: An Open-Source Framework for Fast Simulation and Real-Time Reconstruction of Phase, Fringe, and PSF Maps

Ilya Galaktionov ^{1,2} 

¹ Quantum Center, Moscow Technical University of Communications and Informatics, Aviamotornaya Str., Bld. 8a, Moscow 111024, Russia; ilya.galaktionoff@gmail.com

² Physics Department, Moscow Polytechnic University, Bolshaya Semenovskaya Str., Bld. 38, Moscow 107023, Russia

Abstract

Zernike polynomials constitute an essential mathematical basis for representing functions defined over the unit disk. They are widely used in a diverse range of scientific and engineering disciplines, including adaptive optics for characterizing atmospheric distortions, ophthalmology for quantifying ocular aberrations, microscopy for instrument characterization and aberration correction, and optical metrology for surface profiling. This paper introduces ZernikeViewer, a software framework developed for the rapid calculation and visualization of fringe, phase, and point spread function (PSF) maps from Zernike coefficients. The framework leverages CPU multicore and multithreading capabilities through the .NET Task Parallel Library (TPL), augmented by codebase optimizations and the preloading of precomputed Zernike polynomial matrices. These optimizations reduce computation time by a factor of 7 to 10 compared to a conventional approach; for instance, from 1 ms to 0.1 ms for a radial order of $n = 10$ and from 700 ms to 80 ms for $n = 100$. Numerical error analysis confirms the accuracy of the computation, with an average root-mean-square (RMS) error of 0.11 ms observed in the timing measurements. Furthermore, it is demonstrated that implementing Jacobi recursion relations could potentially reduce the numerical calculation error by up to 5 orders of magnitude.

Keywords: Zernike polynomials; CPU computing; parallel computing; wavefront reconstruction; phase map; spatial light modulator



Academic Editor: Teen-Hang Meen

Received: 18 January 2026

Revised: 11 February 2026

Accepted: 24 February 2026

Published: 26 February 2026

Copyright: © 2026 by the author.

Published by MDPI on behalf of the International Institute of Knowledge Innovation and Invention. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

1.1. Background

In computational science and engineering, orthogonal polynomials constitute a foundational tool for enabling efficient numerical simulations and data analysis. Among these, Zernike polynomials have attracted considerable interest due to their unique mathematical properties and broad applicability [1,2]. Their widespread adoption came from several key properties: foremost, their orthogonality over the unit circle facilitates the independent decomposition and analysis of complex wavefields or surface profiles. Furthermore, their completeness allows for the precise description of arbitrary wavefront shapes. This combination of mathematical elegance and physical relevance led to their widespread use across a diverse range of scientific and engineering disciplines, including adaptive optics for characterizing atmospheric distortion [3], ophthalmology for quantifying ocular aberrations in refractive surgery [4], microscopy for instrument characterization and aberration correction [5], astronomy for telescope mirror design [6], and general optical metrology for

surface profiling and quality assessment [7]. First introduced by Frits Zernike [8,9], Zernike polynomials form a complete orthogonal set over the unit disc. Their extensive adoption in optical and ophthalmological contexts, both clinical and academic, is largely attributable to their direct correspondence with classical Seidel aberrations [8], making them a natural basis for wavefront reconstruction and analysis. Unlike Bessel functions, which can exhibit singularities at the origin, Zernike polynomials are finite and smooth across the entire disc, including the center ($r = 0$), thereby ensuring continuous differentiability along the optical axis.

The utility of Zernike polynomials is particularly pronounced in several key areas. Crucially, the low-order Zernike modes directly correspond to classical optical aberrations such as tilt, defocus, and astigmatism, providing an intuitive and physically meaningful representation of optical system performance. They are extensively employed to characterize wavefronts and quantify aberrations in lenses, mirrors, and the human eye [10–20]. This facilitates the design and optimization of high-precision optical instruments. Their benefits in optics translate directly to astrophysical applications, where they are used to model surface irregularities in telescope mirrors, simulate optical system performance, and mitigate the impact of aberrations on image quality [21–26]. In simulations of magnetically confined plasmas, Zernike polynomials serve as a spectral basis to represent the complex toroidal cross-sections of stellarators and tokamaks, thereby facilitating the numerical solution of the underlying partial differential equations [27–29]. The orthogonality of Zernike polynomials makes them well-suited as basis functions in spectral methods for solving partial differential equations defined on disc-like domains [30]. The applicability of Zernike polynomials extends far beyond the few examples cited here, spanning a wide spectrum of scientific and engineering fields [10].

While Zernike polynomials form a mathematically complete orthogonal basis on the unit disk, this ideal property is often compromised in practical applications. Experimental wavefront data are typically acquired at discrete sampling points rather than as a continuous function. Furthermore, real-world optical apertures frequently deviate from a circular geometry, instead taking the form of polygons, ellipses, annuli, or sectors due to anatomical constraints or specific instrument designs. On such non-standard domains, Zernike polynomials lose their orthogonality and may no longer provide an efficient representation of classical optical aberrations.

Under these conditions, alternative functional bases—not necessarily orthogonal—can become competitive [31–34]. When preserving orthogonality is essential, several strategies have been pursued. These include applying the Gram–Schmidt orthogonalization process to generate a custom orthogonal set for the specific domain [35–37] and employing diffeomorphisms to map basis functions from the unit disk onto the non-standard domain [38,39]. For rectangular domains, an analogue to Zernike polynomials (known in the literature as Furter polynomials) has been proposed [40,41]. Beyond the selection of an appropriate basis, the accuracy of wavefront reconstruction is critically dependent on the fitting procedure used to approximate a continuous function from a finite set of discrete measurements. This process can yield either an interpolating function or a discrete best-fit approximation. An important factor influencing the stability and accuracy of the reconstruction is the spatial distribution of the sampling nodes [42–46]. A notable contribution is the Optimal Concentric Sampling (OCS) scheme [44], which utilizes a set of concentric circles with optimally determined radii. This approach guarantees significantly enhanced numerical stability. This problem is fully analyzed in the research conducted by Diaz-Elbal et al. [47].

In our research, we will concentrate on the conventional circular mask geometry that is most widely used in a vast majority of applications.

The computational evaluation of Zernike polynomials has a rich history, marked by continuous refinement to improve efficiency and numerical stability. Early methods relied on direct evaluation, which, while conceptually straightforward, proved computationally expensive and prone to instability for higher-order polynomials.

Subsequently, more sophisticated algorithms based on recurrence relations were developed [48–54]. These methods effectively mitigated instability issues for high radial and azimuthal orders and provided faster evaluation schemes. Real-time processing with Zernike polynomials becomes a must-have for the super-resolution structured illumination microscopy [55] and real-time holographic cameras [56]. However, many recurrence relations require supplementary formulas for edge cases, introducing minor coding complexity. This is particularly detrimental to performance on Graphics Processing Units (GPUs), which are inefficient at handling conditional execution branches.

The advent of parallel computing has motivated further algorithmic innovations. For instance, Qin et al. [57] leveraged the relationship between Zernike and Jacobi polynomials, combined with the Jacobi recursion relation, to decouple the radial polynomial calculations and vectorize operations. This approach demonstrated superior performance compared to earlier recursive methods, though its implementation was confined to Central Processing Units (CPUs). Alternative strategies include reformulating Zernike polynomials in terms of the discrete Fourier transform (DFT) [58], an approach later adapted for GPU computation by Al-Rawi [59]. This method can be used to generate accurate moments up to high orders and can compute Zernike moments of 4K resolution images in real-time. The proposed computation of Zernike moments using fast Fourier transform is more accurate than the q-recursive method. Furthermore, the method is resilient and numerically stable up to extremely high orders. Numerical accuracies of Zernike moments computed with the proposed FFT approach have been analyzed by the authors using the orthogonality property, and the results showed that they beat other methods in numerical stability. The proposed method is simple and fast and can make use of the huge GPU-FFT libraries that are available in several programming frameworks. Zhang et al. [60] developed an algorithm utilizing a balanced binary scheme to reduce memory usage and increase computational speed.

Despite these significant advances in enhancing the accuracy, speed, and stability of Zernike polynomial computations, a notable gap remains in the availability of comprehensive, open-source software packages that implement these state-of-the-art methods.

1.2. Motivation

The extensive application of Zernike polynomials in the diverse scientific fields, namely adaptive optics for atmospheric and quantum applications [61,62], ophthalmology for quantifying ocular aberrations in refractive surgery, microscopy for instrument characterization and aberration correction, astronomy for telescope mirror design, and general optical metrology for surface profiling and quality assessment, highlights a critical demand for computationally robust and efficient algorithms. Such tools are essential to handle the demanding numerical computations, including high-order evaluations and large-scale spectral simulations, that these applications entail.

Several publicly available web-based resources, often termed “Zernike calculators,” have been developed. For instance, Elmacioglu et al. [63] described a Python-based calculator named Zernipax, while Kim et al. [64] presented a calculator utilizing a deep learning approach. Furthermore, various online calculators offer differing levels of detail in their input parameters and output results [65–68]. A common characteristic of most tools is their reliance on web frameworks and execution via interpreted, rather than compiled, source code. While this approach offers convenience, it introduces two significant limitations: first, inherently low computational speed, and second, a mandatory requirement for persistent

internet access. Although internet connectivity is widespread, some research laboratories often operate without access to the internet. Moreover, a conventional desktop application with a user-friendly graphical interface (GUI) and robust data export functionality is generally more practical and convenient for both novice and advanced users.

In this paper, we introduce ZernikeViewer v3.0—a software framework designed for the rapid, accurate calculation, visualization, and analysis of wavefronts. The framework directly utilizes Zernike polynomial coefficients to generate fringe patterns, phase maps, and Point Spread Function (PSF) maps.

The principal innovative features of ZernikeViewer are as follows:

- **High-Performance Computation.** The framework leverages CPU multicore and multithreading capabilities through the .NET Task Parallel Library (TPL) [69], complemented by low-level codebase optimizations to maximize computational efficiency.
- **Precomputation and Caching.** A preloading feature for a matrix of precalculated Zernike polynomial values further accelerates computation speed for repetitive analyses.
- **Real-Time Wavefront Manipulation.** The capability to reproduce and manipulate preloaded Zernike coefficients in real time enables precise analysis and post-processing of experimentally recorded wavefronts. This also provides a valuable tool for researchers requiring controllable, artificially generated turbulence in laboratory settings. Further details are provided in subsequent sections.
- **Spatial Light Modulator (SLM) Integration.** The tool can construct specialized phase maps with a more than 2π range (SLM-phase map) and project them onto a secondary display, making it a convenient and universal interface for applications involving Spatial Light Modulators [70].
- **A Conventional Desktop Application Design.** The tool has a user-friendly graphical interface (GUI) and robust data export functionality that is generally more practical and convenient for both novice and advanced users. An ability to operate in offline mode makes this tool even more robust.

The principal distinctions of the developed ZernikeViewer framework extend beyond the accelerated computation of Zernike polynomials. The system achieves significant performance gains across the entire visualization pipeline, including the calculation of phase, fringe, SLM-phase, and PSF maps. Furthermore, it enables real-time reproduction of map sequences and provides direct functionality for projecting the SLM-phase onto a secondary display, facilitating integration with optical hardware.

As outlined earlier, the software tool was developed in the .NET framework with a deliberate focus on leveraging the Task Parallel Library (TPL) to address the computational bottleneck inherent in high-order Zernike polynomial calculations. The core algorithm was structured into fine-grained, independent tasks—primarily the concurrent evaluation of individual polynomials across a grid of spatial coordinates. This parallelization strategy directly targets the $O(Nm^2)$ scaling of traditional sequential methods, where N is the number of modes and m the grid resolution. The choice of TPL, with its efficient work-stealing scheduler and low-overhead task management, ensures that the theoretical parallelism is effectively mapped onto modern multi-core processors. This architectural decision is quantified by the observed performance metrics: a near-linear speedup with core count and a linear memory complexity of $O(N)$, achieved by storing double-precision polynomial values in optimized concurrent structures. These metrics are not merely benchmarks but are critical for enabling the application scenarios discussed below, as they transform previously intractable calculations into interactive or batch-processable tasks.

The performance characteristics of the tool are designed to bridge the gap between theoretical analysis and practical optical engineering. In applications such as wavefront sensing and adaptive optics correction, system latency is paramount; the achieved compu-

tational speed allows for the real-time reconstruction of high-order wavefront aberrations from sensor data. Similarly, in optical design and tolerance analysis, where thousands of Monte Carlo simulations are required to assess system performance against manufacturing errors, the tool's efficiency enables comprehensive statistical studies that were previously too time-consuming. Furthermore, the accurate and rapid generation of Zernike surfaces is fundamental in computational microscopy for point-spread function engineering and in ophthalmology for the precise modeling of corneal aberrations. Therefore, the implementation choices—parallel computation for speed and double-precision arithmetic for numerical fidelity—are directly motivated by and validated against these demanding scientific and industrial use cases, ensuring the tool is not just computationally efficient but also scientifically relevant and practically viable. The developed tool's source code is available in a GitHub repository.

The remainder of this paper is structured as follows. Section 2 reviews the mathematical foundation of Zernike polynomials. Section 3 describes the experimental setup used to acquire the raw data (comprising an extensive set of phase maps) and elaborates on the framework's methodology, the core mathematical apparatus, the user interface, and the specific software libraries employed. Section 4 presents a comprehensive performance benchmark and error analysis of the developed framework.

2. Theory: Zernike Circle Polynomials

Zernike polynomials might be defined in the following way (Equation (1)):

$$Z_n^m(\rho, \theta) = \begin{cases} R_n^m(\rho) \cdot \cos(m\theta) & \text{for } m \geq 0 \\ R_n^{|m|}(\rho) \cdot \sin(|m|\theta) & \text{for } m < 0 \end{cases} \quad (1)$$

where $\rho \in [0, 1]$, $\theta \in [0, 2\pi]$, R_n^m is the radial part of Zernike polynomial that can be defined as follows (Equation (2)):

$$R_n^{|m|}(\rho) = \sum_{s=0}^{(n-|m|)/2} \frac{(-1)^s (n-s)!}{s! \cdot \left(\frac{n+|m|}{2} - s\right)! \cdot \left(\frac{n-|m|}{2} - s\right)!} \rho^{n-2s}. \quad (2)$$

The radial part can be simplified to the following form (Equation (3)):

$$R_n^{|m|}(\rho) = \sum_{s=0}^{(n-|m|)/2} (-1)^s \cdot \binom{n-s}{s} \cdot \binom{n-2s}{\frac{n-|m|}{2} - s} \cdot \rho^{n-2s}. \quad (3)$$

The radial order n and the azimuthal order m are integers that index the Zernike polynomials and are constrained by specific mathematical relations. These indices must satisfy the conditions $n \geq 0$, $|m| \leq n$, and the parity condition that $n - |m|$ is an even number. This parity condition directly dictates the permissible sequence of azimuthal orders for a given radial order n , which is given by $n \in \{(-n, (-n+2), \dots, (n-2), n)\}$.

A notable property arising from these definitions is that the binomial coefficients appearing in previous equation are always integers. Consequently, the expansion coefficients of the Zernike polynomials themselves are also integers.

The value of a Zernike polynomial at the center of the unit disc $\rho = 0$ can be derived analytically from its radial component. At this singular point, all terms in the polynomial expansion containing a power of ρ vanish. The only non-vanishing term corresponds to the

instance where the exponent $n - 2s$ is zero. This condition is met when $s = n/2$. Substituting this value results in Equation (4):

$$R_n^{|m|}(\rho = 0) = \frac{(-1)^{n/2}}{\left(-\frac{m}{2}\right)! \cdot \left(\frac{m}{2}\right)!}. \quad (4)$$

The factorial terms within the polynomial definition necessitate that their arguments be non-negative integers. This constraint, when applied at the coordinate origin $\rho = 0$, restricts the permissible value of the azimuthal order to $m = 0$. Consequently, the angular dependence on θ is eliminated at the disc center. This behavior is not merely a mathematical artifact but is a fundamental requirement for the Fourier expansion of analytic functions on a polar plane, as established in [71]. The intrinsic properties of Zernike polynomials, including this specific behavior at the origin, ensure they automatically satisfy these analyticity conditions. This makes them a rigorously sound and particularly robust choice for a basis set in numerical simulations conducted in polar coordinate systems.

The mathematical expressions defined in this section are entirely adequate for determining the values of Zernike polynomials across the unit disk. From a computational perspective, however, a straightforward translation of these expressions into code results in nested loop architectures, which are notoriously inefficient and form the principal bottleneck for accelerating computations. The remainder of this paper addresses this specific challenge by introducing a novel implementation strategy designed to enhance computational frequency.

3. Materials and Methods

3.1. Experimental Setup

An optical scheme for laser beam wavefront measurements [72] that might be used in order to obtain a set of wavefronts is illustrated in Figure 1.

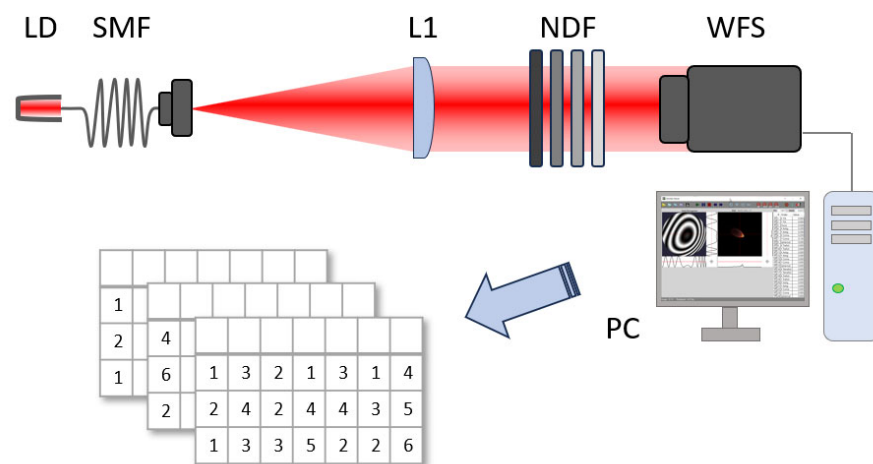


Figure 1. Optical configuration of a scheme for wavefront measurements. The result of measurements is a set of Zernike coefficient values.

A laser diode (LD) at a wavelength $\lambda = 0.65 \mu\text{m}$, coupled to a single-mode fiber (SMF) of numerical aperture $\text{NA} = 0.14$ and mode field diameter $\text{MFD} = 10.4 \mu\text{m}$ is collimated with an achromatic lens (L1) with focal length $f = 400 \text{ mm}$ into a beam with a diameter of 45 mm . The beam is propagated along the optical bench for a distance of about 0.5 m , passes through a set of neutral density filters and falls on the aperture of the Shack–Hartmann wavefront sensor [73,74]. Its output is linked to the control personal computer (PC) using

a USB 3.0 connection. The signal received is processed with our proprietary WaveTuner software v1.1 that calculates the wavefront and returns a set of Zernike coefficients values.

In principle, any dedicated wavefront processing software can be employed for data acquisition, though researchers commonly utilize the proprietary software bundled with their wavefront sensor hardware. Such software typically enables real-time measurement and the storage of wavefront data batches, which can be formatted as a phase matrix or a set of coefficients from a polynomial expansion (e.g., Seidel [75], Zernike [76,77], B-Spline [78]). This data can subsequently be imported into the ZernikeViewer framework for reproduction and advanced analysis. Representative use cases for this workflow include:

1. Characterization of distorted wavefronts. Measuring wavefronts aberrated by natural or artificially generated turbulence [79] is critical for testing turbulence generation methods or wavefront correction algorithms. In this context, ZernikeViewer functions as an analysis tool: a recorded set of wavefronts is imported, the framework performs decomposition and real-time reproduction, simultaneously calculating and displaying the corresponding Zernike coefficients, phase map, fringe pattern, and PSF map for precise quantitative assessment. Furthermore, this generated sequence of phase screens can be inverted and projected onto a secondary monitor connected to an SLM. In a controlled laboratory setting, this allows for the theoretical compensation of wavefront distortions within an optical path, providing a valuable method for system testing, simulation, and dynamic correction.
2. Post-processing of historical data. The framework facilitates the detailed reconstruction and characterization of wavefronts from previously recorded measurements, enabling retrospective analysis.

The following section describes the fundamental mathematical principles underlying the ZernikeViewer framework.

3.2. Methodology

The phase map (the wavefront surface) is calculated using Equation (5):

$$W(x, y) = \sum_{i=0}^N a_i \cdot Z_i(x, y) = \sum_{n=0}^k \sum_{m=0}^n a_{nm} \cdot Z_n^l(\rho, \theta), \quad (5)$$

where a_{nm} is the Zernike coefficient that represents the aberration value, N is a number of the Zernike polynomials used, $l = n - 2m$.

This equation can be written in the matrix form as follows (Equation (6)):

$$\begin{bmatrix} W(x_1, y_1) \\ W(x_2, y_2) \\ \vdots \\ W(x_N, y_N) \end{bmatrix} = \begin{bmatrix} Z_1(x_1, y_1) & Z_2(x_1, y_1) & \dots & Z_N(x_1, y_1) \\ Z_1(x_2, y_2) & Z_2(x_2, y_2) & \dots & Z_N(x_2, y_2) \\ \vdots & \vdots & \ddots & \vdots \\ Z_1(x_N, y_N) & Z_2(x_N, y_N) & \dots & Z_M(x_N, y_N) \end{bmatrix} \cdot \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_M \end{bmatrix}. \quad (6)$$

In this equation, $W(x, y)$ represents the matrix of phase values, $Z_i(x_N, y_N)$ is the Zernike i -th polynomial value at the point (x_N, y_N) , and a_M is the Zernike coefficient value. In order to calculate the fringe map out of the phase map, the following equation is used (Equation (7)):

$$Fringe(x, y) = \frac{1}{2} \cdot (1 + \cos(W(x, y) \times k)), \quad (7)$$

where $Fringe(x, y)$ is a fringe matrix, $W(x, y)$ is a phase matrix, and k is a wave number.

When using SLM to reproduce the phase maps with amplitudes more than 2π , it is necessary to use a special equation for recalculation of phase map values (Equation (8)):

$$W_{SLM}(x,y) = \text{Remainder}\left(\frac{W(x,y)}{\lambda}\right) \cdot 255, \quad (8)$$

where $W_{SLM}(x,y)$ is a phase matrix for spatial light modulator, *Remainder* is an operator that takes the remainder of the operation, $W(x,y)$ is a phase matrix, and λ is a wavelength used.

Finally, in order to estimate what impact the aberrations have on the far field focal spot, the PSF can be used. Actually, it demonstrates the intensity distribution of the focal spot in the far field and can be calculated as follows (Equation (9)):

$$PSF(u,v) = \left| \text{Fourier}\left\{A(x,y) \cdot e^{i \cdot W(x,y)}\right\} \right|^2, \quad (9)$$

where $PSF(u,v)$ is intensity matrix in the image plane coordinates (u,v) , *Fourier* is a two-dimensional Fourier transform operator, $A(x,y)$ is a matrix of intensity values (might be a unity matrix in the simplest case), and $W(x,y)$ is matrix of phase values calculated at the previous step.

An example of all 4 maps described above and calculated by the developed tool is presented in Figure 2, particularly the phase map, the fringes map, the calculated PSF map and the special phase map that is used to project to the SLM. The maps were generated in ZernikeViewer for the following Zernike coefficients: Defocus—1 μm , Astigmatism Y—0.3 μm , Coma X—0.5 μm .

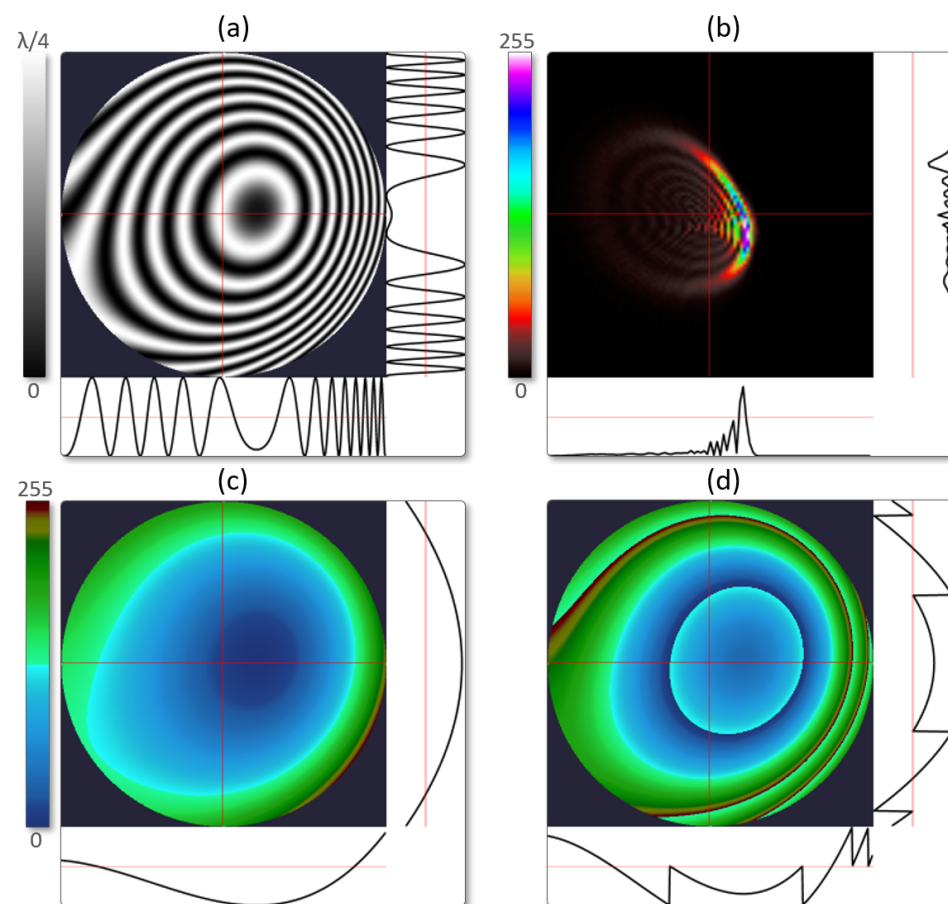


Figure 2. (a) Fringes map, (b) calculated PSF map, (c) phase map, (d) phase map for a spatial light modulator.

3.3. Process Flowchart

The overall workflow of the developed framework is presented in the flowchart in Figure 3.

Zernike polynomial coefficient sets are stored and exported in Comma-Separated Value (CSV) format. While CSV is not an optimized format for large-scale data storage due to its verbosity and lack of inherent data typing, it was selected for its universal interoperability. This format does not require specialized software for access and facilitates straightforward manual data inspection and manipulation. The framework supports exporting results in three primary formats: CSV files containing the raw numerical data (e.g., coefficient values, phase matrix), TXT files for Matlab (R2023b) integration, and Portable Network Graphics (PNG) files for visual representations, including phase, fringe, and PSF maps, as well as rendered tables of Zernike coefficients.

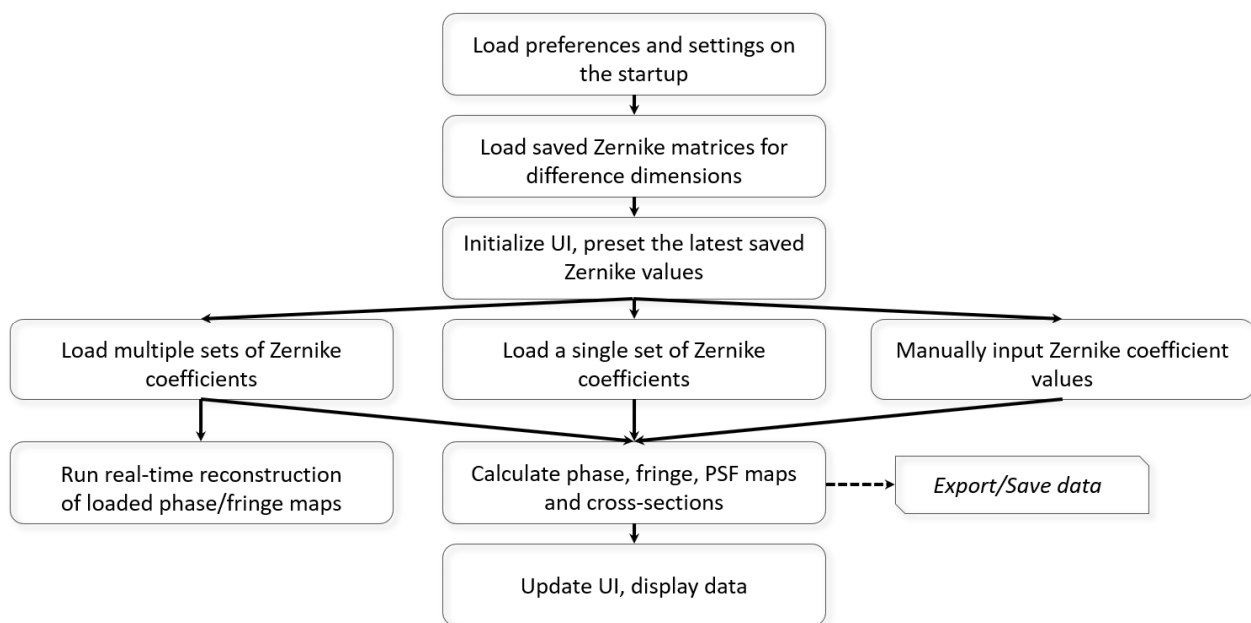


Figure 3. Flowchart of a framework workflow.

3.4. ZernikeViewer Framework UI and Scripts

The graphical user interface (GUI) of the developed tool is presented in Figure 4.

The GUI is logically partitioned into three main functional areas.

Toolbar. This component provides centralized control via buttons for core operations, including loading/saving Zernike coefficient sets, initiating/pausing/terminating the real-time reconstruction process, triggering recalculation, capturing screenshots of various UI elements, and accessing application settings.

Visualization Canvas (Main Area). This section is dedicated to the primary graphical output, displaying the calculated phase or fringe map alongside the corresponding PSF map. Both visualizations are supplemented with horizontal and vertical cross-sectional profiles for quantitative analysis.

Control and Monitoring Panel (Right/Bottom Area). The right-side panel hosts a table of the active Zernike coefficients, which are presented within editable fields to allow for interactive wavefront manipulation. The bottom-left area contains a running chart that visualizes a sequence of preloaded Zernike coefficient sets. User interaction, specifically a mouse click on a data point within this chart, automatically triggers the calculation and display of the phase and PSF maps corresponding to the selected coefficient set.

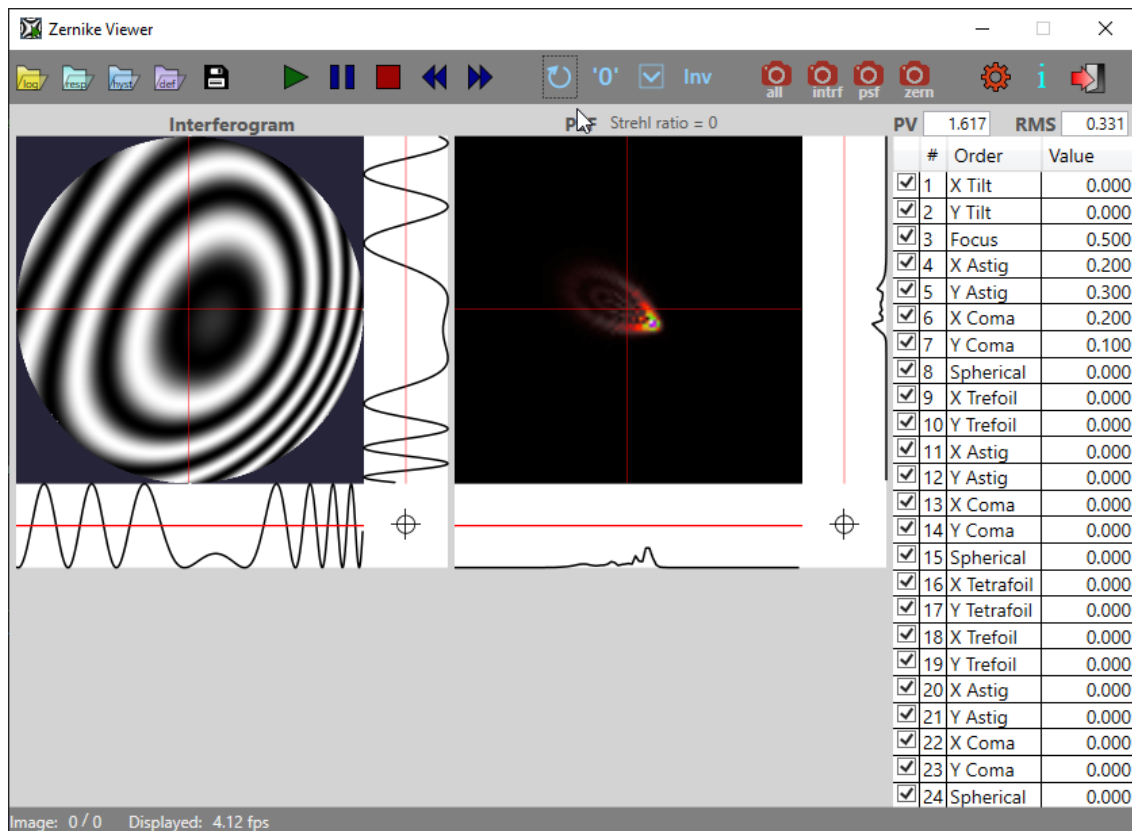


Figure 4. GUI of the ZernikeViewer.

The foundational element of the application's computational engine is the phase calculation algorithm. This algorithm is responsible for generating the phase matrix from the Zernike coefficients. To address performance bottlenecks, the implementation leverages the .NET TPL to parallelize computationally intensive loops.

The choice of the .NET platform can be justified as follows: (1) For CPU-bound numerical computation, .NET typically delivers significantly higher performance than many other frameworks and languages. (2) It provides comprehensive, integrated desktop development tooling. There is no need to select a separate GUI framework or manage the integration of front-end and back-end components, as the platform offers a complete, ready-to-use solution. (3) .NET features robust, built-in support for multithreading, parallel processing, and asynchronous operations, which is essential for efficient scientific computing. (4) The platform is under active and rapid development, supported by a large and enthusiastic community dedicated to advancing its capabilities and maximizing its performance. A key design goal was to avoid dependency on a dedicated Graphics Processing Unit (GPU), as not all laboratory workstations include discrete GPU hardware. The objective was to create a universal tool that is both computationally robust and highly performant across diverse hardware configurations.

As detailed in Algorithm 1, this parallelization strategy distributes the workload across available CPU cores, resulting in an empirically measured performance increase of approximately eight times compared to a sequential implementation. The pseudo-code in Algorithm 1 outlines the core logic for phase matrix computation and the integration of TPL for concurrent execution.

Algorithm 1. Phase calculation algorithm using TPL.

Input parameters:

N —phase map resolution (per one dimension);

hN —center coordinate of a phase map;

C —a vector of Zernike coefficients;

$Z(x, y, p)$ —a matrix of Zernike polynomials values in discrete points;

(x, y) —point in Descartes coordinates;

λ —wavelength;

k —wavenumber.

Output parameters:

$W(x, y)$ —the matrix of phase values;

$W_{SLM}(x, y)$ —the matrix of phase values for the spatial light modulators;

$F(x, y)$ —the matrix of fringe values.

Algorithm pseudocode:

```
#1. Populate Zernike values matrix  $Z(x, y)$ ;
#2. Calculate a wavenumber  $k = 2\pi/\lambda$ ;
#3. Calculate  $hN = (N - 1)/2.0$ ;
#4. Reset matrix  $W(x, y)$ ;
#5. Reset matrix  $W_{SLM}(x, y)$ ;
#6. PARALLEL.FOR from  $int\ i = 0$  to  $i < N$ ;
#7. PARALLEL.FOR from  $int\ i = 0$  to  $i < N$ ;
#8. Define coordinate variable  $X = (i - hN)/hN$ ;
#9. FOR from  $int\ j = 0$  to  $j < N$ ;
#10. Define coordinate variable  $Y = (j - hN)/hN$ ;
#11. IF  $(X, Y) \in UNIT\_CIRCLE$ 
#12. FOR from  $int\ p = 0$  to  $i < ZernikeCount$ 
#13.  $W(x, y) + = C(p) \cdot Z(x, y, p)$ ;
#14.  $W(x, y) + = C(0)$ ;
#15.  $W_{SLM}(x, y) = REMINDER(\frac{W(x, y)}{\lambda}) \cdot 255$ ;
#16.  $F(x, y) = \frac{1 + \cos(W(x, y) \cdot k)}{2}$ ;
#17. Calculate MIN and MAX values of  $W(x, y)$ ;
#18. Calculate MIN and MAX values of  $W_{SLM}(x, y)$ ;
#19. Calculate MIN and MAX values of  $F(x, y)$ ;
#20. ELSE
#21.  $W(x, y) = 0.0$ ;
#22.  $W_{SLM}(x, y) = 0.0$ ;
#23.  $F(x, y) = 0.0$ ;
```

4. Results and Discussion

4.1. Performance Analysis

This section presents an extensive performance analysis of the developed framework, evaluating both the computational efficiency of phase map generation and a comparative assessment of Zernike polynomial calculation against existing solutions. While prior research and available tools typically benchmark the computational efficiency of Zernike polynomial evaluation in isolation, the primary practical objective for an end-user is the rapid generation of derived optical maps. Therefore, the following analysis focuses on

comparing the efficiency of phase, fringe, and PSF map computation under varying input parameters and across different personal computer configurations, characterized by their respective CPU performance capabilities.

The accompanying chart in Figure 5 illustrates the dependence of the computation time for phase, fringe, and SLM-phase maps on the number of Zernike modes, tested on different CPUs with and without the implemented optimizations. Zernike modes were selected in the range of 3 to 100.

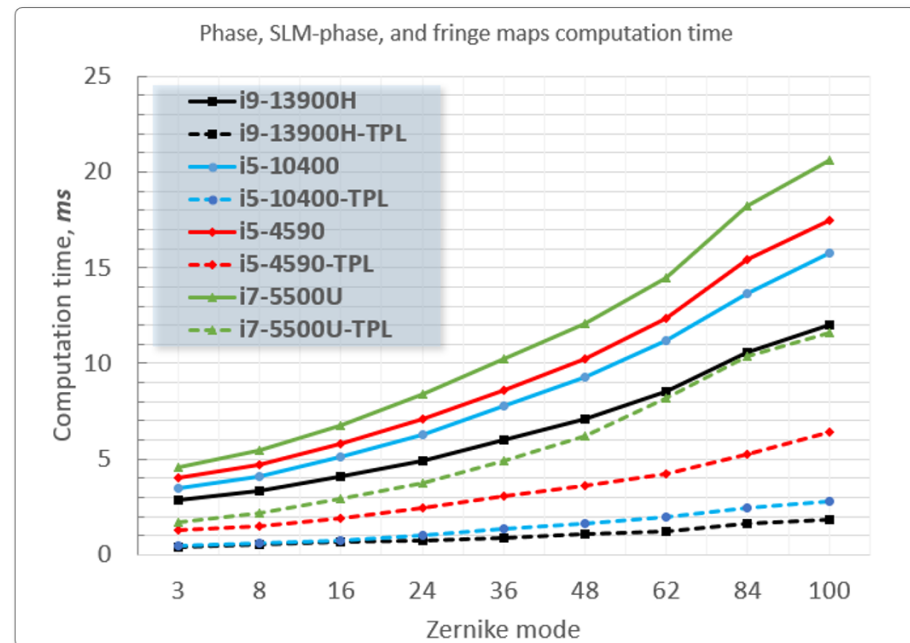


Figure 5. Dependence of a computation time of fringe, phase, and SLM-phase maps on Zernike mode number for various CPU without optimization (solid curves) and with developed optimization (dashed curves).

The computational performance of the framework was evaluated across four CPUs with varying architectures and capabilities. The Intel Core i9-13900H (8 cores, 16 threads) demonstrated the highest computational efficiency. The Intel Core i5-10400 (6 cores, 12 threads) exhibited significantly lower single-threaded performance yet achieved parallelized computation speeds comparable to the i9-13900H. The older-generation processors—specifically, the Intel Core i5-4590 and the Intel Core i7-5500U, both with 4 cores and 4 threads and lacking Hyper-Threading technology—delivered the lowest performance. Notably, the i7-5500U underperformed relative to the i5-4590, a result that is counterintuitive given its higher product tier designation. This discrepancy is attributable to the “U” suffix, which denotes an ultra-low-power mobile processor variant optimized for energy efficiency rather than peak performance in laptop systems. This highlights the critical importance of considering specific CPU microarchitecture and power design, not just core count or product family, when performance is a primary metric. What is even more interesting is that the single-core computation on modern i9-13900H CPU is approaching the parallelized computation on an old i7-5500U CPU.

Figure 6 demonstrates the CPU cores load for both calculation methods—conventional and developed (leveraging parallelism).

Figure 6 demonstrates the efficacy of the optimized algorithms in leveraging the parallel architecture of modern multicore, multithreaded CPUs during phase map calculation. The data clearly indicate a substantial reduction in computation times achieved by the

optimized algorithm. Furthermore, since the underlying TPL is optimized at the hardware level, the performance is designed to scale favorably with the capabilities of the host CPU.

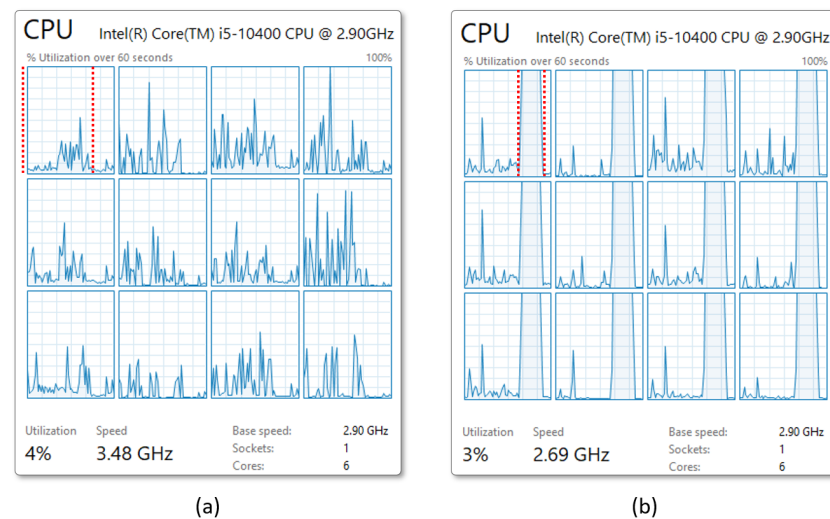


Figure 6. CPU cores load (a) for conventional computation algorithm and (b) for developed computation algorithm that leverages the parallel architecture. Red vertical dashed lines shows the start and the end of computation process.

Although the majority of scientific publications on Zernike polynomials focus on methodological descriptions, a limited number of open-source projects provide the source code used to obtain their results, such as ZERNIPAX [63], ZERNPY [80], ZERNIKE [81], ZERN [82], or FFT/GPU based technique [59]. The availability of these resources enables a direct comparison of computational efficiency.

A comprehensive comparison of the computational efficiency of the aforementioned tools is provided in [63]. We have replicated a similar benchmarking procedure and incorporated the results from our developed tool. The benchmark consisted of calculating the radial part of Zernike polynomials across a unit disk discretized into a 100×100 -point grid. The radial order n was varied from 10 to 100. Figure 7a presents the dependence of computational time (in milliseconds) on the number of Zernike modes (up to 30) on a linear scale, while Figure 7b displays the same relationship for modes from 10 to 100 on a logarithmic scale.

The comparison of results of various packages is presented in Table 1.

Table 1. Comparison of computation efficiency (in ms) of various packages.

Comp. Time, ms	Zernike Mode Number								
	10	12	14	16	18	20	24	26	28
ZERNIPAX	0.11	0.13	0.15	0.17	0.19	0.21	0.25	0.29	0.35
ZERNIKE-VIEWER	0.20	0.24	0.31	0.45	0.67	0.88	1.38	1.67	2.02
ZERN-100	1.01	1.42	1.82	2.51	3.54	4.42	6.81	8.52	11.01
ZERNIKE-100	1.22	1.64	2.21	3.02	4.56	6.04	9.02	11.5	13.8

A comparative analysis of computational methods reveals that the ZERNIKE package and the developed ZernikeViewer framework both employ a direct polynomial evaluation strategy, whereas ZERN and ZERNIPAX utilize the Jacobi recursion relation. Despite this methodological difference, the benchmarking results demonstrate that the algorithmic and parallelization optimizations implemented in ZernikeViewer yield a substantial performance enhancement—reducing computation time by a factor of 7 to 10—even when using

the conventional direct evaluation method. The implementation of the Jacobi recursion relation is anticipated to further decrease computation time and represents a primary objective for the subsequent phase of this ongoing research.

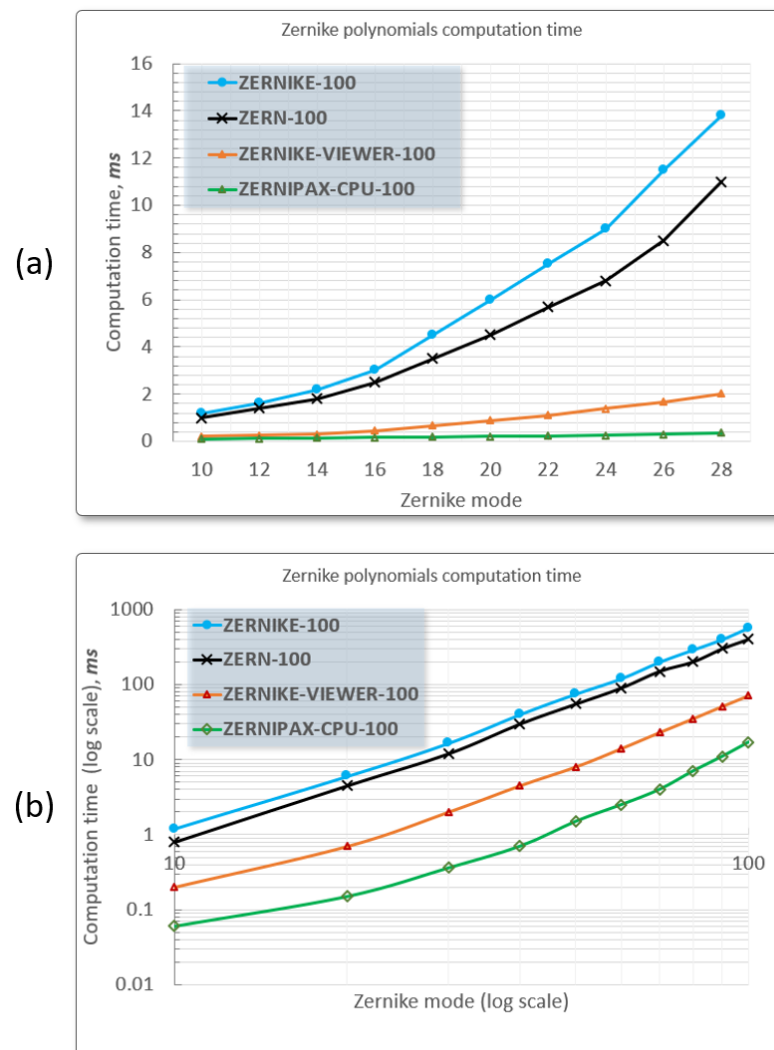


Figure 7. The dependence of the computational time in ms on the number of Zernike modes (a) up to 30 in linear scale, and (b) from 10 to 100 in logarithmic scale. The calculations were done using the PC with the Intel Core i5-10400 CPU (2.8 GHz, 6 cores, 12 threads).

We have also analyzed the memory usage of the developed tool. Since the tool employs double-precision variables to store Zernike polynomial values, the memory complexity scales linearly and can be characterized as $O(N)$, where N is the number of Zernike modes. For example, in tests performed on a workstation with 16 GB of RAM, increasing the number of modes to 50 raised memory usage by approximately 100 MB, corresponding to an increase of about 1%.

4.2. Error Analysis

To quantify the numerical error associated with the direct polynomial evaluation method, a comparative analysis was conducted using the *mpmath* library [83], which facilitates arbitrary-precision arithmetic. The accuracy of the direct calculation for Zernike polynomials of varying radial orders is presented in Figure 8. The reported error is defined as the absolute difference between the polynomial values computed by ZernikeViewer and those obtained using *mpmath*. The results indicate that the numerical error increases pro-

gressively with the radial order n , a characteristic behavior of direct evaluation algorithms for high-order polynomials. Similar calculation was performed by the authors of [63].

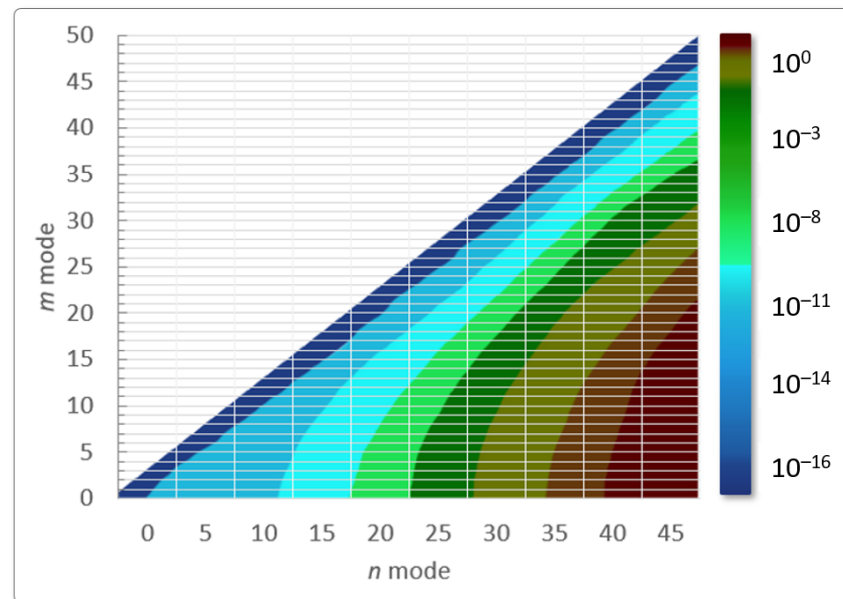


Figure 8. Error analysis of direct Zernike polynomial calculations.

For the error, $\text{MAX}_x |Z(x) - \overline{Z(x)}|$, we have evaluated direct Zernike polynomial calculation at 256 linearly spaced radial points for each (n, m) mode where $n \in [0, 50]$ and $m \in [0, 50]$. After that, we took the maximum absolute value difference with the exact calculation using *mpmath* 100 significant digit precision— $\overline{Z(x)}$, for the error of that mode. As stated in the manuscript, the evaluation was performed using the Intel Core i5-4590.

Regarding the acceptability of the error levels for typical applications of Zernike polynomials, in adaptive optics and astronomy, where wavefront characterization typically relies on low-order Zernike modes, the achieved error levels on the order of 10^{-12} to 10^{-14} are entirely acceptable.

The influence of increasing errors at higher radial orders of Zernike polynomials on point spread function (PSF) maps will be proportional to their influence on phase maps. This relationship arises from the numerical method used to compute PSF maps from intensity and phase distributions. Initially, phase values are converted into sine and cosine components; because a uniform intensity distribution is assumed, the resulting amplitude error is constrained to a range of ± 1 . These values are subsequently processed by a fast Fourier transform (FFT) algorithm. As illustrated in Figure 8, even for the maximum mode values ($n = 50, m = 50$), the peak error in Zernike polynomial calculation remains approximately 1, which is negligible for PSF map computation.

We also estimated the phase computation standard deviation for several discrete resolutions, polynomial numbers and CPUs. We used the following test configuration: number of Zernike polynomials—100, number of calculation repetitions—50, phase map resolution— 256×256 . The results are presented in Figure 9.

It can be seen from the figure above that the RMS of the classical algorithm is almost 3 times smaller compared to the developed parallel algorithm. This is due to the fact that when only one core of a CPU is dedicated to the computation process, it is distracted to a smaller degree to perform other tasks in the operating system. On the contrary, when each core and each thread are used in computation process (refer to Figure 6 for details), most of them might be (and actually will be) distracted.

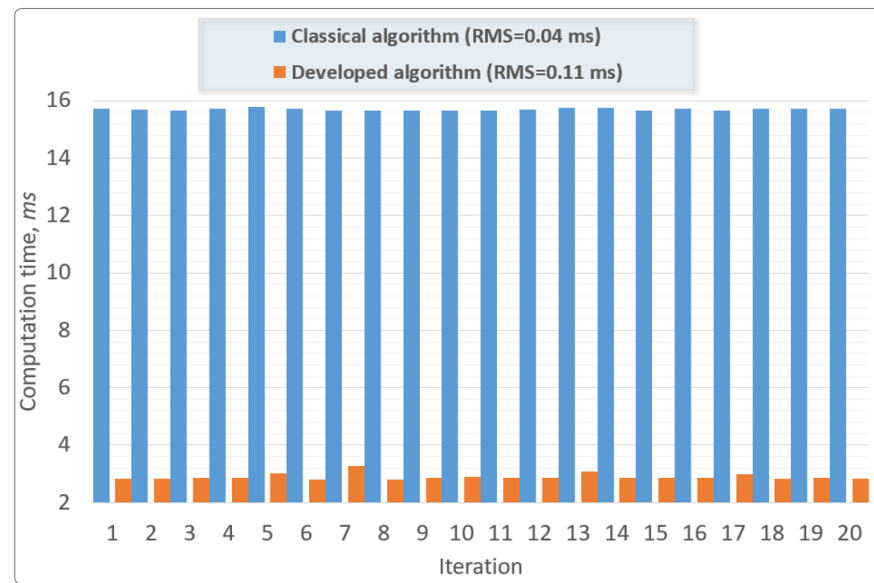


Figure 9. Standard deviation estimation of the classical and developed (parallel) computational algorithms.

4.3. Limitations and Future Work

The developed tool in its current form might be helpful in applications such as wavefront sensing and adaptive optics correction, in optical design and tolerance analysis, in computational microscopy for point-spread function engineering and in ophthalmology for the precise modeling of corneal aberrations.

Future work will address several current limitations of this research. The immediate next step is to implement a fast calculation algorithm based on Jacobi recursion relations to achieve further improvements in computational speed. Subsequently, we plan to integrate wavefront reconstruction from a matrix of wavefront slopes, such as the data produced by a Shack–Hartmann wavefront sensor, as the current inability to directly calculate phase from slopes limits the sensor’s processing speed in this workflow. Another significant limitation of the present tool is its restriction to circular apertures; it cannot yet process wavefronts with square, annular, or arbitrary shapes, which is an important capability for many practical applications, especially in adaptive optics for lasers and astronomy applications. As a longer-term objective, we are considering the development of a mobile version of ZernikeViewer to enhance its accessibility and utility in field settings.

5. Conclusions

In this work, we have presented ZernikeViewer, a software framework designed for the rapid and accurate calculation and visualization of fringe/phase and PSF maps from Zernike coefficients. The framework’s computational core is optimized through the strategic use of the .NET Task Parallel Library to leverage multicore and multithreading CPU architectures, complemented by low-level code optimizations. A comprehensive performance analysis confirmed that these parallelization strategies reduce computation time by a factor of 7 to 10, exemplified by a reduction from 1 ms to 0.1 ms for a radial order $n = 10$, and from 700 ms to 80 ms for $n = 100$.

The architecture incorporates a preloading feature for matrices of precomputed Zernike polynomial values, which accelerates repetitive analyses. The capability for real-time manipulation and reproduction of Zernike coefficients enables precise post-processing of experimentally recorded wavefronts and provides a valuable tool for generating controllable, artificial turbulence in laboratory settings. Furthermore, the software can generate

specialized phase maps with a dynamic range exceeding 2π and project them onto a secondary display, establishing it as a versatile interface for applications involving SLMs. To promote reproducibility and community development, ZernikeViewer has been released as an open-source project, with its source code hosted in a public GitHub repository.

Funding: This research received no external funding.

Data Availability Statement: The datasets presented in this article are not readily available because the data is a part of an ongoing study.

Conflicts of Interest: The author declares that there are no conflicts of interest.

References

1. Niu, K.; Tian, C. Zernike polynomials and their applications. *J. Opt.* **2022**, *24*, 123001. [\[CrossRef\]](#)
2. Lakshminarayanan, V.; Fleck, A. Zernike polynomials: A guide. *J. Mod. Opt.* **2011**, *58*, 545–561. [\[CrossRef\]](#)
3. Galaktionov, I.; Toporovsky, V. Performance Analysis of an Optical System for FSO Communications Utilizing Combined Stochastic Gradient Descent Optimization Algorithm. *Applied System Innovation* **2025**, *8*, 143. [\[CrossRef\]](#)
4. Artal, P.; Rodríguez-Fernández, I.; Yago, I.; Marín, J.; Paniagua-Díaz, A. Optical correction in excised cataractous crystalline lenses by using wavefront shaping techniques. *Biomed. Opt. Express* **2025**, *16*, 3936–3948. [\[CrossRef\]](#) [\[PubMed\]](#)
5. Zhou, Z.; Toda, K.; Kurata, R.; Horie, K.; Horisaki, R.; Ideguchi, T. Phase retrieval via Zernike phase contrast microscopy with an untrained neural network. *Opt. Express* **2025**, *33*, 18698–18708. [\[CrossRef\]](#)
6. Warzecha, M.; Schatz, M.E.; Lucarelli, S.; Jüttner, C. Increasing Structural Performance of Space Telescope Mirrors through Simultaneous Shape and Size Optimization. *Aerospace* **2022**, *9*, 646. [\[CrossRef\]](#)
7. Tam, J.; Wong, H.; Han, Y.; Jin, G.; Chow, H. Measurement of complex surfaces using precision coordinate measuring machine with Zernike algorithms. *Meas. Sci. Technol.* **2025**, *36*, 065009. [\[CrossRef\]](#)
8. Zernike, F. Diffraction theory of the knife-edge test and its improved form, the phase-contrast method. *Mon. Not. R. Astron. Soc.* **1934**, *94*, 377–384. [\[CrossRef\]](#)
9. Zernike, F. Beugungstheorie des schneidenverfahrens und seiner verbesserten form, der phasenkontrastmethode. *Physica* **1934**, *1*, 689–704. [\[CrossRef\]](#)
10. Schwiererling, J.; Greivenkamp, J. Using corneal height maps and polynomial decomposition to determine corneal aberrations. *Optom. Vis. Sci. Off. Publ. Am. Acad. Optom.* **1997**, *74*, 906–916. [\[CrossRef\]](#)
11. Schwiererling, J.; Greivenkamp, J.; Miller, J. Representation of videokeratographic height data with Zernike polynomials. *JOSA A* **1995**, *12*, 2105–2113. [\[CrossRef\]](#) [\[PubMed\]](#)
12. Soloviev, A.; Kotov, A.; Martyanov, M.; Perevalov, S.; Zemskov, R.; Starodubtsev, M.; Aleksandrov, A.; Galaktionov, I.; Samarkin, V.; Kudryashov, A.; et al. Improving focusability of post-compressed PW laser pulses using a deformable mirror. *Opt. Express* **2022**, *30*, 40584–40591. [\[CrossRef\]](#) [\[PubMed\]](#)
13. Iskander, D.; Collins, M.; Davis, B. Optimal modeling of corneal surfaces with Zernike polynomials. *IEEE Trans. Biomed. Eng.* **2001**, *48*, 87–95. [\[CrossRef\]](#) [\[PubMed\]](#)
14. Kotov, A.; Perevalov, S.; Starodubtsev, M.; Zemskov, R.; Alexandrov, A.; Galaktionov, I.; Kudryashov, A.; Samarkin, V.; Soloviev, A. Adaptive system for correcting optical aberrations of high-power lasers with dynamic determination of the reference wavefront. *Quantum Electron.* **2021**, *51*, 593–596. [\[CrossRef\]](#)
15. Rodrigues, M.; Rosa, A.; Vieira, N.; Murta, J. Modeling ophthalmic surfaces using Zernike, Bessel and Chebyshev type functions. *J. Phys. Conf. Ser.* **2019**, *1194*, 012093. [\[CrossRef\]](#)
16. Klyce, S.; Karon, M.; Smolek, M. Advantages and disadvantages of the Zernike expansion for representing wave aberration of the normal and aberrated eye. *J. Refract. Surg.* **2004**, *20*, S537–541. [\[CrossRef\]](#)
17. Soloviev, A.; Kotov, A.; Perevalov, S.; Eshyunin, M.; Starodubtsev, M.; Alexandrov, A.; Galaktionov, I.; Samarkin, V.; Kudryashov, A.; Ginzburg, V.; et al. Adaptive system for wavefront correction of the PEARL laser facility. *Quantum Electron.* **2020**, *50*, 1115–1122. [\[CrossRef\]](#)
18. Skvortsov, A.; Zuev, S.; Koryachko, M.; Voloshinov, E. Specific features of motion of molten zones in the field of silicon structural inhomogeneity. *Tech. Phys. Lett.* **2017**, *43*, 705–707. [\[CrossRef\]](#)
19. Koryachko, M.; Pshonkin, D.; Skvortsov, A. Features of melt droplet formation during electrical destruction aluminum films on the semiconductor surface. *Defect Diffus. Forum* **2021**, *410*, 737–741. [\[CrossRef\]](#)
20. Wan, K.; Gu, Z.; Yan, C. An analytic expression for the field dependence of three-order Zernike aberrations in decentered and/or tilted optical systems. *Opt. Commun.* **2024**, *565*, 130696. [\[CrossRef\]](#)

21. Capalbo, V.; De Petris, M.; De Luca, F.; Cui, W.; Yepes, G.; Knebe, A.; Rasia, E. The Three Hundred project: Quest of clusters of galaxies morphology and dynamical state through Zernike polynomials. *Mon. Not. R. Astron. Soc.* **2021**, *503*, 6155–6169. [\[CrossRef\]](#)
22. Bará, S.; Nievas, M.; de Miguel, A.S.; Zamorano, J. Zernike analysis of all-sky night brightness maps. *Appl. Opt.* **2014**, *53*, 2677–2686. [\[CrossRef\]](#) [\[PubMed\]](#)
23. Skvortsov, A.; Koryachko, M.; Zuev, S.; Demchenkova, A. Melt drops movement over semiconductor surfaces controlled by electric field. *Period. Tche Quim.* **2019**, *16*, 681–687. [\[CrossRef\]](#)
24. Skvortsov, A.A.; Koryachko, M.V.; Skvortsov, P.A.; Luk'yAnov, M.N. On the Issue of Crack Formation in a Thin Dielectric Layer on Silicon under Thermal Shock. *J. Mater. Eng. Perform.* **2020**, *29*, 4390–4395. [\[CrossRef\]](#)
25. Skvortsov, A.A.; Pshonkin, D.; Koryachko, M.V.; Rybakova, M.R. The Effect of Constant Magnetic Fields on the Dynamics of Molten Zones in the Field of Structural Inhomogeneity of Silicon. *Tech. Phys. Lett.* **2018**, *44*, 498–501. [\[CrossRef\]](#)
26. Bará, S.; Tilve, V.; Nievas, M.; Miguel, A.; Zamorano, J. Zernike power spectra of clear and cloudy light-polluted urban night skies. *Appl. Opt.* **2015**, *54*, 4120–4129. [\[CrossRef\]](#)
27. Qu, Z.; Pfefferlé, D.; Hudson, S.; Baillod, A.; Kumar, A.; Dewar, R.; Hole, M. Coordinate parameterisation and spectral method optimization for Beltrami field solver in stellarator geometry. *Plasma Phys. Control. Fusion* **2020**, *62*, 124004. [\[CrossRef\]](#)
28. Dudt, D.; Kolemen, E. DESC: A stellarator equilibrium solver. *Phys. Plasmas* **2020**, *27*, 102513. [\[CrossRef\]](#)
29. Hudson, S.; Dewar, R.; Hole, M.; McGann, M. Nonaxisymmetric, multi-region relaxed magnetohydrodynamic equilibrium solutions. *Plasma Phys. Control. Fusion* **2011**, *54*, 014005. [\[CrossRef\]](#)
30. Boyd, J.; Yu, F. Comparing seven spectral methods for interpolation and for solving the Poisson equation in a disk: Zernike polynomials, Logan-Shepp ridge polynomials, Chebyshev-Fourier Series, cylindrical Robert functions, Bessel-Fourier expansions, square-to-disk confor. *J. Comput. Phys.* **2011**, *230*, 1408–1438. [\[CrossRef\]](#)
31. Martínez-Finkelshtein, A.; Delgado, A.; Castro, G.; Zarzo, A.; Alió, J. Comparative analysis of some modal reconstruction methods of the shape of the cornea from the corneal elevation data. *Invest. Ophthalmol. Vis. Sci.* **2009**, *50*, 5639–5645. [\[CrossRef\]](#) [\[PubMed\]](#)
32. Ares, M.; Royo, S. Comparison of cubic B-spline and Zernike-fitting techniques in complex wavefront reconstruction. *Appl. Opt.* **2006**, *45*, 6945–6964. [\[CrossRef\]](#) [\[PubMed\]](#)
33. Schneider, M.; Iskander, D.; Collins, M.J. Modeling corneal surfaces with rational functions for high-speed videokeratometry data compression. *IEEE Trans. Biomed. Eng.* **2009**, *56*, 493–499. [\[CrossRef\]](#) [\[PubMed\]](#)
34. Martínez-Finkelshtein, A.; Ramos-López, D.; Iskander, D. Computation of 2D Fourier transforms and diffraction integrals using Gaussian radial basis functions. *Appl. Comput. Harmon. Anal.* **2017**, *43*, 424–448. [\[CrossRef\]](#)
35. Mahajan, V. Zernike annular polynomials and optical aberrations of systems with annular pupils. *Appl. Opt.* **1994**, *33*, 8125–8127. [\[CrossRef\]](#)
36. Mahajan, V.; Dai, G. Orthonormal polynomials for hexagonal pupils. *Opt. Lett.* **2006**, *31*, 2462–2464. [\[CrossRef\]](#)
37. Mahajan, V.; Dai, G. Orthonormal polynomials in wavefront analysis: Analytical solution. *J. Opt. Soc. Am. A* **2007**, *24*, 2994–3016. [\[CrossRef\]](#)
38. Navarro, R.; López, J.; Díaz, J.; Sinusía, E. Generalization of Zernike polynomials for regular portions of circles and ellipses. *Opt. Express* **2014**, *22*, 21263–21279. [\[CrossRef\]](#)
39. Ferreira, C.; López, J.; Navarro, R.; Pérez Sinusía, E. Zernike-like systems in polygons and polygonal facets. *Appl. Opt.* **2015**, *54*, 6575–6583. [\[CrossRef\]](#)
40. Furter, A. Orthogonal polynomials of two variables defined on a rectangle. *Proc. Acad. Sci. USSR* **1956**, *110*, 963–966. (In Russian)
41. He, P.; Zong, X.; Fu, R. Zernike polynomials for rectangular area and applications in off-axis asphere testing. In *Proc. SPIE 7656, 5th International Symposium on Advanced Optical Manufacturing and Testing Technologies: Optical Test and Measurement Technology and Equipment*; SPIE Optical Metrology: Munich, Germany, 2010; p. 765615. [\[CrossRef\]](#)
42. Navarro, R.; Arines, J.; Rivera, R. Direct and inverse discrete Zernike transform. *Opt. Express* **2009**, *17*, 24269–24281. [\[CrossRef\]](#) [\[PubMed\]](#)
43. Cuyt, A.; Yaman, I.; Ibrahimoglu, B.; Benouahmane, B. Radial orthogonality and Lebesgue constants on the disk. *Numer. Algorithms* **2012**, *61*, 291–313. [\[CrossRef\]](#)
44. Ramos-López, D.; Sánchez-Granero, M.A.; Fernández-Martínez, M.; Martínez-Finkelshtein, A. Optimal sampling patterns for Zernike polynomials. *Appl. Math. Comput.* **2016**, *274*, 247–257. [\[CrossRef\]](#)
45. Carnicer, J.; Godés, C. Interpolation on the disk. *Numer. Algorithms* **2014**, *66*, 1–16. [\[CrossRef\]](#)
46. Meurant, G.; Sommariva, A. On the computation of sets of points with low Lebesgue constant on the unit disk. *J. Comput. Appl. Math.* **2019**, *345*, 388–404. [\[CrossRef\]](#)
47. Díaz-Elbal, S.; Martínez-Finkelshtein, A.; Ramos-López, D. Sampling patterns for Zernike-like bases in non-standard geometries. *Appl. Math. Comput.* **2026**, *511*, 129727. [\[CrossRef\]](#)
48. Chong, C.-W.; Raveendran, P.; Mukundan, R. A comparative analysis of algorithms for fast computation of Zernike moments. *Pattern Recognit.* **2003**, *36*, 731–742. [\[CrossRef\]](#)

49. Novák, P.; Novák, J. Efficient and stable numerical method for evaluation of Zernike polynomials and their Cartesian derivatives. In *Proc. SPIE 8789, Modeling Aspects in Optical Metrology IV, San Francisco, CA, USA, 2–4 February 2013*; SPIE Optical Metrology: Munich, Germany, 2013; p. 878913. [\[CrossRef\]](#)
50. Singh, C.; Walia, E. Fast and numerically stable methods for the computation of Zernike moments. *Pattern Recognit.* **2010**, *43*, 2497–2506. [\[CrossRef\]](#)
51. Deng, A.-W.; Wei, C.-H.; Gwo, C.-Y. Stable, fast computation of highorder Zernike moments using a recursive method. *Pattern Recognit.* **2016**, *56*, 16–25. [\[CrossRef\]](#)
52. Papakostas, G.; Boutalis, Y.; Papaodysseus, C.; Fragoulis, D. Numerical stability of fast computation algorithms of Zernike moments. *Appl. Math. Comput.* **2008**, *195*, 326–345. [\[CrossRef\]](#)
53. Kintner, E.C. A Recurrence Relation for Calculating the Zernike Polynomials. *Opt. Acta Int. J. Opt.* [\[CrossRef\]](#)
54. Prata, A.; Rusch, W.V.T. Algorithm for computation of Zernike polynomials expansion coefficients. *Appl. Opt.* **1989**, *28*, 749–754. [\[CrossRef\]](#) [\[PubMed\]](#)
55. Chen, X.; Zhong, S.; Hou, Y.; Cao, R.; Wang, W.; Li, D.; Dai, Q.; Kim, D.; Xi, P. Superresolution structured illumination microscopy reconstruction algorithms: A review. *Light Sci. Appl.* **2023**, *12*, 172. [\[CrossRef\]](#) [\[PubMed\]](#)
56. Li, Z.S.; Liu, C.; Li, X.W.; Zheng, Y.; Huang, Q.; Zheng, Y.-W.; Hou, Y.-H.; Chang, C.-L.; Zhang, D.-W.; Zhuang, S.-L.; et al. Real-time holographic camera for obtaining real 3D scene hologram. *Light. Sci. Appl.* **2025**, *14*, 74. [\[CrossRef\]](#)
57. Qin, H.; Qin, L.; Xue, L.; Yu, C. A parallel recurrence method for the fast computation of Zernike moments. *Appl. Math. Comput.* **2012**, *219*, 1549–1561. [\[CrossRef\]](#)
58. Janssen, A. Computing Zernike polynomials of arbitrary degree using the discrete Fourier transform. *J. Eur. Opt. Soc. Publ.* **2007**, *2*, 07012. [\[CrossRef\]](#)
59. Al-Rawi, M. Ultra-Fast Zernike Moments using FFT and GPU. *arXiv* **2023**. [\[CrossRef\]](#)
60. Zhang, H.-Y.; Zhou, Y.; Feng, Z.-Q. Balanced Binary Tree Schemes for Computing Zernike Radial Polynomials. *IEEE Access* **2023**, *11*, 106567–106579. [\[CrossRef\]](#)
61. Belova, O.S.; Bolotov, D.V.; Kazantsev, S.Y.; Klubkov, A.V.; Kolesnikov, O.V.; Paimtsev, G.P.; Yarygin, M.A. The influence of pulsed electromagnetic action on a quantum key distribution. *Russ. Electr. Eng.* **2025**, *96*, 289–294. [\[CrossRef\]](#)
62. Boev, A.A.; Vorobey, S.S.; Kazantsev, S.Y.; Kernosov, M.Y.; Kolesnikov, O.V.; Kuznetsov, S.N.; Mironov, Y.B.; Parshin, A.A.; Rudavin, N.V. Possibility of creating a modular system for quantum key distribution in the atmosphere. *Tech. Phys. Lett.* **2022**, *48*, 11–14. [\[CrossRef\]](#)
63. Elmacioglu, Y.; Conlin, R.; Dudt, D.; Panici, D.; Kolemen, E. ZERNIPAX: A fast and accurate Zernike polynomial calculator in Python. *Appl. Math. Comput.* **2025**, *505*, 129534. [\[CrossRef\]](#)
64. Kim, S.; Youk, Y.; Kim, G.; Ryu, D.; Yoon, J. ZernikeNet: A deep learning-based approach for accurate Zernike coefficients calculation in aspheric optical components. In *Proc. SPIE 12677, Astronomical Optics: Design, Manufacture, and Test of Space and Ground Systems IV, Edinburgh, UK, 17–21 April 2023*; SPIE Digital Library: Bellingham, WA, USA, 2023; Volume 12677, p. 126770S. [\[CrossRef\]](#)
65. Wyant College of Optical Sciences. Available online: <https://webs.optics.arizona.edu/gsmith/Zernike.html> (accessed on 23 February 2026).
66. Gatinel, D. Available online: <https://www.gatinel.com/recherche-formation/wavefront-sensing/wavefront-analyzer-zernike-optics-simulator/> (accessed on 23 February 2026).
67. Wolfram Projects. Available online: <https://demonstrations.wolfram.com/PlotsOfZernikePolynomials/> (accessed on 23 February 2026).
68. Optics Website. Available online: <https://www.opticsthewebsite.com/Zernike> (accessed on 23 February 2026).
69. Luo, Z.; Zheng, Q.; Hei, X.; Giacaman, N. Parallel Programming Based on Microsoft.NET TPL. In *Proceedings of the 2nd International Conference on Computer Science and Electronics Engineering (ICCSEE 2013)*, Dubai, United Arab Emirates, 24–25 March 2013. [\[CrossRef\]](#)
70. Toporovsky, V.; Galaktionov, I. Modal wavefront reconstruction of fixed aberrations for piezoelectric deformable mirrors. *Lect. Notes Electr. Eng.* **2026**, *1521*, 286–298. [\[CrossRef\]](#)
71. Lewis, H.R.; Bellan, P.M. Physical constraints on the coefficients of Fourier expansions in cylindrical coordinates. *J. Math. Phys.* **1990**, *31*, 2592–2596. [\[CrossRef\]](#)
72. Galaktionov, I.; Toporovsky, V. Gradient descent algorithm for automatic optimization of a laser beam cross-section in free-space optical communication applications. *Lect. Notes Electr. Eng.* **2026**, *1521*, 231–240. [\[CrossRef\]](#)
73. Wyant, J.C.; Creath, K. Basic Wavefront Aberration Theory for Optical Metrology. *Appl. Opt. Opt. Eng.* **1992**, *11*, 28–39.
74. Southwell, W.H. Wavefront estimation from wavefront slope measurements. *J. Opt. Soc. Am.* **1980**, *70*, 998–1005. [\[CrossRef\]](#)
75. Simpkins, J.; Stevenson, R. A spatially varying PSF model for Seidel aberrations and defocus. In *Proc. SPIE 8666, Visual Information Processing and Communication IV, San Francisco, CA, USA, 21–23 January 2013*; SPIE Digital Library: Bellingham, WA, USA, 2020; Volume 8666, p. 86660F. [\[CrossRef\]](#)

76. Mahajan, V. Zernike polynomials and aberration balancing. In *Proc. SPIE 5173, Current Developments in Lens Design and Optical Engineering IV, San Francisco, CA, USA, 13–14 August 2003*; SPIE Digital Library: Bellingham, WA, USA, 2020; Volume 5173, p. 517302. [\[CrossRef\]](#)
77. Galaktionov, I.; Toporovsky, V. A Multi-Stage Algorithm of Fringe Map Reconstruction for Fiber-End Surface Analysis and Non-Phase-Shifting Interferometry. *Appl. Syst. Innov.* **2026**, *9*, 31. [\[CrossRef\]](#)
78. Galaktionov, I.; Nikitin, A.; Sheldakova, J.; Toporovsky, V.; Kudryashov, A. Numerical Analysis of Wavefront Approximation Accuracy by Means of Zernike Polynomials for Optical Surface Flatness Measurements Using a Hartmannometer Device. *Opt. Mem. Neural Netw.* **2024**, *33*, S146–S157. [\[CrossRef\]](#)
79. Galaktionov, I.; Toporovsky, V. A Multi-Deformable-Mirror 500 Hz Adaptive Optical System for Atmospheric Turbulence Simulation, Real-Time Reconstruction, and Wavefront Correction Using Bimorph and Tip-Tilt Correctors. *Photonics* **2025**, *12*, 592. [\[CrossRef\]](#)
80. Klykov, S. Zernpy. Available online: <https://github.com/sklykov/zernpy.git> (accessed on 23 February 2026).
81. Antonello, J. Zernike. Available online: <https://github.com/jacopoantonello/zernike.git> (accessed on 23 February 2026).
82. Menduina, A. Zern. Available online: <https://github.com/AlvaroMenduina/ZERN.git> (accessed on 23 February 2026).
83. The Mpmath Development Team. Mpmath: A Python Library for Arbitraryprecision Floating-Point Arithmetic (Version 1.3.0). 2023. Available online: <https://mpmath.org/> (accessed on 23 February 2026).

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.