

Supplementary Information for Temporal Super-Resolution for High-Speed 3D Imaging Using Triple-Frequency Color-Multiplexed Fringe Projection

Yifan Liu[†] Wenwu Chen[†] Shijie Feng^{*} Yutong Xiao Jinyang Jiang Shengqi Yu Yiheng Liu
Wei Yin Qian Chen^{*} Chao Zuo^{*}

Y. Liu, W. Chen, S. Feng, Y. Xiao, J. Jiang, S. Yu, Y. Liu, W. Yin, Q. Chen, C. Zuo
Smart Computational Imaging Laboratory (SCILab), School of Electronic and Optical Engineering, Nanjing University of Science and Technology, Nanjing, Jiangsu Province 210094, China
Smart Computational Imaging Research Institute (SCIRI) of Nanjing University of Science and Technology, Nanjing, Jiangsu Province 210019, China
Jiangsu Key Laboratory of Visual Sensing and Intelligent Perception, Nanjing, Jiangsu Province 210094, China
State Key Laboratory of Extreme Environment Optoelectronic Dynamic Measurement Technology and Instrument, Taiyuan, Shanxi Province 030051, China
Nanjing University of Science and Technology, Nanjing, Jiangsu Province 210094, China
Email: shijiefeng@njust.edu.cn; chenqian@njust.edu.cn; zuochao@njust.edu.cn

[†] These authors contributed equally to this work.

Keywords: *Network, A3DR, Physics-embedded deep learning, Data acquisition, Digital twin*

This Supplementary Information provides implementation details and additional validations for the proposed TFCMFPP framework. We first describe the network architectures used throughout the pipeline, including TriPath-CNN, MultiResUnet, the adaptive Gating Network, and the U-Net backbone. We then present the system calibration procedure and the Augmented 3D Reconstruction (A3DR) strategy that establish an accurate phase-to-coordinate mapping. Next, we detail the physics-informed training methodology, including the virtual system setup for digital-twin-based synthetic data generation. Finally, we provide the protocols for real-data acquisition and high-precision ground-truth generation, and summarize the design principles for selecting angular multiplexing parameters.

Contents

1 Network Architecture

1.1 TriPath-CNN

1.2 MultiResUnet

1.3 Gating Network

1.4 U-Net

2 System Calibration

3 Augmented 3D Reconstruction (A3DR)

4 Virtual System Setup and Digital Twin Implementation

5 Data Acquisition and Ground-Truth Generation

6 Design Principles for Angular Multiplexing Parameters

1 Network Architecture

In this section, we briefly introduce four different network architectures used in the TFCMFPP method: **TriPath-CNN**, **MultiResUnet**, **U-Net**, and the **Gating Network**.

1.1 TriPath-CNN

In the proposed TFCMFPP decoding framework, TriPath-CNN is employed as the backbone of the first-stage preprocessing network, ZOCCR-Net (Zero-Order and Color-Crosstalk Removal Network). Its primary function is to remove the zero-order components and color-channel crosstalk present in the long-exposure multiplexed fringe images, thereby providing clean input features for the subsequent frequency decomposition and phase-recovery stages.

TriPath-CNN is a lightweight three-branch convolutional framework designed to capture features across complementary receptive fields. As illustrated in Fig. S1, the network processes the input simultaneously through three distinct paths, each tailored to extract information at different scales and levels of abstraction [1]. In the first path, the input is passed through a shallow stack of convolutional layers, which preserves high-resolution spatial details and captures fine-grained local features. The second path incorporates additional convolutional blocks, effectively enlarging the receptive field to encode mid-range contextual cues. The third path applies progressive downsampling before convolution, enabling the extraction of coarse semantic representations that complement the finer features learned by the other paths. To improve computational efficiency, each branch factorizes large convolutional kernels into successive 3×3 convolutions [2], reducing memory consumption while retaining representational capacity. Residual shortcuts are also introduced to stabilize training and alleviate feature degradation [3]. By optimizing the depth and width of these branches, the TriPath-CNN maintains a compact parameter scale of approximately 2.0 M, facilitating rapid preprocessing without compromising accuracy. After feature extraction, the outputs of the three branches are merged via a concatenation block, producing a multi-channel feature tensor that integrates detailed, contextual, and abstracted representations. Finally, a concluding convolutional layer projects this fused representation into the desired output space, allowing TriPath-CNN to balance global semantic understanding with local spatial precision.

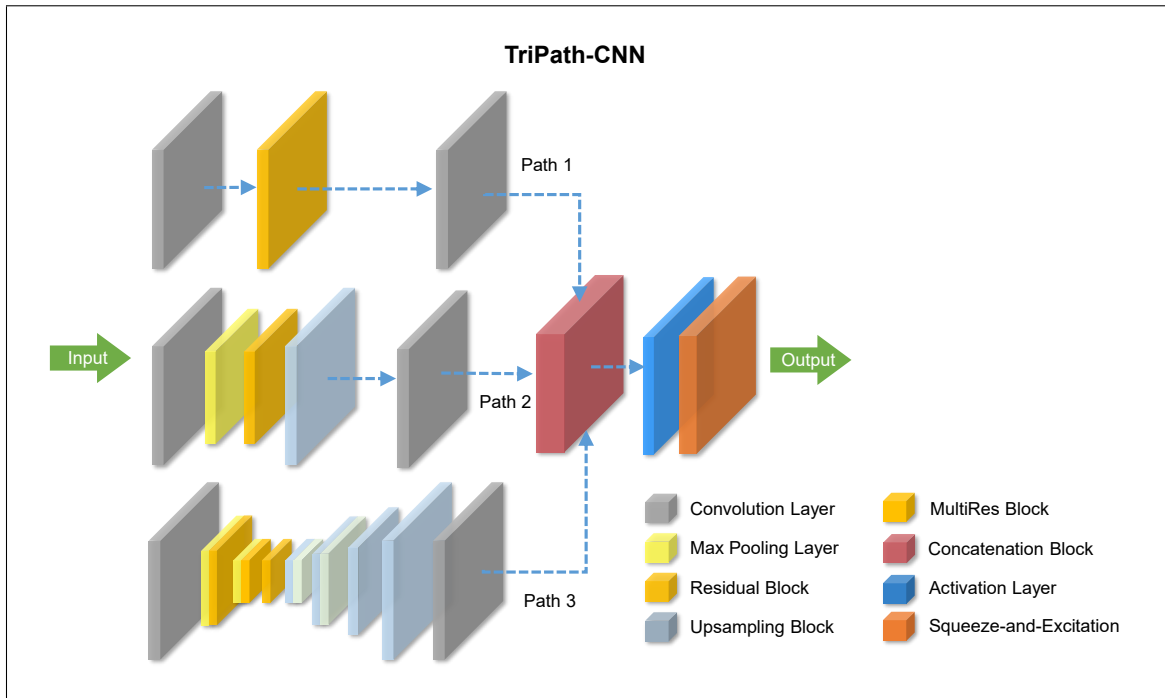


Figure S1: Network architecture of TriPath-CNN.

1.2 MultiResUnet

Within the proposed TFCMFPP framework, MultiResUnet serves as the backbone of the second-stage network, SFD-Net (Spatial-Frequency Decomposition Network). Its primary objective is to decompose the pre-processed color-multiplexed fringe images into mutually independent directional components by jointly learning representations in both the spatial and frequency domains. This enables effective disentanglement of spectral-directional information embedded within a single long-exposure color fringe image. The network takes as input the zero-order-removed color fringe images processed by ZOCC-Net, denoted as $\{I_{LE}^{dR}, I_{LE}^{dG}, I_{LE}^{dB}\}$, together with their corresponding Fourier spectra $\{F_{LE}^{dR}, F_{LE}^{dG}, F_{LE}^{dB}\}$. By fusing complementary features across these two domains, SFD-Net reconstructs a sequence of demodulated directional fringe patterns, expressed as $B_m \cos \phi_m$. These recovered fringe components are subsequently fed into the third-stage phase-recovery network (FDD-Net) for precise estimation of multi-directional absolute phase maps.

MultiResUnet, introduced as a potential successor to the renowned U-Net framework, utilizes a novel design to enhance segmentation capabilities. As depicted in Fig. S2(a), the overall architecture retains the encoder-decoder scheme of the classical U-Net but incorporates additional modules to strengthen feature representation. Specifically, instead of directly concatenating encoder and decoder outputs, a residual path (Res-path) structure is employed [see Fig. S2(b)], integrating multiple 1×1 and 3×3 convolutional layers along shortcut connections. This design alleviates the feature discrepancy between encoder and decoder, thereby facilitating smoother information propagation. Moreover, the conventional two-layer convolution sequence found in the classical U-Net is replaced by MultiRes blocks [Fig. S2(c)]. In this scheme, the filter count increases progressively over three layers, and a residual connection—supported by a 1×1 filter for dimension preservation—is introduced. To further optimize efficiency, the MultiRes block factorizes costly 5×5 and 7×7 filters into a series of 3×3 convolutions, reconciling features of varying context sizes, reducing memory demands, and accelerating network training. Through this high-efficiency integration of multi-scale residual features, the MultiResUnet architecture achieves a robust balance between representational power and computational cost, encompassing approximately 16.0 M parameters. Together, these modifications enable MultiResUnet to enhance both local detail preservation and global contextual understanding, thus improving overall segmentation performance.

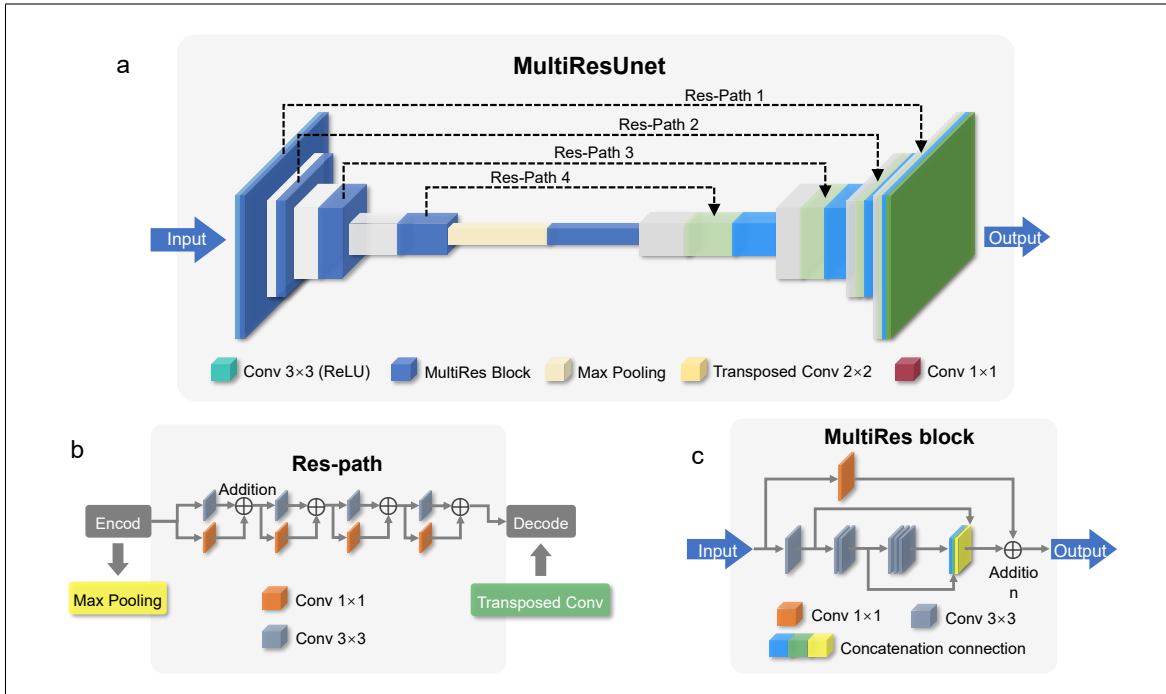


Figure S2: Network architecture of MultiResUnet.

1.3 Gating Network

Within the proposed TFCMFPP decoding framework, the Gating Network serves as a key component of the third-stage network, FDD-Net (Fringe Direction Decoding Network). It is designed to perform adaptive expert selection and feature allocation within the multi-directional mixture-of-experts system. While FDD-Net focuses on high-precision phase recovery for fringe patterns with different orientations and frequencies, the Gating Network functions as the control module that predicts routing weights to determine which expert sub-network each input feature should be directed to, thereby realizing a direction-aware decoding mechanism.

In the overall workflow, SFD-Net outputs a set of directional fringe patterns $\{B_m \cos \phi_m\}$, which are used as the inputs of the Gating Network. The network analyzes the directional characteristics of these fringe patterns. It then produces a direction-aware weight vector $\mathbf{w} = [w_1, w_2, \dots, w_M]^T$, where M denotes the number of candidate directions and each element w_i represents the confidence that the current fringe pattern belongs to the i -th expert module. This direction vector subsequently guides the expert assignment process in the mixture-of-experts framework, enabling each input fringe image to be adaptively routed to the most suitable expert network for accurate phase decoding.

The Gating Network, illustrated in Fig. S3, is a compact decision module designed to generate adaptive weights for expert selection. Despite its multi-scale feature extraction capability, the Gating Network maintains a minimalist architecture with approximately 0.5 M parameters. The input feature map of size $W \times H \times C$ is progressively encoded through a cascade of convolutional and pooling layers. In the first stage, the spatial dimensions are reduced to $\frac{W}{2} \times \frac{H}{2} \times 2C$, followed by successive convolution–pooling operations that further downsample the resolution to $\frac{W}{4} \times \frac{H}{4} \times 4C$, $\frac{W}{8} \times \frac{H}{8} \times 8C$, $\frac{W}{16} \times \frac{H}{16} \times 16C$, and finally $\frac{W}{32} \times \frac{H}{32} \times 32C$. This progressive reduction enlarges the receptive field while capturing multi-scale contextual features.

Subsequently, the compressed representation is flattened and passed through a series of fully connected layers, which project the high-dimensional embedding into a compact gating vector. The final output is a weight vector $\mathbf{w} \in \mathbb{R}^{M \times 1}$, where each element corresponds to the activation strength of one expert branch in the mixture-of-experts framework [4, 5]. By integrating convolutional feature extraction, hierarchical downsampling, and dense projection, the Gating Network provides an efficient mechanism for adaptive expert selection and feature routing [6].

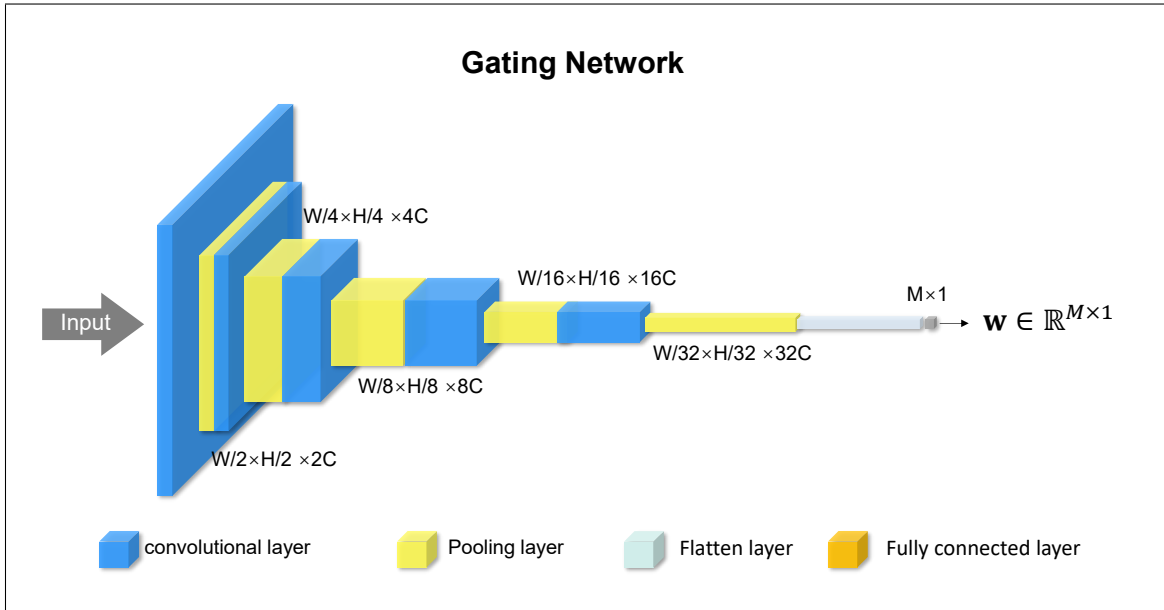


Figure S3: Network architecture of Gating Network.

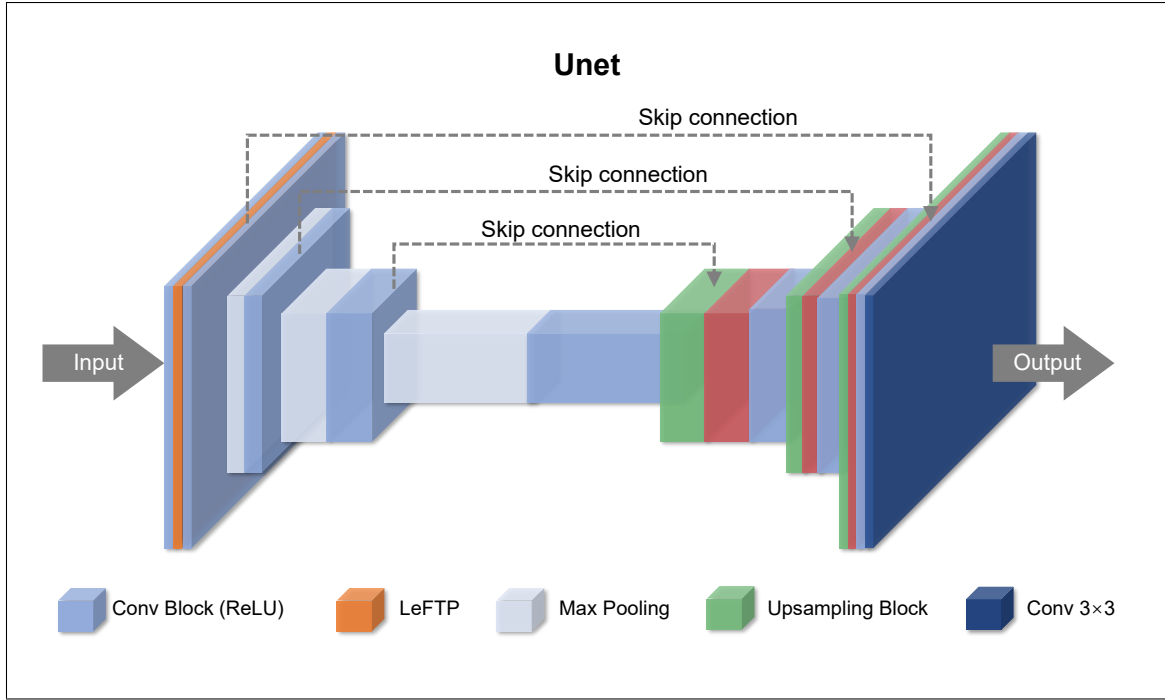


Figure S4: Network architecture of U-Net.

1.4 U-Net

In the proposed TFCMFPP framework, U-Net is employed as a key component of the third-stage network, FDD-Net (Fringe Direction Decoding Network), to perform phase prediction. U-Net is a widely adopted architecture for biomedical image segmentation, originally proposed by Ronneberger et al. [7]. As illustrated in Fig. S4, the network consists of two symmetric paths: a contracting (encoder) path and an expanding (decoder) path. This U-Net based phase prediction module is optimized for high-fidelity reconstruction with a moderate parameter scale of approximately 4.0 M. In the encoder, each stage is composed of two successive convolutional layers, followed by a downsampling operation that progressively reduces the spatial resolution. This design enables the network to capture multi-scale contextual information from the input images. To further enhance the encoder, we incorporate a learning-enhanced Fourier transform profilometry (LeFTP) module, which leverages prior knowledge of Fourier transform profilometry to perform physics-informed augmented fringe pattern analysis (PI-AFPA). More details about the LeFTP module can be found in Yin et al. [8]. In the decoder, the upsampling path gradually restores the spatial resolution, while two consecutive convolutions refine the feature representations at each level. Crucially, skip connections link corresponding stages of the encoder and decoder, allowing high-resolution features from the contracting path to be combined with the upsampled features. These skip connections help preserve spatial details and mitigate the information loss introduced by pooling operations, making U-Net particularly effective for accurate and detailed segmentation tasks.

2 System Calibration

To accurately determine the geometric relationship between the camera and the projector in the TFCMFPP setup, a 9×11 calibration board with evenly distributed white circular markers on a black background was employed. The distance between adjacent marker centers was precisely 15 mm. As illustrated in Fig. S5, nine images of the calibration board were captured from different positions and orientations to comprehensively cover the measurement volume of $450 \text{ mm} \times 350 \text{ mm} \times 350 \text{ mm}$. These images served as the input data for intrinsic and extrinsic parameter estimation.

During calibration, the projector was modeled as an inverse camera, following the flexible calibration framework proposed by Zhang [9]. The pre-calibrated color camera assisted the process by capturing the projected fringe patterns on the calibration board. Two orthogonal sets of sinusoidal fringe patterns with spatial periods of 1, 8, and 64 along the x -axis, and 1, 8, and 48 along the y -axis, were sequentially projected. For the highest-frequency fringes, a 16-step phase-shifting (PS) algorithm [10] was employed to obtain high-precision wrapped phases. By combining the PS method with multi-frequency temporal phase unwrapping (MF-TPU) [11], absolute phase maps were retrieved, establishing pixel-to-pixel correspondence between the camera sensor and the projector's digital micromirror device (DMD).

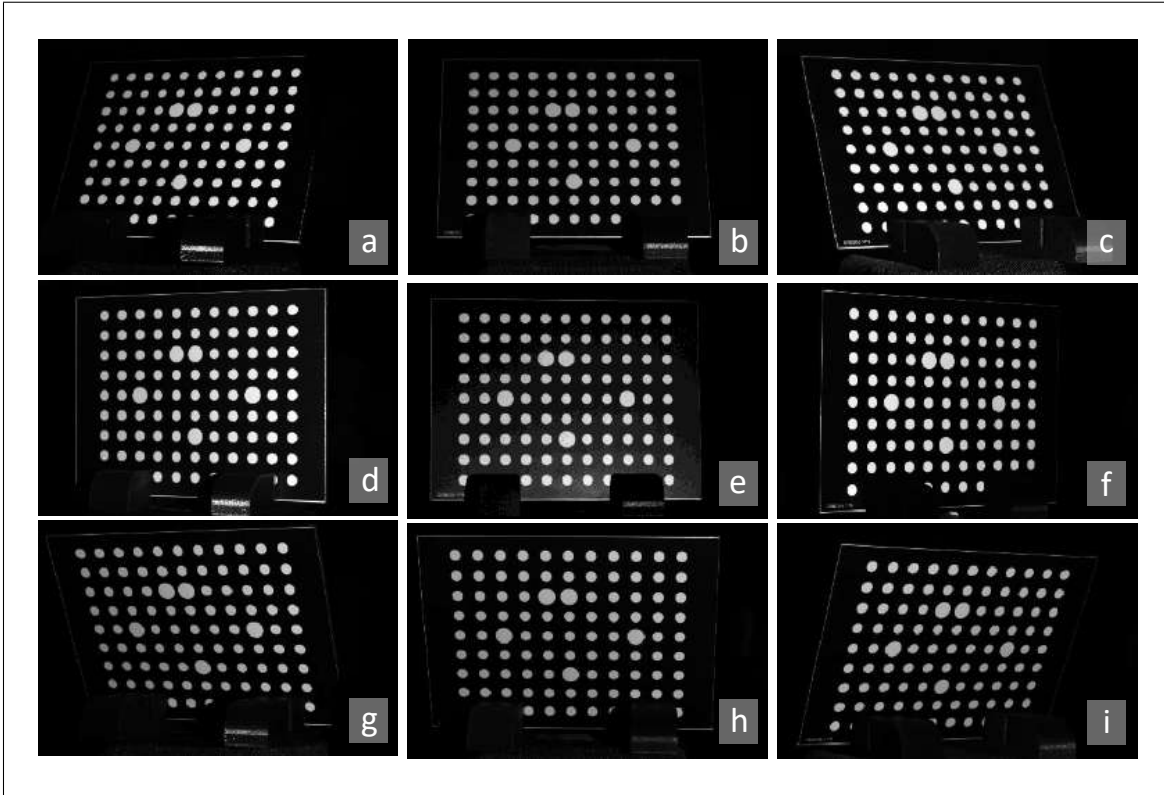


Figure S5: 9 images of the calibration board with different poses in the calibration of TFCMFPP system.

Both the camera and the projector were calibrated using the MATLAB Camera Calibration Toolbox [12], and the estimated parameters were further refined through bundle adjustment [13]. This optimization process effectively minimized reprojection residuals while compensating for minor measurement noise and manufacturing inaccuracies of the calibration board. The experimentally determined intrinsic parameters of the camera, determined experimentally, were: focal lengths [1850.40, 1850.61] px, principal point [319.82, 219.12] px, and distortion coefficients [-0.040, 0.150, 0.0001, 0.0002, 0.0002]. The corresponding parameters of the projector were: focal lengths [1880.57, 1880.71] px, principal point [380.25, 251.22] px, and distortion coefficients [-0.030, -0.170, 0.0002, -0.0003, 0.0000]. The final root-mean-square (RMS) reprojection errors were 0.05 px for the camera and 0.07 px for the projector, confirming sub-pixel calibration accuracy. The calibrated parameters provided a precise geometric foundation for pixel-level correspondence and accurate 3D reconstruction in the TFCMFPP system. As summarized in Table

S1, the calibration results confirm that the TFCMFPP setup achieves high geometric consistency between the camera and projector.

Table S1. Calibrated intrinsic, extrinsic, and distortion parameters of the TFCMFPP system.

Parameter	Camera	Projector
Principal point (px)	[319.82, 219.12]	[380.25, 251.22]
Focal length (px)	[1850.40, 1850.61]	[1880.57, 1880.71]
Skew coefficient	0.0000	0.0000
Distortion coefficients (k_1, k_2, p_1, p_2, k_3)	[-0.040, 0.150, 0.0001, 0.0002, 0.0002]	[-0.030, -0.170, 0.0002, -0.0003, 0.0000]
Rotation matrix	$\begin{bmatrix} -0.999 & 0.010 & 0.040 \\ 0.010 & 0.999 & 0.010 \\ -0.040 & 0.010 & -0.999 \end{bmatrix}$	$\begin{bmatrix} -0.990 & -0.040 & 0.150 \\ -0.030 & 0.998 & 0.050 \\ -0.140 & 0.050 & -0.990 \end{bmatrix}$
Translation vector (mm)	[-190.00, 25.00, 75.00]	[-50.00, -85.00, 1175.00]
RMS error (px)	0.05	0.07

3 Augmented 3D Reconstruction (A3DR)

Conventional fringe projection profilometry (FPP) systems are typically arranged on a common horizontal reference plane, i.e., the optical axes of the camera and the projector are nearly parallel to the same baseline. Under such configurations, the classical phase-to-height (or phase-to-depth) algorithms are only suitable for vertical fringes. However, in the proposed TFCMFPP system, the projected fringes exhibit different orientations and spatial frequencies across the RGB channels. To accurately convert the color-multiplexed phase information into spatial coordinates, we extend the geometric model of conventional FPP to accommodate multi-orientation and multi-frequency color patterns.

In the 3D imaging geometry, the relationship between the 3D world coordinates of a measured point (x^w, y^w, z^w) and the corresponding camera pixel coordinates (x^c, y^c) can be expressed as:

$$s \begin{bmatrix} x^c \\ y^c \\ 1 \end{bmatrix} = \mathbf{K}^c \begin{bmatrix} \mathbf{R}^c & \mathbf{t}^c \end{bmatrix} \begin{bmatrix} x^w \\ y^w \\ z^w \\ 1 \end{bmatrix} = \mathbf{P}^c \begin{bmatrix} x^w \\ y^w \\ z^w \\ 1 \end{bmatrix}, \quad (\text{S1})$$

where the superscript c denotes the camera. \mathbf{K}^c is the 3×3 intrinsic matrix, and $[\mathbf{R}^c, \mathbf{t}^c]$ represents the 3×4 extrinsic matrix (rotation and translation) transforming the world coordinate system to the camera coordinate system. \mathbf{P}^c is the 3×4 projection matrix of the camera. The scalar s is a normalization factor related to the image depth.

Similarly, modeling the projector as an inverse camera [9], the mapping between the 3D world coordinates (x^w, y^w, z^w) and the projector pixel coordinates (x^p, y^p) is written as:

$$s \begin{bmatrix} x^p \\ y^p \\ 1 \end{bmatrix} = \mathbf{K}^p \begin{bmatrix} \mathbf{R}^p & \mathbf{t}^p \end{bmatrix} \begin{bmatrix} x^w \\ y^w \\ z^w \\ 1 \end{bmatrix} = \mathbf{P}^p \begin{bmatrix} x^w \\ y^w \\ z^w \\ 1 \end{bmatrix}, \quad (\text{S2})$$

where \mathbf{K}^p and $[\mathbf{R}^p, \mathbf{t}^p]$ denote the intrinsic and extrinsic matrices of the projector, respectively, and \mathbf{P}^p is its perspective projection matrix. The superscript p corresponds to the projector coordinate system.

In the TFCMFPP system, each color channel (R, G, B) encodes fringe patterns of distinct spatial frequencies and orientations. The phase $\Phi_m^k(x^c, y^c)$ ($k \in \{R, G, B\}$) of the m -th frequency component corresponding to each camera pixel (x^c, y^c) can be described as:

$$\Phi_m^k(x^c, y^c) = 2\pi (\alpha^k x^p + \beta^k y^p), \quad (\text{S3})$$

where x^p, y^p are the corresponding projector pixel coordinates, and α^k, β^k are direction-dependent coefficients determined by the fringe's orientation θ_k and spatial frequencies $(\lambda_x^k, \lambda_y^k)$:

$$\alpha^k = \frac{\cos \theta_k}{\lambda_x^k}, \quad \beta^k = \frac{\sin \theta_k}{\lambda_y^k}. \quad (\text{S4})$$

When the camera and projector are calibrated within a unified world coordinate system, the 3D coordinates (x^w, y^w, z^w) corresponding to each camera pixel (x^c, y^c) can be determined by combining Eqs. (S1) and (S2), yielding:

$$\begin{bmatrix} x^w \\ y^w \\ z^w \end{bmatrix} = \left(\begin{bmatrix} p_{11}^c - p_{31}^c x^c & p_{12}^c - p_{32}^c x^c & p_{13}^c - p_{33}^c x^c \\ p_{21}^c - p_{31}^c y^c & p_{22}^c - p_{32}^c y^c & p_{23}^c - p_{33}^c y^c \\ p_{11}^p \alpha^k + p_{21}^p \beta^k - p_{31}^p (\alpha^k x^p + \beta^k y^p) & p_{12}^p \alpha^k + p_{22}^p \beta^k - p_{32}^p (\alpha^k x^p + \beta^k y^p) & p_{13}^p \alpha^k + p_{23}^p \beta^k - p_{33}^p (\alpha^k x^p + \beta^k y^p) \end{bmatrix} \right)^{-1} \quad (\text{S5})$$

$$\times \begin{bmatrix} p_{14}^c - p_{34}^c x^c \\ p_{24}^c - p_{34}^c y^c \\ p_{14}^p \alpha^k + p_{24}^p \beta^k - p_{34}^p (\alpha^k x^p + \beta^k y^p) \end{bmatrix}.$$

In this equation, the parameters α^k and β^k are known from the design of the fringe patterns in each color channel. The quantities $(\alpha^k x^p + \beta^k y^p)$ can be derived from the absolute phase Φ_m^k by $\Phi_m^k/2\pi$. Therefore, a one-to-one mapping can be established from each camera pixel (x^c, y^c) to the 3D world coordinate (x^w, y^w, z^w) . This mapping constitutes the foundation for accurate 3D reconstruction in the TFCMFPP system, allowing precise phase-to-coordinate conversion for fringes of multiple orientations and frequencies multiplexed across color channels.

4 Virtual System Setup and Digital Twin Implementation

Constructing a large-scale real-world dataset for dynamic high-speed scenes is inherently labor-intensive and time-consuming. To overcome the limitations of dataset scale and alleviate the acquisition bottleneck, we developed a Digital Twin strategy leveraging the Blender software suite (Fig. S6a). By controlling Blender via its Python API, we constructed a physics-based virtual imaging system that rigorously mirrors the optical parameters of our physical TFCMFPP setup. This simulation framework serves two critical purposes: it provides a massive, diverse, and error-free dataset for learning the fundamental priors of color-multiplexed fringe demodulation, and it establishes a transfer learning pipeline to bridge the domain gap between simulation and reality.

Scene Content

To ensure the network learns robust geometric features, we leveraged the Thingi10K dataset as the source of 3D object models. This massive repository comprises 10,000 varying geometries, ranging from smooth shapes (e.g., anatomical scans, sculptures) to mechanical parts with high-frequency spatial details (e.g., CAD parts, gears). Furthermore, to simulate the varying reflectivity of real-world objects, we randomized the color (Base Color) and surface roughness (Roughness) for each simulation instance. In the optical model (Eq. S6), these material variations directly result in a certain range of background intensities $A(x, y)$ and fringe modulation $B(x, y)$, allowing the network to learn to demodulate phase information reliably even under conditions of low contrast or varying surface reflectivity.

Motion Simulation

In the physical world, creating a precise ground truth for a moving object is difficult. In the virtual environment, however, we can generate perfect labels. We utilized the Blender Python API to programmatically control object movement during the virtual exposure window. For each simulation instance, we defined the object's motion trajectory by setting start and end keyframes. We simulated three types of motion patterns to cover diverse dynamic scenarios:

- **Linear Translation:** Randomizing the object's position coordinates (x, y, z) between frames to simulate uniform linear motion at various speeds.
- **Rotation:** Randomizing the Euler angles to simulate rotational motion around arbitrary axes, mimicking the rotor/fan experiments.
- **Composite Motion:** We used Bezier curves to create non-linear paths, by combining this curved movement with rotation, we simulated complex and irregular motion.

Noise Formulation

To bridge the simulation-to-reality gap, we implemented a unified physics-driven model combined with a domain randomization strategy. The intensity distribution of the generated fringe pattern I_m^p is formulated to rigorously align with the imaging chain:

$$I_m^p(x, y) = A(x, y) + B(x, y) \cos \left(\frac{2\pi x \cos \theta_m}{\lambda_x^c} + \frac{2\pi y \sin \theta_m}{\lambda_y^c} \right) + \delta \quad (\text{S6})$$

where $A(x, y)$ and $B(x, y)$ represent the background intensity and modulation, which vary according to the surface material properties defined in Scene Content. The spatial parameters $\lambda_{x,y}^c$ and θ_m are set to identical values as those in the real experiment.

Crucially, to ensure the network is robust to varying signal-to-noise ratios (SNR) encountered in real-world measurements, we modeled the term δ using a domain randomization approach, where the noise intensity σ is randomized for each training sample. Specifically, based on the noise characteristics calibrated from our CMOS sensor, the standard deviation is randomly sampled within the range of $[0, 2.51]$.

This strategy forces the network to learn robust phase demodulation features that are resilient to diverse noise conditions.

Transfer Learning Strategy

Based on the constructed virtual and real-world datasets, we implemented a Transfer Learning strategy, as shown in Fig. S6b, to maximize network performance while minimizing real-world data acquisition costs. The training process progresses from simulation to reality: first, the network is pre-trained from scratch using the large-scale virtual dataset, allowing it to learn the fundamental physics of triple-frequency color de-multiplexing, spectral decoupling, and geometric reasoning in a controlled, noise-defined environment. This establishes a robust parametric initialization. Subsequently, these pre-trained weights are transferred and fine-tuned using the real-world dataset. This second stage is critical for bridging the domain gap by adapting the network to residual unmodeled factors, effectively combining the perfect physical priors from the digital twin with the realistic domain characteristics of the physical system.

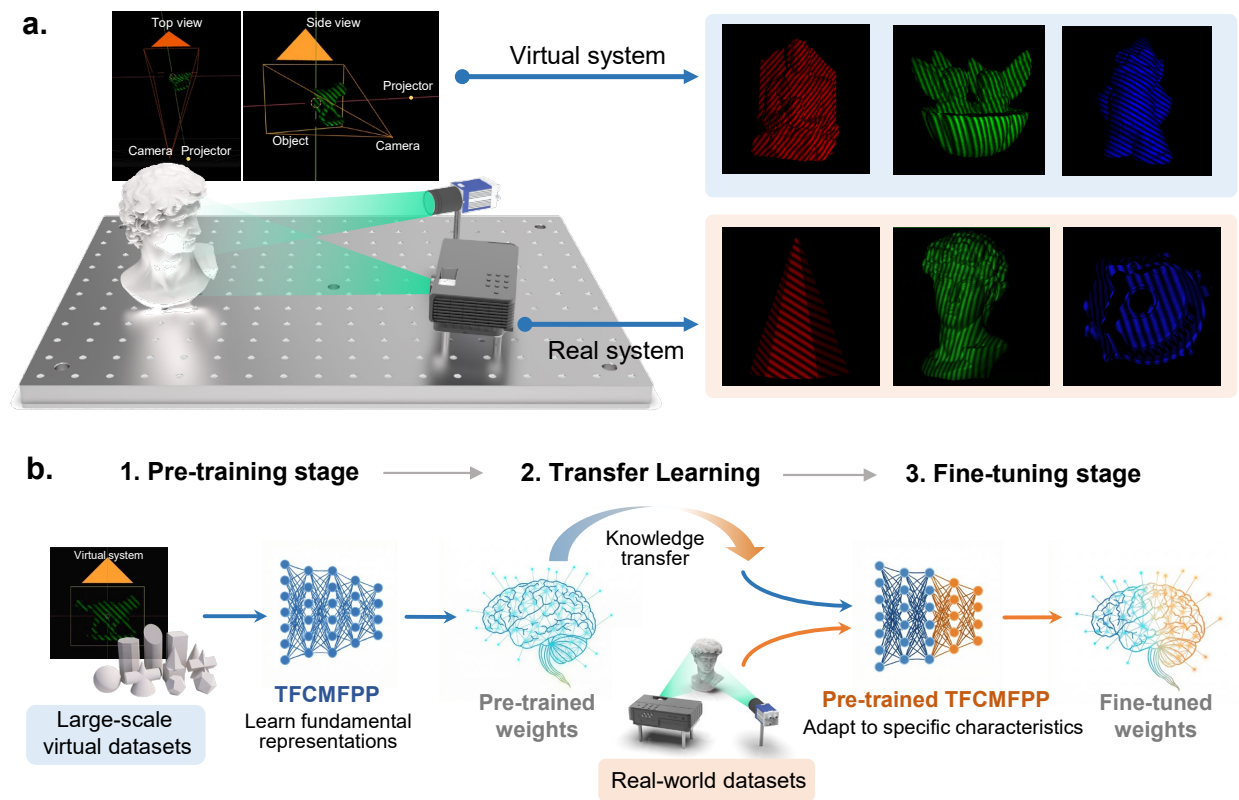


Figure S6: Digital twin framework and transfer learning strategy. **a**, The physics-based virtual environment constructed via Blender and the physical experimental setup. The digital twin accurately mirrors real-system parameters to generate massive synthetic training data. **b**, The transfer learning workflow. The network undergoes pre-training on virtual datasets to learn fundamental physical priors, followed by fine-tuning on real-world data to bridge the simulation-to-reality domain gap.

5 Data Acquisition and Ground-Truth Generation

In this section, we detail the acquisition procedure for the training data. The process is structurally divided into two phases: first, the rigorous design of three specific projection protocols (Sequences A, B, and C); and second, the experimental acquisition and computational generation of the corresponding input data and ground-truth labels.

Projection Sequences for Dataset Construction

The construction of a high-fidelity dataset is a complex engineering task. To systematically acquire the specific input data and corresponding ground-truth labels required for each module of our network, we designed three distinct fringe projection sequences, as shown in Fig. S7.

- **Sequence A:** Sequence A represents the standard projection mode of the TFCMFPP method. The projector outputs a continuous stream of color-multiplexed fringe patterns $I_m^p(x^p, y^p)$, identical to the mathematical definition provided in the Methods section of the main text.
- **Sequence B:** Sequence B is designed to physically separate the fringe signal from background illumination and crosstalk. It is constructed by inserting a uniform frame between consecutive fringe patterns of Sequence A. The projected pattern sequence I_k^{pa} is defined as:

$$I_k^{pa}(x^p, y^p) = \begin{cases} I_m^p(x^p, y^p), & k = 2m - 1 \\ a, & k = 2m \end{cases} \quad (S7)$$

where m denotes the index of the original fringe pattern in Sequence A, and a represents the uniform DC intensity component (consistent with the definition in the main text).

- **Sequence C:** Sequence C is designed to retrieve absolute phase maps with the highest possible accuracy. We employ a Multi-Frequency Multi-Step Phase-Shifting (MF-MSPS) strategy. For a specific channel c , frequency group $g \in \{L, M, H\}$, the n -th phase-shifted pattern is:

$$I_{g,m,n}^{(c)}(x, y) = a^{(c)} + b^{(c)} \cos \left[2\pi f_g^{(c)}(x \cos \theta_m + y \sin \theta_m) + \frac{2\pi n}{N_g} \right] \quad (S8)$$

Crucially, to minimize nonlinear gamma errors and sensor noise, the phase-shifting steps are explicitly set to $N_L = 3$, $N_M = 3$, and $N_H = 12$ (for the high-frequency component).

Ground-Truth Generation

Based on the three designed protocols, we acquired the corresponding physical quantities used for network training. Crucially, to ensure dataset consistency, we employed a high-precision programmable motion platform to enforce identical motion trajectories for the target objects across all three sequences. Sequence A records the continuous motion integration, while Sequences B and C record the “frozen” states at discrete positions along the same trajectory.

Sequence A was captured in long-exposure mode while the object was undergoing the programmed motion. The resulting captured image serves directly as the raw input for the network, denoted as I_{LE} .

Sequence B was captured in Single-frame exposure mode with the object kept static at the corresponding positions of the motion trajectory. The purified fringe image I_m^d was mathematically obtained by performing a pixel-wise subtraction between the captured images: specifically, subtracting the captured image of the uniform pattern (background) from the captured image of the fringe pattern. This operation effectively removes the background texture and ambient light. Furthermore, by digitally superimposing the purified fringe components from all $3M$ projection moments, we obtained the purified color-multiplexed image I_{LE}^d .

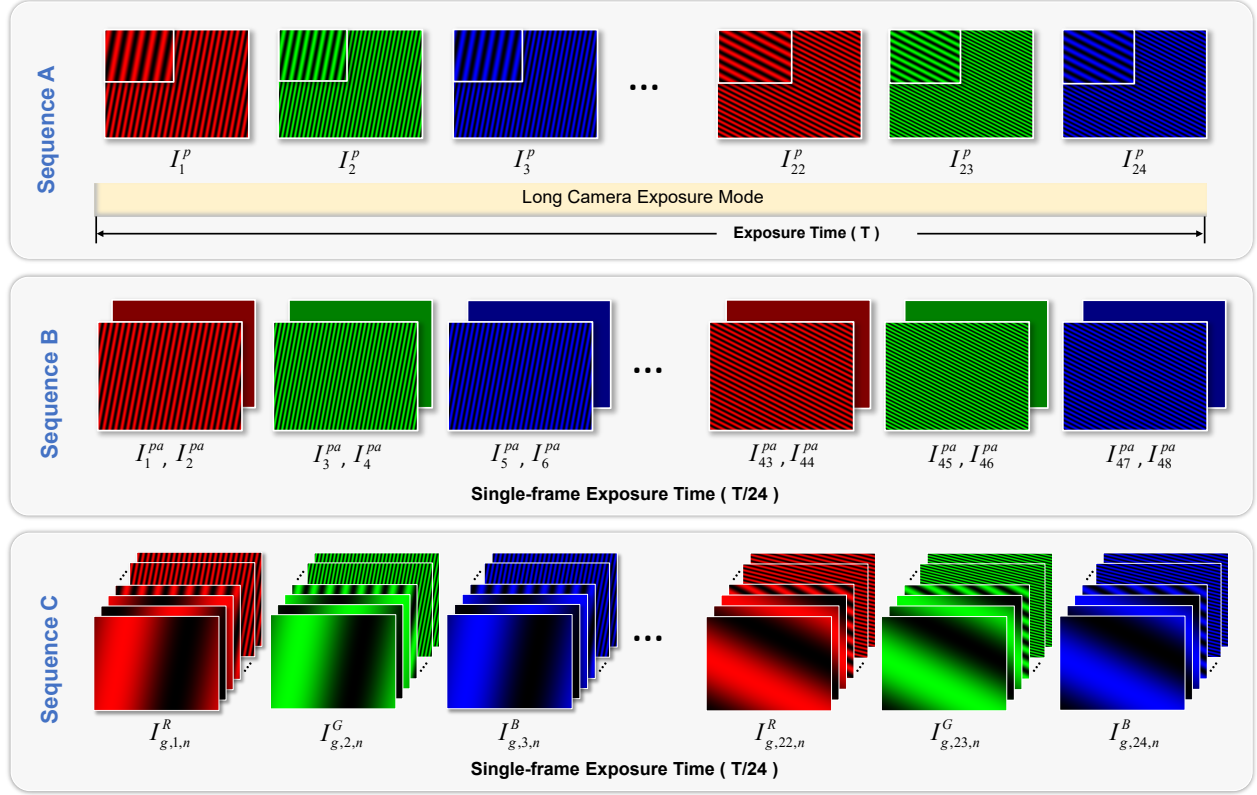


Figure S7: Schematic of projection sequences. Sequence A: Long-exposure mode performing temporal integration of directional fringes to generate the color-multiplexed image. Sequence B: Discrete single-frame capture mode for acquiring purified fringe labels I_m^d . Sequence C: Three-frequency multi-step phase-shifting sequence for generating high-precision ground-truth phase labels.

Sequence C was also captured in Single-frame exposure mode under the same static configuration. The high-precision absolute phase labels $\Phi_{H,m}^{(c)}$ were computationally derived from these images. Crucially, during the least-squares calculation, we explicitly extracted and stored the numerator ($M_{g,m}^{(c)}$) and denominator ($D_{g,m}^{(c)}$). These terms are defined as:

$$\begin{cases} M_{g,m}^{(c)}(x, y) = \sum_{n=0}^{N_g-1} I_{g,m,n}^{(c)}(x, y) \cdot \sin(2\pi n/N_g) \\ D_{g,m}^{(c)}(x, y) = \sum_{n=0}^{N_g-1} I_{g,m,n}^{(c)}(x, y) \cdot \cos(2\pi n/N_g) \end{cases} \quad (S9)$$

Using these components, the wrapped phase $\phi_{g,m}^{(c)}$ is computed as $\phi_{g,m}^{(c)} = -\tan^{-1}(M_{g,m}^{(c)}/D_{g,m}^{(c)})$. Subsequently, we recovered the absolute phase using temporal phase unwrapping.

$$\Phi_{H,m}^{(c)}(x, y) = \phi_{H,m}^{(c)}(x, y) + 2\pi \cdot \text{round} \left(\frac{\Phi_{L,m}^{(c)} \cdot (f_H^{(c)}/f_L^{(c)}) - \phi_{H,m}^{(c)}}{2\pi} \right) \quad (S10)$$

Through this process, we obtained Long-exposure color-multiplexed image (I_{LE}), purified intensity references (I_{LE}^d, I_m^d), phase numerator and denominator maps ($M_{H,m}^{(c)}, D_{H,m}^{(c)}$), and absolute phase maps ($\Phi_{H,m}^{(c)}$).

6 Design Principles for Angular Multiplexing Parameters

In this section, we provide the theoretical derivation for the selection of fundamental angular interval parameter (θ_0). The design objective is to maximize the information throughput while strictly satisfying the Nyquist sampling theorem and ensuring spectral separability to prevent aliasing artifacts.

In the proposed TFCMFPP framework, the projection system generates fringe patterns along M distinct spatial orientations. While color multiplexing separates signals into RGB channels, multiple directional components must be strategically arranged within the frequency domain to avoid interference. Let n denote the index of the projection angle ($n = 1, 2, \dots, M$). The orientation of the n -th fringe pattern, denoted as θ_n , is mathematically defined by the scalar parameter θ_0 as follows:

$$\theta_n = (-1)^{n+1} \left(\frac{n}{2} - \frac{(-1)^n + 1}{4} \right) \theta_0 \quad (\text{S11})$$

In the Fourier domain, the spatial spectrum of the n -th directional component exhibits conjugate spectral peaks located at coordinates (u_n, v_n) . These coordinates are determined by the carrier frequency f_c and the specific orientation θ_n :

$$\begin{cases} u_n = f_c \cos \theta_n \\ v_n = f_c \sin \theta_n \end{cases} \quad (\text{S12})$$

• Local Non-overlapping Condition (Lower Bound)

To ensure accurate phase retrieval, the spectral component of each directional must be isolated without spectrum overlapping. We define the Effective Spectral Bandwidth (B_w) of the object, which represents the radius of the spectral distribution (lobes) around the carrier frequency. Physically, B_w is proportional to the maximum gradient of the object's surface height variation.

To prevent spectrum overlapping, the Euclidean distance between spectral peaks of adjacent directional components must exceed the sum of their bandwidths. Using the spectral coordinates defined in Eq. (S12), the exact Euclidean distance d_{adj} between adjacent peaks (with angular separation θ_0) is calculated as:

$$d_{adj} = \sqrt{(u_{n+1} - u_n)^2 + (v_{n+1} - v_n)^2} = 2f_c \sin \left(\frac{\theta_0}{2} \right) \quad (\text{S13})$$

By applying the small-angle approximation ($\sin x \approx x$ for small x), Eq. (S13) simplifies to the arc length:

$$d_{adj} \approx 2f_c \cdot \frac{\theta_0}{2} = f_c \cdot \theta_0 \quad (\text{S14})$$

Consequently, to prevent spectral overlap, this distance must be greater than the full spectral width ($2B_w$). This establishes the Lower Bound condition for the angular parameter θ_0 :

$$f_c \cdot \theta_0 > 2B_w \implies \theta_0 > \frac{2B_w}{f_c} \quad (\text{S15})$$

This inequality implies that for a given carrier frequency f_c , the angular interval θ_0 must be sufficiently large to accommodate the spectral broadening caused by the object's complex surface details.

• Global Non-overlapping Condition (Upper Bound)

The total available angular space in the frequency domain for unique orientation encoding is π (180 degrees). To multiplex M distinct directional channels, the cumulative angular coverage must mathematically satisfy:

$$\theta_0 < \frac{180^\circ}{M} \quad (\text{S16})$$

While Eq. (S16) establishes the theoretical upper limit, practical parameter selection necessitates a more stringent constraint to ensure system robustness. In standard fringe projection profilometry systems (typically employing a horizontal baseline), the vertical frequency axis (corresponding to horizontal

fringes at 90°) constitutes a low-sensitivity zone for depth retrieval due to the triangulation geometry. Therefore, the angular interval θ_0 is designed to be slightly smaller than the theoretical upper bound. This strategy ensures that the entire set of M directional spectra is compactly arranged while explicitly avoiding the singular vertical direction, thereby preventing the allocation of information to an invalid reconstruction zone.

References

- [1] Fisher Yu and Vladlen Koltun. Multi-scale context aggregation by dilated convolutions. In *International Conference on Learning Representations (ICLR)*, 2016.
- [2] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations (ICLR)*, 2015.
- [3] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016.
- [4] Noam Shazeer et al. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations (ICLR)*, 2017.
- [5] Yoshua Bengio et al. Estimating or propagating gradients through stochastic neurons for conditional computation. *arXiv preprint arXiv:1308.3432*, 2013.
- [6] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-excitation networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 7132–7141, 2018.
- [7] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.
- [8] Wei Yin, Yuxuan Che, Xinsheng Li, Mingyu Li, Yan Hu, Shijie Feng, Edmund Y Lam, Qian Chen, and Chao Zuo. Physics-informed deep learning for fringe pattern analysis. *Opto-Electronic Advances*, 7(1):230034–1, 2024.
- [9] Zhengyou Zhang. A flexible new technique for camera calibration. *IEEE Transactions on pattern analysis and machine intelligence*, 22(11):1330–1334, 2000.
- [10] Xiaojun Su, Song Zhang, and Philipe Huang. Accuracy enhancement for three-dimensional shape measurement using projector calibration. *Optics and lasers in engineering*, 48(2):213–223, 2010.
- [11] Xiao Xu, Chao Zuo, Min Zhang, and et al. Fast and accurate phase-shifting profilometry using a one-dimensional windowed fourier transform. *Optics and Lasers in Engineering*, 126:105899, 2020.
- [12] Bing Pan, Lei Huang, and et al. Phase error compensation for a three-dimensional shape measurement system based on the phase-shifting method. *Optics and lasers in engineering*, 47(7-8):865–871, 2009.
- [13] Jean-Yves Bouguet. Matlab camera calibration toolbox.
https://github.com/kyamagu/mexopencv/blob/master/samples/calibration_example.cpp. Accessed : 2025 – 09 – 29.