

# VIRTUS-FPP: Virtual Sensor Modeling for Fringe Projection Profilometry in NVIDIA Isaac Sim

Adam Haroon<sup>1\$</sup>, Anush Lakshman<sup>1\$</sup>, Badrinath Balasubramaniam<sup>2</sup>, Beiwen Li<sup>2\*</sup>

**Abstract**—Fringe projection profilometry (FPP) has been established as a high-accuracy 3D reconstruction method capable of achieving sub-pixel accuracy. However, this technique faces significant constraints due to complex calibration requirements, bulky system footprint, and sensitivity to environmental conditions. To address these limitations, we present VIRTUS-FPP, the first comprehensive physics-based virtual sensor modeling framework for FPP built in NVIDIA Isaac Sim. By leveraging the physics-based rendering and programmable sensing capabilities of simulation, our framework enables end-to-end modeling from calibration to reconstruction with full mathematical fidelity to the underlying principles of structured light. We conduct comprehensive virtual calibration and validate our system’s reconstruction accuracy through quantitative comparison against ground truth geometry. Additionally, we demonstrate the ability to model the virtual system as a digital twin by replicating a physical FPP system in simulation and validating correspondence between virtual and real-world measurements. Experimental results demonstrate that VIRTUS-FPP accurately models optical phenomena critical to FPP and achieves results comparable to real-world systems while offering unprecedented flexibility for system configuration, sensor prototyping, and environmental control. This framework significantly accelerates the development of real-world FPP systems by enabling rapid virtual prototyping before physical implementation.

**Index Terms**—Fringe projection profilometry, structured light, 3D reconstruction, physics-based simulation, NVIDIA Isaac Sim, optical metrology, synthetic data generation

## I. INTRODUCTION

Fringe Projection Profilometry (FPP) has emerged as a powerful non-contact 3D imaging technology based on the principles of structured light, offering high-resolution surface measurements with sub-millimeter and sub-pixel accuracy [1], [2]. Operating on triangulation principles similar to stereo vision systems, FPP employs a camera-projector pair where the projector displays sinusoidal fringe patterns onto object surfaces, which become distorted based on the surface morphology. These distortions encode phase information that are captured by the camera and subsequently used for precise 3D reconstruction [3], [4]. The entire process of FPP follows a systematic sequence of steps: camera calibration, fringe pattern generation, fringe image capture of calibration target, stereo

Manuscript received xx, 2025; revised xx, 2025. (*Corresponding author: Beiwen Li*)

<sup>1</sup>Adam Haroon and Anush Lakshman are with the Department of Mechanical Engineering, Iowa State University, Ames, IA 50012, USA anushlak@iastate.edu; aharoon@iastate.edu

<sup>2</sup>Badrinath Balasubramaniam and Beiwen Li are with the College of Engineering, University of Georgia, Athens, GA 30602, USA bb2@uga.edu; beiwen.li@uga.edu

<sup>\$</sup>These authors contributed equally.

calibration of camera and projector, fringe image capture of target object, phase analysis (wrapping and unwrapping), and finally 3D reconstruction [5]–[7].

The high-resolution measurement capability and non-invasive nature of FPP make it increasingly suitable for deployment in various critical applications including precision manufacturing monitoring and real-time industrial quality control [8]–[10], precise robotic manipulation and bin picking [11]–[13], corrosion analysis [14], [15], and advanced biomedical imaging applications [16], [17]. The versatility of FPP extends to challenging measurement scenarios including microscale surface characterization [18], large-scale industrial inspection [19], and dynamic scene reconstruction [20], making it an indispensable tool in modern metrology and computer vision applications.

Despite the demonstrated accuracy and versatility of FPP systems, several fundamental practical limitations continue to hinder their widespread deployment. Calibration, although absolutely essential for obtaining accurate phase-to-3D mapping relationships, remains an exceptionally time-consuming process that significantly impacts operational efficiency [21]. Contemporary FPP systems predominantly rely on circular board patterns due to their superior robustness and noise characteristics compared to conventional checkerboard patterns, yet even with these improvements, the calibration procedure still involves capturing numerous calibration target poses with multiple phase-shifted fringe patterns to establish reliable stereo correspondence and phase-to-depth conversion matrices [22], [23]. This procedure requires specialized expertise in optical metrology and consumes significant time resources, often taking several hours to complete for a single system configuration. The situation is further complicated by the fact that even minor system disturbances, such as mechanical vibrations, thermal drift, or slight component repositioning, often necessitate complete recalibration, significantly impacting operational efficiency and limiting the practical deployment of FPP systems in dynamic or uncontrolled environments [21], [24].

Environmental factors introduce additional layers of complexity that further complicate FPP deployment in real-world scenarios. Variations in ambient lighting conditions can severely interfere with fringe visibility and degrade phase accuracy, particularly problematic in uncontrolled industrial environments where lighting conditions cannot be precisely managed [25], [26]. The challenge becomes even more pronounced when dealing with complex material properties including varying surface reflectivity, transparency, translucency, and subsurface scattering effects that introduce sophis-

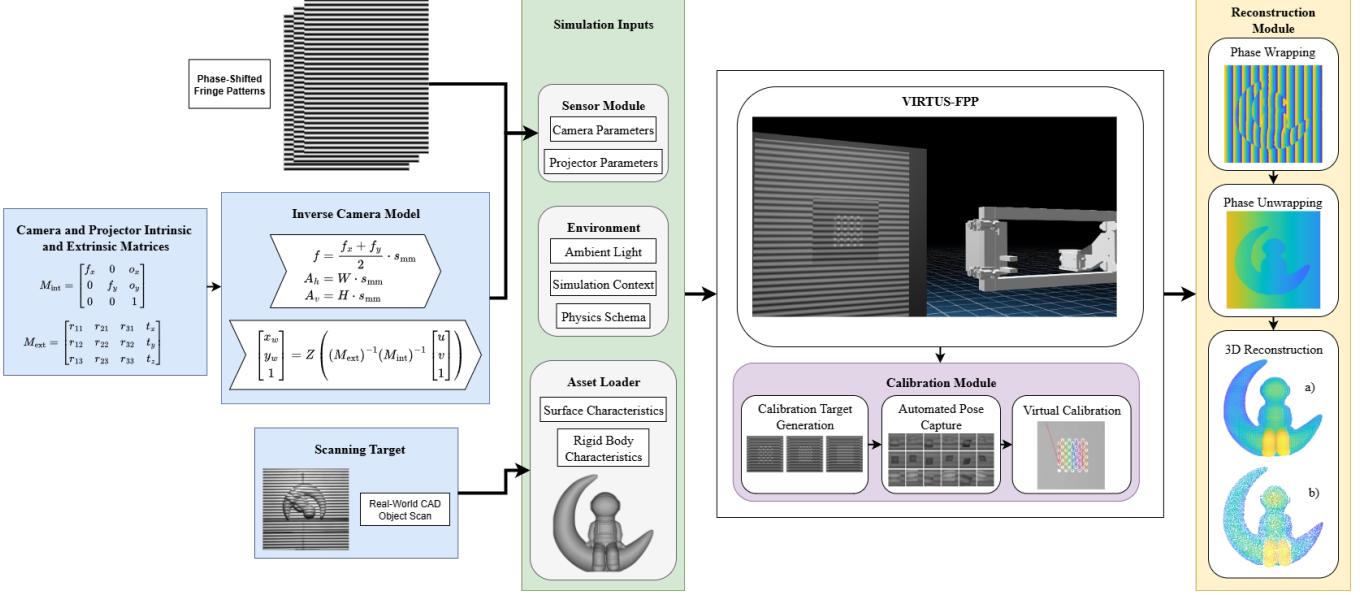


Fig. 1: VIRTUS-FPP system architecture and processing pipeline. The framework integrates camera and projector intrinsic/extrinsic parameters, inverse camera modeling, and NVIDIA Isaac Sim’s physics-based photorealistic rendering to enable end-to-end virtual fringe projection profilometry. The pipeline encompasses fringe pattern generation, virtual calibration target positioning, automated pose capture, scanning target fringe image capture, phase wrapping/unwrapping, and virtual 3D reconstruction a) with ground truth validation b).

ticated optical phenomena such as specular reflections, inter-reflections, and phase distortions that fundamentally challenge traditional reconstruction algorithms [27]–[29]. Materials with high reflectivity can cause fringe pattern saturation and phase errors, while translucent materials introduce subsurface scattering that corrupts phase measurements, and specular surfaces create unwanted highlights that interfere with accurate phase extraction [30]–[32]. Understanding and systematically mitigating these effects requires extensive empirical testing across diverse environmental conditions and material properties, a process that is inherently time-consuming, expensive, and often impractical to achieve comprehensively with physical hardware setups [27]. These challenges are magnified when developing machine learning methods, where large, diverse, and labeled datasets are essential for accurate model training and performance [33]–[35].

To address these fundamental limitations of physical FPP systems, virtual sensor modeling and digital twin methodologies have emerged as powerful alternatives [36], [37]. These simulation environments enable systematic evaluation of FPP performance under controlled, repeatable conditions such as varying ambient lighting, occlusion, surface materials, sensor configurations, and object geometries. By accelerating experimentation and reducing dependence on hardware, virtual systems support rapid development and testing of calibration pipelines and reconstruction algorithms. Moreover, they can generate large-scale synthetic datasets with known ground truths critical for training, testing, and benchmarking machine learning models, effectively bridging the gap between traditional optical metrology and modern data-driven approaches [34], [38], [39].

The integration of synthetic data generation with deep

learning has proven particularly valuable for optical metrology applications, where traditional physics-based methods often struggle with complex material properties and environmental variations [40]. Recent advances in neural network architectures specifically designed for fringe pattern analysis have demonstrated significant improvements in phase unwrapping [41], fringe pattern denoising [42], and surface reconstruction quality when trained on large-scale synthetic datasets [43]. However, the effectiveness of these data-driven approaches critically depends on the fidelity of the underlying simulation framework and its ability to accurately model the complex light transport phenomena that govern structured light measurements [44], making physics-based virtual sensor modeling essential for advancing machine learning applications in optical metrology. Further, domain randomization techniques [45] can be applied within these virtual environments to systematically vary environmental parameters, improving algorithm robustness and facilitating successful sim-to-real transfer for deployment in real-world scenarios.

This emerging research direction has primarily focused on developing digital twins of FPP systems, typically by transferring real-world calibration matrices from physical camera-projector setups into virtual environments. Zheng et al. [46] pioneered this approach by developing a graphics-based digital twin capable of automatically generating 7200 fringe images with 800 corresponding 3D depth maps in 1.5 hours, significantly accelerating data acquisition compared to physical systems. Ueda et al. [47] utilized Unity game engine’s capabilities to develop an FPP simulator with a graphical user interface for adjusting camera parameters such as aperture, exposure time, and depth-of-field, allowing for verification of measurement conditions that would be challenging to set up experimentally.

Zhang et al. [48] employed ray-tracing techniques to simulate fringe pattern projection and imaging processes, allowing for detailed modeling of optical phenomena including perspective distortion and lens effects. More recently, Tang et al. [49] proposed a projection pattern pre-correction method based on projection error decoupling that effectively addresses the simultaneous occurrence and mutual influence of geometric distortion and grayscale inconsistency in FPP by separating the calibration process into two independent procedures.

While these digital twin systems aid in FPP system development and improvement, they exhibit fundamental limitations that compromise their effectiveness as true digital twins. Current approaches primarily consider only extrinsic parameters and empirically determine intrinsic parameters, particularly for the projector, which prevents accurate geometric modeling. Moreover, these methods fundamentally depend on pre-calibrated physical systems, requiring real hardware to model the virtual FPP system. By transferring real-world calibration matrices into virtual environments, any physical inaccuracies and measurement errors are inherently propagated into the simulation, reducing reliability for precision measurement applications. To address these limitations, our work presents an end-to-end virtual modeling framework that eliminates dependence on physical systems by establishing projector intrinsics through theoretical formulation rather than empirical transfer, enabling the creation of truly independent and accurate digital twins.

These approaches predominantly utilize Blender [46], Unity [47], and MATLAB [49], each of which presents specific limitations for advanced FPP research. Blender, a widely adopted platform in computer graphics applications, uses the Cycles path tracer for physically based rendering, enabling subsurface scattering, specular interreflections, and global illumination [50]. However, Cycles lacks support for spectral rendering, wave interference, and coherent light modeling, which are critical for simulating fringe pattern behavior with high optical fidelity [50]. Unity's High-Definition Render Pipeline (HDRP) similarly prioritizes high visual fidelity over physically accurate light transport, limiting its ability to faithfully reproduce complex light-material interactions critical for accurate FPP simulation [51]. While both Blender and Unity can be extended via scripting, neither provides native support for optimized large-scale parallel simulation across randomized environments, with their synthetic data generation pipelines remaining relatively limited in scope. MATLAB presents even more restrictive constraints, offering only simplified geometric ray tracing with minimal physically grounded material modeling capabilities. Although MATLAB supports parallel computing for numerical simulations, it lacks a native photorealistic rendering engine capable of large-scale synthetic dataset generation at the fidelity required for FPP. As a result, all three platforms lack tightly integrated, high-throughput pipelines needed for testing, developing, and validating data-driven FPP algorithms at scale.

In order to overcome the limitations of prior systems, we propose VIRTUS-FPP, the first comprehensive physics-based simulation framework for FPP that models the entire imaging pipeline, from calibration to 3D reconstruction, with full mathematical fidelity to the structured light principles underlying the technique.

We build VIRTUS-FPP on NVIDIA Isaac Sim, a high-fidelity robotics simulator built on the Omniverse platform. NVIDIA Isaac Sim provides native integration of PhysX for physically accurate interactions, Universal Scene Description (USD) for scalable 3D environment composition, and OptiX for RTX-accelerated ray tracing, enabling realistic simulation of complex optical behaviors central to fringe pattern deformation and phase reconstruction [52]. In contrast to Blender and MATLAB, NVIDIA Isaac Sim supports photorealistic rendering pipelines, advanced material modeling via the Material Definition Language (MDL) framework, and high-speed synthetic data generation. Moreover, it allows for parallelized simulation of multiple environments using GPU acceleration, dramatically increasing synthetic dataset generation throughput and experimentation efficiency. These capabilities make VIRTUS-FPP uniquely suited for generating physically accurate fringe images under diverse and controlled conditions, enabling rapid prototyping, rigorous development, and systematic evaluation of FPP algorithms.

Our primary contributions are as follows:

- 1) We develop the first fully end-to-end virtual FPP system with complete virtual calibration, enabling accurate modeling without dependence on pre-calibrated physical systems.
- 2) The framework provides physics-accurate simulation of complex optical effects through RTX-accelerated ray tracing and supports systematic domain randomization for robust algorithm development.
- 3) Our system achieves 3 FPS data acquisition, more than twice the speed of current state-of-the-art approaches [46].
- 4) We mathematically model the projector using the inverse camera model, enabling accurate virtual projector behavior with geometric and photometric fidelity.
- 5) We construct and validate a digital twin of a calibrated real-world FPP system, demonstrating high fidelity in simulation-to-reality transfer through rigorous experimental comparison.

The remainder of this paper is structured as follows: Section II establishes the theoretical foundations of FPP and introduces NVIDIA Isaac Sim's capabilities for physics-based simulation. Section III details our VIRTUS-FPP framework, describing the system architecture, implementation details, and virtual calibration process. Section V presents comprehensive experimental validation demonstrating the accuracy and flexibility of our approach through quantitative comparisons with ground truth data, systematic evaluation under varying environmental conditions, and digital twin validation against a physical FPP system. We then analyze the implications and limitations of our framework in Section VI, followed by conclusions and future research directions in Section VII.

## II. METHODOLOGY

### A. Fringe Projection Profilometry (FPP)

The fundamental working principle of a fringe projection system is triangulation, similar to stereo vision systems, but

instead of two cameras, it uses one camera and one projector. The projector serves as an inverse imaging device that projects structured fringe patterns onto the object surface, while the camera captures the resulting deformations. This triangulation mechanism is illustrated in Figure 2. The process begins by loading predefined fringe patterns into the projector and synchronizing it temporally with the camera to ensure precise image capture. The captured fringe images encode the object's surface geometry, which is then decoded through three key steps: phase wrapping, phase unwrapping, and finally, calibration and 3D reconstruction.

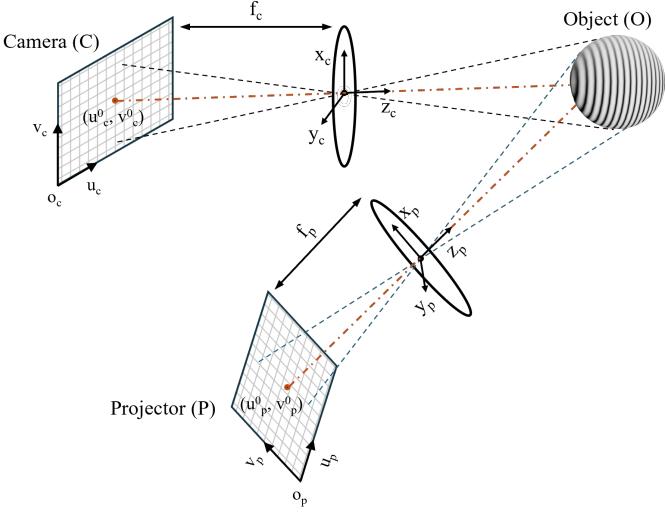


Fig. 2: Geometric triangulation principles underlying fringe projection profilometry, showing the camera-projector configuration and structured light projection onto a target object.

- **Phase Wrapping:** A wrapped phase map is computed from the captured fringe images using the  $N$ -step phase-shifting method. The intensity distribution of the  $n^{\text{th}}$  fringe image,  $I_n(x, y)$ , is given by:

$$I_n(x, y) = I'(x, y) + I''(x, y) \cos \left( \phi(x, y) + \frac{2\pi n}{N} \right), \quad (1)$$

where  $N$  is the total number of phase steps,  $n = 1, 2, \dots, N$ ,  $I'(x, y)$  is the background (average) intensity,  $I''(x, y)$  is the modulation amplitude, and  $\phi(x, y)$  is the phase to be extracted. In our setup, we use an 18-step phase-shifting technique ( $N = 18$ ) for both imaging systems, as it enables high-resolution 3D reconstruction with sub-millimeter accuracy [1]. Each of the 18 fringe images corresponds to a phase shift  $\delta_n = \frac{2\pi n}{N}$ . Solving Equation 1 for  $\phi(x, y)$  yields the wrapped phase map:

$$\phi(x, y) = -\tan^{-1} \left( \frac{\sum_{n=1}^N I_n(x, y) \sin \delta_n}{\sum_{n=1}^N I_n(x, y) \cos \delta_n} \right). \quad (2)$$

- **Phase Unwrapping:** The wrapped phase map  $\phi(x, y)$  obtained above contains  $2\pi$  discontinuities due to the periodic nature of the  $\tan^{-1}$  function. To resolve these

discontinuities, we perform temporal phase unwrapping, which is more robust than spatial methods [7]. Combined with Gray coding, this technique determines the integer number  $k(x, y)$  of  $2\pi$  phase cycles to be added at each pixel. The resulting unwrapped phase map  $\Phi(x, y)$  is given by:

$$\Phi(x, y) = \phi(x, y) + 2\pi k(x, y). \quad (3)$$

- **Calibration and 3D Reconstruction:** This crucial step involves converting the unwrapped phase map  $\Phi(x, y)$  into 3D coordinates. The camera and projector are modeled as pinhole devices and characterized by their respective transformation matrices, which combine intrinsic parameters ( $A$ ) with extrinsic parameters—rotation ( $R$ ) and translation ( $t$ ) [27]:

$$M^c = A^c[R^c | t^c], \quad M^p = A^p[R^p | t^p], \quad (4)$$

where  $A^c$  and  $A^p$  are the intrinsic matrices of the camera and projector, respectively. Using these transformations, a 3D point in world coordinates  $[x, y, z]^T$  can be related to its image coordinates  $(u^c, v^c)$  in the camera and  $(u^p, v^p)$  in the projector as:

$$s^c \begin{bmatrix} u^c \\ v^c \\ 1 \end{bmatrix} = M^c \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \quad s^p \begin{bmatrix} u^p \\ v^p \\ 1 \end{bmatrix} = M^p \begin{bmatrix} x \\ y \\ z \end{bmatrix}, \quad (5)$$

where  $s^c$  and  $s^p$  are the respective scaling factors. Once the camera matrix  $M^c$  and projector matrix  $M^p$  are calibrated, the 3D coordinates can be computed by solving:

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = \mathbf{A}^{-1} \mathbf{b}, \quad (6)$$

where:

$$\mathbf{A} = \begin{bmatrix} m_{11}^c - u^c m_{31}^c & m_{12}^c - u^c m_{32}^c & m_{13}^c - u^c m_{33}^c \\ m_{21}^c - v^c m_{31}^c & m_{22}^c - v^c m_{32}^c & m_{23}^c - v^c m_{33}^c \\ m_{11}^p - u^p m_{31}^p & m_{12}^p - u^p m_{32}^p & m_{13}^p - u^p m_{33}^p \end{bmatrix}; \quad (7)$$

$$\mathbf{b} = \begin{bmatrix} u^c m_{34}^c - m_{14}^c \\ v^c m_{34}^c - m_{24}^c \\ u^p m_{34}^p - m_{14}^p \end{bmatrix}, \quad (8)$$

and  $m_{ij}^c$ ,  $m_{ij}^p$  are elements of the camera and projector matrices  $M^c$  and  $M^p$ , respectively.

The aforementioned principles were applied to model and calibrate the virtual sensor in NVIDIA Isaac Sim. The principles and capabilities of Isaac Sim utilized for modelling FPP sensor are elaborated in Section II-B. The calibrated system was to obtain 3D reconstructions, whose fidelity was validated using RMSE which was obtained through Cloud-to-Mesh Distance (C2M Distance).

## B. NVIDIA Isaac Sim

NVIDIA Isaac Sim is a physics-based robotics simulation platform built on NVIDIA Omniverse, designed for developing, testing, and training AI-based robotic systems in photorealistic environments [53]. As a scalable, high-fidelity simulator, Isaac Sim leverages GPU acceleration to enable realistic sensor modeling and synthetic data generation critical for developing perception systems like our FPP implementation. The platform provides a comprehensive suite of tools that support both visual programming through the OmniGraph interface and programmatic control via Python APIs, offering flexibility for both rapid prototyping and advanced custom implementations [53].

The adoption of Isaac Sim for virtual sensor modeling has demonstrated significant potential in optical and structured light applications. Our previous work demonstrated Isaac Sim's capability for streamlined large-scale 3D reconstruction of complex warehouse environments by leveraging its synthetic data generation capabilities [54]. Recent advances in structured light simulation have shown that physically-based rendering environments like Isaac Sim significantly outperform traditional approaches, achieving superior sim2real transfer for robotic manipulation tasks through domain randomized high-fidelity synthetic data [13].

*1) Simulation Scene Development:* Scene construction in NVIDIA Isaac Sim is built on Universal Scene Description (USD), Pixar's open-source framework for describing, composing, and simulating complex 3D scenes [55]. USD provides a hierarchical, component-based architecture that enables us to define, manipulate, and render complex scenes with precise control over every element's properties. This capability is essential for accurately positioning the camera and projector in our FPP system and defining their relative geometric relationships.

NVIDIA Isaac Sim offers two primary workflows for scene development. The OmniGraph visual programming interface provides a node-based system suitable for rapid prototyping with pre-defined components. However, for our implementation, we utilize the Python-based extension system, which offers more granular control over scene elements and enables programmatic automation of the calibration process. This approach allows us to dynamically generate and configure the camera-projector system, control fringe pattern projection, and manipulate calibration targets with precise programmatic control.

The simulation is underpinned by NVIDIA PhysX, which provides physically accurate dynamic interactions between objects in the scene [56]. While our current implementation primarily utilizes static objects for calibration, PhysX support enables future extensions of our work to dynamic scenes and robotic integration. The platform's pre-defined assets include various camera models and light sources that serve as foundational components for our virtual sensor system.

*2) Rendering:* NVIDIA Isaac Sim's rendering capabilities are built on NVIDIA RTX technology with OptiX ray tracing, providing physically-based rendering essential for simulating the optical characteristics of FPP systems [57]. OptiX enables accurate modeling of light transport phenomena such

as reflection, refraction, and scattering—effects that directly impact the quality and accuracy of fringe pattern projection and subsequent phase analysis.

The efficacy of OptiX for advanced optical modeling has been demonstrated across various applications requiring accurate light transport simulation. In medical imaging, OptiX has been successfully applied to model X-ray and gamma radiation propagation in SPECT imaging systems, demonstrating significant computational speedups compared to traditional Monte Carlo simulations [58]. GPU-accelerated optical simulation studies have shown OptiX's capability to achieve real-time performance for complex optical systems design and analysis [59]. The foundational OptiX architecture provides a programmable ray tracing engine specifically designed for high-performance optical simulations [52].

The renderer supports key physically-based light transport simulations including:

- Path tracing for photorealistic global illumination
- Physically accurate material properties through the Material Definition Language (MDL)
- Multi-bounce light transport essential for modeling complex surface interactions
- Accurate shadow generation and ambient occlusion for realistic depth cues

These capabilities allow our virtual FPP system to simulate critical optical phenomena that affect real-world structured light systems, including surface reflectivity variations, ambient light interference, and specular highlights. By configuring rendering parameters such as ray depth, sample count, and denoising filters, we can balance physical accuracy with computational efficiency for our specific application needs.

*3) Synthetic Data Generation:* NVIDIA Isaac Sim's synthetic data generation pipeline is built on NVIDIA Replicator, a framework for producing annotated training data at scale [60]. This capability is crucial for our work as it enables high-throughput acquisition of calibration images and fringe pattern captures under controlled, repeatable conditions.

The platform's advanced sensor simulation capabilities have been validated across many specialized sensing applications. Event-based camera simulation for robotic slip detection, inertial measurement units (IMUs) for drone localization, and recent work in underwater imaging sonar development has demonstrated Isaac Sim's ability to accurately model complex sensor modalities and generate synthetic data that transfers effectively to real-world scenarios [61]–[63]. This establishes Isaac Sim's maturity as a platform for generating diverse, high-quality synthetic sensor data beyond traditional RGB imaging.

The synthetic data pipeline provides several key advantages for our FPP implementation:

- Domain randomization to systematically vary scene parameters including lighting conditions, material properties, and geometric configurations
- Parallel scene simulation across multiple GPUs to accelerate data collection
- Automated ground truth generation, providing perfect correspondence between captured images and 3D geometry

- Programmatic control of data capture sequences through the Python API

Through Replicator’s physics-based simulation, we achieve synthetic data that closely approximates real-world image capture while maintaining complete control over experimental conditions. This approach enables us to systematically validate the performance of our virtual FPP system across a wide range of environmental conditions and material properties that would be impractical to test with physical hardware.

Building on these capabilities, we implement our VIRTUS-FPP framework as described in the following section, leveraging the photorealistic rendering and scene composition tools to achieve high-fidelity simulation of fringe projection profilometry.

### III. VIRTUS-FPP: SYSTEM ARCHITECTURE AND IMPLEMENTATION

We present a complete implementation of the fringe projection profilometry active imaging technique in NVIDIA Isaac Sim. Based on the principles of FPP elaborated in Section II-A, complete virtual sensor modeling and development of this system was achieved. This section details the design of the simulated camera-projector system, followed by the generation of an asymmetric circular calibration board which we use for performing 18-step fringe projection profilometry calibration in simulation. Through this process, we acquire the virtual intrinsic and extrinsic camera and projector matrices for 3D reconstruction.

#### A. Virtual Camera-Projector System

We utilize the pre-defined pinhole camera asset within NVIDIA Isaac Sim with an attached RGB render product for synthetic data acquisition. To simulate a digital light projector, we utilize the rectangular light source asset defined by UsdLux. The rectangular light source asset has limited controllable parameters compared to a real-world projector. Rather than specifying optical centers, scalings, and a focus distance, the primary controllable parameters for light sources in simulation are intensity, physical dimensions, and texture mapping. Table I summarizes the key parameters for both the camera and projector assets used in our implementation.

Rectangular light sources in NVIDIA Isaac Sim are designed to emulate light generated from a panel with configurable width and height dimensions that control the aspect ratio of the projected light. The `isProjector` parameter, in conjunction with `texture:file`, enables a specified image to be projected as a texture onto surfaces. While the OmniGraph visual programming interface discussed in Section II-B lacks the flexibility to dynamically update this texture file parameter during simulation, we overcome this limitation by developing a custom Python extension that provides direct programmatic control over all simulation parameters.

Our custom user extension architecture is divided into three primary components:

- `setup_scene`: Defines the entire simulation scene including the camera-projector FPP system, calibration targets, and scanning objects

TABLE I: Parameters for Simulated Camera and Projector Assets

Camera Parameters	
Parameter	Value
Focal Length	50 cm
Horizontal Aperture	20.9995 cm
Vertical Aperture	15.2908 cm
Near Clip	0.5 m
Far Clip	1000.0 m
Resolution	960 × 960
Focus Distance	Auto (default)
Projector Parameters	
Parameter	Value
Intensity	40 nits
Height	0.625 m
Width	0.5 m
Color	White (RGB: 1,1,1)
Exposure	0.0
Texture File	Fringe pattern
Projector Light Type	True

- `setup_post_load`: Configures active scene operations such as object reorienting, fringe texture loading, and synthetic data acquisition through physics callbacks
- `setup_post_reset`: Resets key simulation parameters after each episode to ensure reproducible results

In `setup_scene`, we configure the camera and projector with the parameters shown in Table I. We define the projector as a rectangular light source with a height of 0.625 meters and a width of 0.5 meters to maintain the 5:4 aspect ratio of our generated fringe pattern images with known fringe shift. An additional overhead sphere light provides global scene illumination. To ensure optimal camera-projector triangulation for structured light reconstruction, the projector is positioned 0.1 meters below and 0.125 meters to the left of the camera, while sharing the same orientation and distance from scanning target objects. Here, parameterized mesh models for the objects we wish to scan are also defined, specifying their geometries, material properties, and physics.

We utilize the physically-based material system described in Section II-B to define consistent surface properties across our scanning objects. Specifically, each mesh is configured with material properties of 0.95 for roughness, 0.15 for specular, and 0.95 for ambient occlusion to diffuse ratio. These parameters create predominantly matte surfaces with minimal specular highlights and enhanced diffuse shading, enabling more accurate simulation of typical surface reflectance conditions encountered in structured light scanning [30], [64], [65]. Our simulation setup is shown in Figure 3.

#### B. Synthetic Data Acquisition

Leveraging the synthetic data generation capabilities discussed in Section II-B, fringe pattern projection and image acquisition are handled by our `update_texture_callback` function, initialized in `setup_post_load`. A set of fringe image patterns is first given as input to the `texture:file` attribute of the rectangular light source where each pattern is projected in the specified order at each time step. After a fringe image pattern is loaded and projected by the rectangular light source, the `update_texture_callback` function

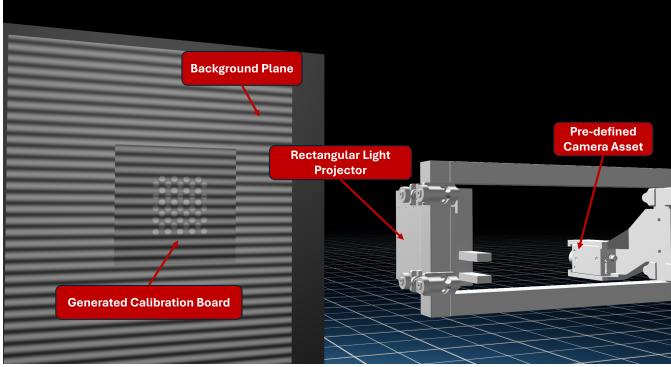


Fig. 3: Virtual camera-projector calibration setup with pre-defined pinhole camera asset, rectangular light source projector, generated calibration board, and matte-like background plane.

calls the camera’s render product to capture the current camera frame as a grayscale RGB image after a small delay to ensure complete rendering.

Through the GPU-accelerated rendering pipeline of NVIDIA Isaac Sim described in Section II-B, our system achieves significantly faster capture rates compared to physical systems. A complete simulation episode results in a full set of projected fringe images saved to a specified directory for phase analysis and 3D reconstruction. This high-throughput approach enables rapid acquisition of calibration datasets across multiple camera-projector configurations and environmental conditions.

#### IV. VIRTUAL CALIBRATION AND RECONSTRUCTION

##### A. Calibration Board Generation

To facilitate full end-to-end calibration within simulation, we present the procedural generation of custom asymmetric circular calibration boards. Leveraging the USD framework described in Section II-B, we programmatically generate and apply calibration patterns as texture maps to plane meshes. Given a particular number of rows, columns, a circle diameter, a circle center distance, and plane dimensions, we generate an RGB image of an asymmetric circular calibration board following these parameters to be applied as a diffuse texture image to the albedo material properties of a defined plane mesh model. Sampled calibration board configurations are shown in Figure 4.

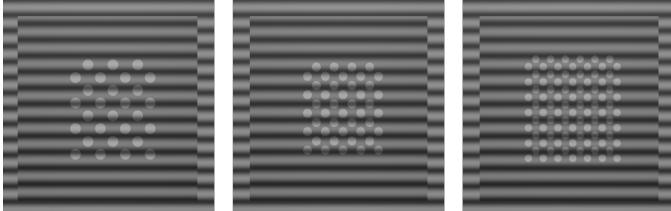


Fig. 4: Example generated circular calibration boards. A widely spaced  $4 \times 7$  board (left), a standard  $5 \times 9$  board (middle), and a densely spaced  $7 \times 13$  board (right).

As a result of applying the circular calibration board pattern as a diffuse texture map to a mesh, however, the original

measurements for circle diameter and circle center distance become scaled to fit the geometry of the plane. To determine the true dimensions of these calibration features in the simulation environment, we compute a scaling factor derived from the ratio of the physical plane dimensions to the calculated pattern dimensions. First, we calculate the pattern’s physical dimensions in meters:

$$W_{\text{pattern}} = \left( 2L_{\text{border}} + \frac{(C - 1)D_{\text{centers}}}{2} + D_{\text{circle}} \right) \cdot k_{\text{mm2m}} \quad (9)$$

$$H_{\text{pattern}} = (2L_{\text{border}} + (R - 1)D_{\text{centers}} + D_{\text{circle}}) \cdot k_{\text{mm2m}} \quad (10)$$

where  $L_{\text{border}}$  is the border length,  $C$  is the number of columns,  $R$  is the number of rows,  $D_{\text{centers}}$  is the circle center distance,  $D_{\text{circle}}$  is the circle diameter (all in millimeters), and  $k_{\text{mm2m}} = 0.001$  is the millimeter to meter conversion factor.

Next, we compute a scaling factor to fit the pattern within the plane dimensions:

$$S_{\text{pattern}} = \min \left( \frac{W_{\text{plane}}}{W_{\text{pattern}}}, \frac{H_{\text{plane}}}{H_{\text{pattern}}} \right) \quad (11)$$

where  $W_{\text{plane}}$  and  $H_{\text{plane}}$  are the plane width and height in meters. The final dimensions of the circles and their spacing in the simulation are then given by:

$$D_{\text{circle}}^{\text{sim}} = D_{\text{circle}} \cdot k_{\text{mm2m}} \cdot S_{\text{pattern}} \quad (12)$$

$$D_{\text{centers}}^{\text{sim}} = D_{\text{centers}} \cdot k_{\text{mm2m}} \cdot S_{\text{pattern}} \quad (13)$$

Empirical validation in NVIDIA Isaac Sim confirms that our calculated circle diameter and circle center distances match the measured values within 7% error (1-2mm difference). The ability to programmatically generate calibration boards with precise dimensional control provides a flexible foundation for virtual calibration, enabling systematic experimentation on diverse calibration patterns across simulation scenarios with reproducibility.

##### B. Virtual Calibration Process

Through the programmatic control and physics-based simulation capabilities of NVIDIA Isaac Sim discussed in Section II-B, we develop a systematic calibration procedure with precise calibration board poses. We define the calibration board plane to be a distance of 0.5 meters away from the front of the rectangular light source with the same orientation as the camera-projector system. This serves as the base position of the calibration board plane. We explicitly define a set of 18 unique poses for calibration wherein the calibration board plane is placed at unique orientations and deviations from this base position.

To ensure high accuracy camera calibration and sub-millimeter stereo and projector error, the calibration board plane is translated vertically by a deviation of  $\pm 0.015$  meters, laterally by a deviation of  $\pm 0.015$  meters, and reoriented

with 5 to 15 degree tilts along the x and z rotational axis across the poses. For each of the 18 poses shown in Figure 5, a full capture of every fringe pattern image given in the specified fringe image directory is saved. The GPU-accelerated synthetic data generation capabilities of NVIDIA Isaac Sim allow us to collect this set of 936 fringe image captures for 18-step fringe projection profilometry in just over 5 minutes, an average of 10,530 fringe image captures in one hour.

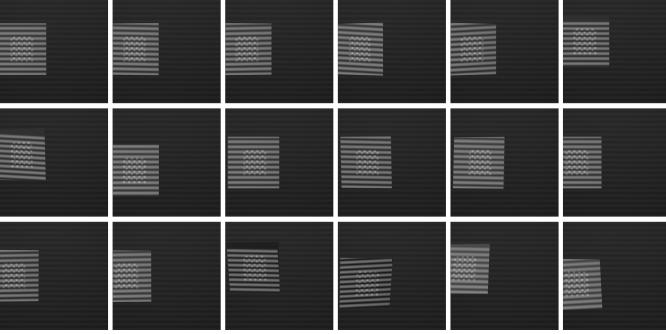


Fig. 5: Automatic repositioning of the generated circular calibration board in 18 distinct poses for virtual camera-projector system calibration.

With the full set of captured fringe images for each pose from simulation, we can begin calibration for our virtual system. Calibration is done using the conventional methodology mentioned in Section II-A. Algorithm 1 outlines our systematic virtual calibration procedure. The accuracy of the calibration methodology in this case, is determined by two error metrics, namely, stereo reprojection error and projector errors. Stereo reprojection error quantifies the geometric accuracy of camera calibration by measuring the pixel distance between detected and reprojected points, while projector error represents the discrepancy between expected and actual projection patterns on the target surface. The obtained **stereo reprojection error and projector errors were 0.055506 and 0.048609 respectively**, demonstrating the sub-pixel accuracy of our virtual system. With a fully virtual calibrated system, we can begin 3D reconstruction. Using the reconstruction principles discussed in II-A, we obtain the 3D reconstruction of a standard sphere mesh evaluated against its ground truth geometry and evaluate how our simulation’s reconstruction fidelity changes under various ambient lighting and material response scenarios in V to demonstrate the ability of our simulator to study FPP behavior under challenging conditions.

## V. EXPERIMENTAL VALIDATION AND RESULTS

### A. Reconstruction Accuracy Validation

To validate the reconstruction accuracy of our virtual fringe projection profilometry system, we performed 3D reconstruction of a standard sphere mesh, of radius 50 mm, using the calibrated parameters obtained from the virtual calibration process. The sphere geometry serves as an ideal test case due to its well-defined mathematical properties and uniform curvature, allowing for precise quantitative evaluation of reconstruction fidelity.

---

### Algorithm 1 Virtual FPP System Calibration

---

```

Require: Fringe patterns  $\{F_1, F_2, \dots, F_N\}$ , Pose set  $\{P_1, P_2, \dots, P_{18}\}$ 
Ensure: Calibrated camera matrix  $M^c$ , projector matrix  $M^p$ 
1: Initialize simulation environment
2: Load fringe pattern textures
3: for  $i = 1$  to  $18$  do
4:   Position calibration board at pose  $P_i$ 
5:    $captured\_images_i = \{\}$ 
6:   for  $j = 1$  to  $N$  do
7:     Project fringe pattern  $F_j$ 
8:     Wait for rendering completion
9:     Capture camera frame  $I_{i,j}$ 
10:     $captured\_images_i \leftarrow captured\_images_i \cup \{I_{i,j}\}$ 
11:   end for
12:   Extract circular features from  $captured\_images_i$ 
13:   Compute phase maps using Eq. (2)
14:   Perform temporal phase unwrapping using Eq. (3)
15: end for
16: Perform camera calibration using detected circle centers
17: Perform stereo calibration between camera and projector
18: Compute reprojection errors
19: if errors < threshold then
20:   return  $M^c, M^p$ 
21: else
22:   Refine calibration parameters and repeat
23: end if

```

---

The reconstruction process employed the phase-shifting algorithm described in Section II-A, utilizing the captured fringe patterns to generate unwrapped phase maps and subsequently compute 3D point coordinates through triangulation. The reconstructed point cloud data was then processed to generate a mesh representation of the sphere geometry.

For accuracy assessment, we performed point cloud alignment between the reconstructed sphere mesh and the baseline ground truth sphere mesh using iterative closest point (ICP) registration. This alignment process ensures optimal correspondence between the reconstructed and reference geometries, minimizing the influence of rigid body transformations on the error analysis.

Following the registration process, we use MATLAB’s built-in `pcfitsphere` function that utilizes M-estimator Sample Consensus (MSAC) for fitting a sphere using the pointcloud point [66]. This method is a variant of the Random Sampling Consensus (RANSAC) algorithm, which is an iterative algorithm for robust model fitting in the presence of outliers. RANSAC repeatedly samples minimal point sets to estimate model parameters and identifies the model with the largest consensus set of inliers. This method yields the best-fit radius and the percentage of inliers and outliers. The results are illustrated in Figure 6.

The sphere radius was calculated to be 50.512 mm, and the number of inliers was 133,802/134,512 (99.7%). To quantify the reconstruction accuracy, we calculated the absolute radial error ( $R_{abs}$ ), where  $R_{est}$  is the estimated radius from the sphere fitting and  $R_{act}$  is the true radius of the sphere mesh scanned

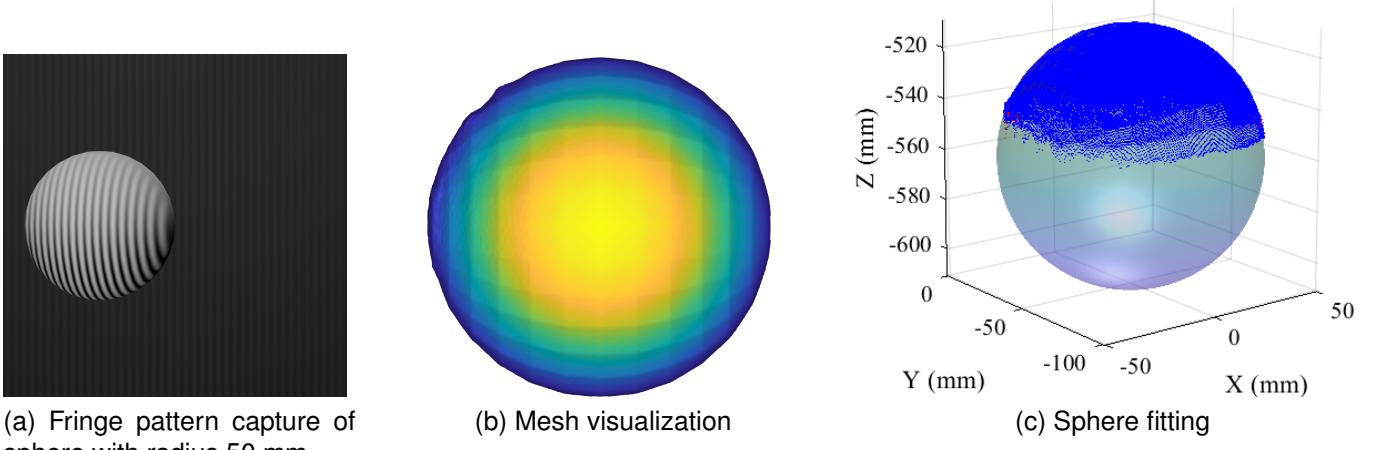


Fig. 6: Sphere reconstruction results with single overhead ambient lighting: (a) captured fringe pattern, (b) reconstructed 3D mesh of the sphere, and (c) overlay of the fitted ideal sphere and the measured sphere result with M-estimator SAmple Consensus (MSAC) yielding a radial error of 0.512 mm.

in the virtual environment:

$$R_{\text{abs}} = |R_{\text{est}} - R_{\text{act}}| = |50.512 - 50.0| = 0.512 \text{ mm} \quad (14)$$

This yields a relative error of 1.02% and 0.512 mm. These experimental results demonstrate sub-millimeter measurement accuracy, which validates the effectiveness of our virtual fringe projection system and confirms that the calibration parameters obtained through our methodology enable precise 3D reconstruction. The achieved accuracy is comparable to physical fringe projection systems, demonstrating the viability of our virtual approach for high-precision metrology applications. These results establish baseline reconstruction accuracy for our system under controlled conditions. The following subsection presents an ablation study examining how this accuracy varies with different environmental and material properties.

#### B. Adverse Conditions Testing

To demonstrate our simulator's capability to study the behavior of FPP under harsh operating conditions, we conduct a series of experiments with varied ambient lighting conditions and scanning target material properties.

For each configuration, a standard sphere mesh of radius 50 mm is used as the scanning target. We simulate the effect of outdoor extreme ambient lighting by creating two rectangular light sources of arbitrarily high intensity toward the left-hand and right-hand sides of the camera-projector system, in addition to the pre-existing overhead ambient sphere light in our simulation scene. Our ambient lighting experimental setup is shown in Figure 7.

We define our set of experimental configurations by combining different material properties and lighting conditions:

- **Material Properties:** We test three different material configurations:
  - **Baseline:** Standard matte material with ambient occlusion to diffuse ratio of 0.95, roughness of 0.95, and specular of 0.15.

- **AO\_to\_diffuse\_0:** Material with ambient occlusion to diffuse ratio set to 0 (disabled), which results in less shadowing in occluded areas and increased perceived lighting intensity.

- **Metallic:** Material with full metallic properties (metallic constant = 1.0), low surface roughness (0.2), and light gray base color, creating a silver-like appearance with significant specular reflections that challenge fringe pattern shape.

- **Lighting Conditions:** We test four different lighting scenarios:

- **Baseline:** Only the overhead ambient sphere light is enabled, providing uniform illumination.

- **No\_Ambient:** All ambient lighting is disabled, testing FPP performance with only the projector's fringe pattern illumination.

- **One\_Ambient:** Overhead sphere light plus one rectangular ambient light source positioned to the left of the sphere, simulating directional environmental lighting.

- **Two\_Ambient:** Overhead sphere light plus two rectangular ambient light sources positioned on both sides of the sphere, simulating complex environmental lighting with potential interference.

We conduct a detailed analysis of the 12 possible material-lighting combinations from this setup. Each experimental configuration follows the naming convention "MaterialProperty\_LightingCondition" (e.g., "Baseline\_Baseline" represents the standard matte material under uniform overhead lighting). Figure 8 demonstrates the progressive degradation of fringe pattern quality as material specularity increases and ambient lighting interference intensifies, with clearly visible fringes in the Baseline configurations deteriorating to nearly imperceptible patterns under highly reflective materials and extreme ambient lighting.

These experiments demonstrate a key advantage of our sim-

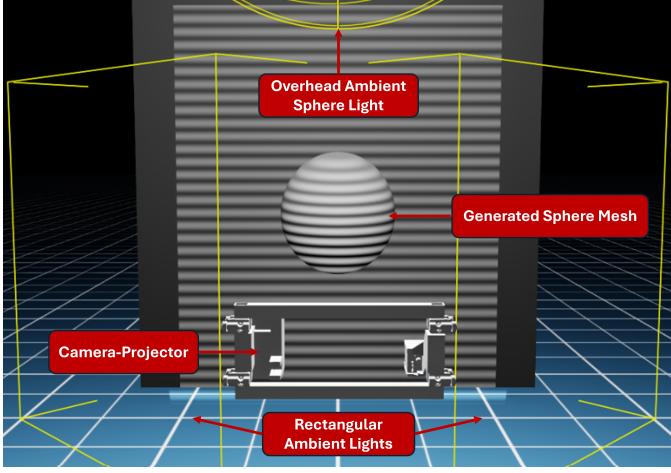


Fig. 7: Ambient lighting experiment setup in simulation: two rectangular ambient light sources, one overhead sphere ambient light source, and the generated standard sphere mesh of radius 50 mm.

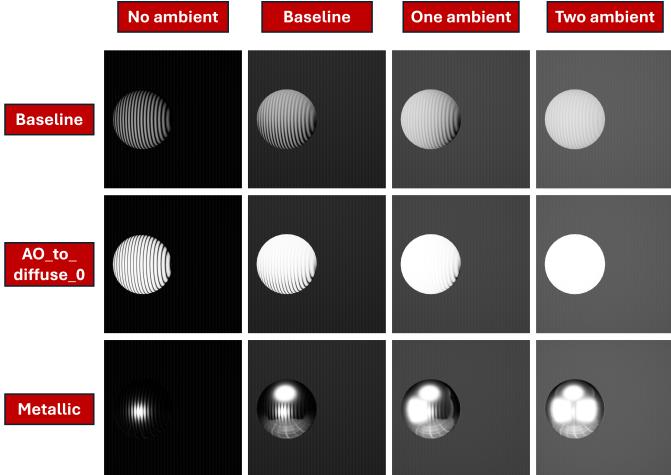


Fig. 8: Systematic comparison of fringe pattern quality under varying material properties and ambient lighting conditions. The figure shows a  $3 \times 4$  grid where rows represent material types (Baseline, AO\_to\_diffuse\_0, and Metallic) and columns represent lighting conditions (No ambient, Baseline, One ambient, Two ambient). The progression shows how increasing ambient light and material specularity degrade fringe pattern visibility.

ulator. It is well-known that FPP fails under outdoor conditions due to excess ambient light, subsurface scattering, reflection-induced over-saturation, and other challenging factors. The ability to simulate these conditions with photorealistic rendering and well-established ground truth, namely the precise geometry of the scanned asset, enables the development of more robust FPP systems capable of handling outdoor or extreme operating conditions. Furthermore, the availability of perfect ground truth data in simulation facilitates the training of deep learning models that can subsequently improve real-world FPP performance under challenging conditions.

This comprehensive experimental design enables us to eval-

uate the robustness of FPP under challenging conditions and provides valuable insights into how material properties and ambient lighting affect FPP performance. Such systematic testing would be extremely time-consuming or impossible to achieve with physical FPP systems, demonstrating a key advantage of our physics-based simulation approach. Beyond demonstrating the flexibility of our virtual system, however, a critical validation step is to establish correspondence between our simulation framework and physical FPP systems. This digital twin capability enables hybrid workflows that combine the advantages of both virtual and physical experimentation.

### C. Digital Twin Validation

The final step of experimentation is to demonstrate the capability to create a digital twin of a existing FPP system. In this process, the camera's intrinsic parameters (focal length, apertures, and principal points) obtained from stereo-calibration are transferred to the simulator using the following equations as provided in the NVIDIA Isaac Sim documentation [67].

$$f = \frac{f_x + f_y}{2} \cdot s_{mm} \quad (15)$$

$$A_h = W \cdot s_{mm} \quad (16)$$

$$A_v = H \cdot s_{mm} \quad (17)$$

where  $f$  denotes the effective focal length of the camera to be used in the simulation,  $A_h$  and  $A_v$  represent the horizontal and vertical aperture dimensions, respectively, and  $[W, H]$  correspond to the image resolution in pixels along the width and height. The parameter  $s_{mm}$  indicates the physical pixel size (in millimeters), while  $f_x$  and  $f_y$  denote the focal lengths along the x- and y-axes as obtained from the calibrated camera intrinsic matrix.

On the other hand, the current version of this software does not provide a direct way to transfer the calibration matrices of the projector from the real system to simulation. In Isaac Sim, light sources can project textures similar to fringe patterns used in FPP; however, users must specify the physical dimensions (in meters) of the projected image to accurately emulate digital light projector (DLP) behavior. No other digital twin simulator, such as Blender, provides a direct interface for specifying projector intrinsic parameters, which is essential for accurate projector modeling in simulation. Furthermore, when transferring real-world projector properties to simulation, direct measurement of projected pattern dimensions is impractical due to two primary limitations. First, the inverse-square law of light propagation causes dimensional variations at different distances between the screen and the camera-projector system. Second, such distance-dependent measurements are inherently prone to systematic errors that would compromise the digital twin's fidelity. This necessitates a robust theoretical formulation to determine the metric dimensions of the projected image at a given distance, an approach that our fully developed simulator enables and validates.

Hence, in order to obtain the metric dimensions of the projector image, in simulation, from the intrinsic and extrinsic matrices of a real-world calibrated FPP system, we use the

inverse pinhole camera model [68]. The mathematical description utilized here is as follows:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = (M_{\text{ext}})^{-1} (M_{\text{int}})^{-1} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \quad (18)$$

where,  $M_{\text{ext}}$  and  $M_{\text{int}}$  correspond to the extrinsic and intrinsic matrices of the projector, obtained during calibration of the real-world FPP system. The  $M_{\text{ext}}, M_{\text{int}}$  for the projector as well as camera are of the format given below:

$$M_{\text{int}} = \begin{bmatrix} f_x & 0 & o_x \\ 0 & f_y & o_y \\ 0 & 0 & 1 \end{bmatrix}, \quad (19)$$

$$M_{\text{ext}} = [R_p \mid T_p] = \begin{bmatrix} r_{11} & r_{21} & r_{31} & t_x \\ r_{12} & r_{22} & r_{32} & t_y \\ r_{13} & r_{23} & r_{33} & t_z \end{bmatrix}$$

where,  $f_x, o_x, f_y, o_y$  correspond to the focal lengths and principal points along  $x-, y-$ axes respectively.  $R_p, T_p$  correspond to the extrinsic parameters, namely, rotation and translation vectors of the projector with respect to the camera. In order to model the projector in NVIDIA Isaac Sim, we are required to find the inverse of these matrices to obtain the metric dimensions of the projected image at a given distance. For the intrinsic matrix, being a square matrix, this is a straight-forward process and the inverse is calculated as shown below:

$$M_{\text{int}}^{-1} = \begin{bmatrix} \frac{1}{f_x} & 0 & -\frac{o_x}{f_x} \\ 0 & \frac{1}{f_y} & -\frac{o_y}{f_y} \\ 0 & 0 & 1 \end{bmatrix} \quad (20)$$

In contrast to the intrinsic matrix, the extrinsic matrix  $M_{\text{ext}} \in \mathbb{R}^{3 \times 4}$  is non-square and therefore not directly invertible. As a result, we compute its Moore–Penrose pseudo-inverse via singular value decomposition (SVD), yielding a  $4 \times 3$  matrix. Mathematically, this pseudo-inverse introduces a many-to-one mapping due to the inherent ambiguity of projecting from a lower-dimensional space (2D image coordinates) to a higher-dimensional space (3D world coordinates), losing depth information. However, we can circumvent through a natural constraint provided by the pinhole camera model: the projected image dimensions scale linearly with the distance from the camera (or projector), following the geometry of similar triangles [69]. This proportional scaling implies that for any given image point, the corresponding world coordinates lie along a ray parameterized by the depth  $Z$ . In our formulation, we explicitly resolve this ambiguity by parameterizing the  $X$  and  $Y$  world coordinates in terms of the depth  $Z$ . Hence, factoring  $Z$  from the left-hand side of the projection equation yields the following expression:

$$Z \begin{bmatrix} \frac{X}{Z} \\ \frac{Y}{Z} \\ 1 \end{bmatrix} = (M_{\text{ext}})^{-1} (M_{\text{int}})^{-1} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \quad (21)$$

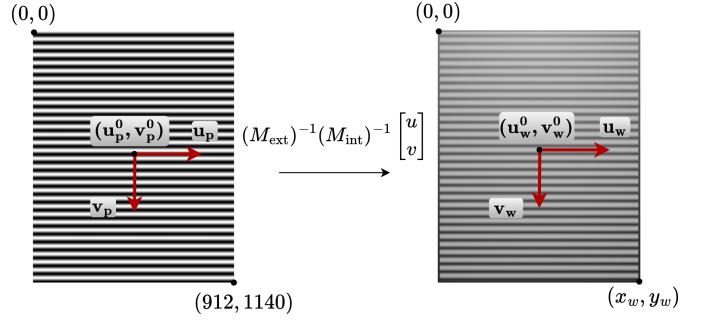


Fig. 9: Coordinate transformation from projector image space to world coordinates, illustrating the mathematical conversion process used to determine projected pattern dimensions in metric units.

$$\implies \begin{bmatrix} x_w \\ y_w \\ 1 \end{bmatrix} = Z \left( (M_{\text{ext}})^{-1} (M_{\text{int}})^{-1} \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \right) \quad (22)$$

Here,  $(u, v)$  and to obtain the real-world dimensions of the projected fringe image, we find the  $X, Y$  values at the image coordinates  $[0, 0]$  and  $[912, 1140]$  to get the  $x_w, y_w$  of the corner points and hence the height and width of the image in metric dimesnions. This process is illustrated in Figure 9.

In order to validate our proposed mathematical approach, we used the camera-projector matrices obtained during the calibration of the virtual system, where the calibration board was positioned exactly 1 meter from the camera-projector system. The **projector error and stereo reprojection error are 0.058357 and 0.065518 respectively**, indicating perfect calibration. The intrinsic and extrinsic projector matrices, as well as the values obtained for the metric dimensions of the projector pattern at 1m are shown below.

$$M_{\text{int}} = \begin{bmatrix} 1820.10 & 0 & 455.74 \\ 0 & 1819.95 & 571.74 \\ 0 & 0 & 1 \end{bmatrix}, \quad (23)$$

$$M_{\text{ext}} = \begin{bmatrix} 1.0000 & 0.0000 & -0.0001 & 89.72 \\ -0.0000 & 0.9999 & -0.0012 & -71.70 \\ 0.0001 & 0.0012 & 0.9999 & -0.75 \end{bmatrix}$$

$$\begin{bmatrix} x_w \\ y_w \end{bmatrix} = \begin{bmatrix} 626.3 \\ 501.1 \end{bmatrix} \quad (24)$$

These values are very close to the actual dimensions of the initial projector which was set during calibration as shown in Table I. This proved as an initial validation of our projector calculations from Equation 22.

To further validate this mathematical formulation, we measured the dimensions of projected fringes using the Measure Tool in NVIDIA Isaac Sim at distances ranging from 400 to 1000 mm. We compared the theoretical values predicted by the inverse camera model (Equation 18) with the corresponding measured values. Figure 10 illustrates how the width and height vary with distance from the camera-projector system, which is analogous to the linear perspective projection scaling inherent to the geometric model, as modeled by Equation 18. The mean absolute errors (MAE) between measured and

theoretical values were 1.7014 mm for width and 1.4170 mm for height, respectively. These small discrepancies demonstrate excellent agreement between theoretical predictions and experimental measurements, strongly validating the accuracy of the theoretical model.

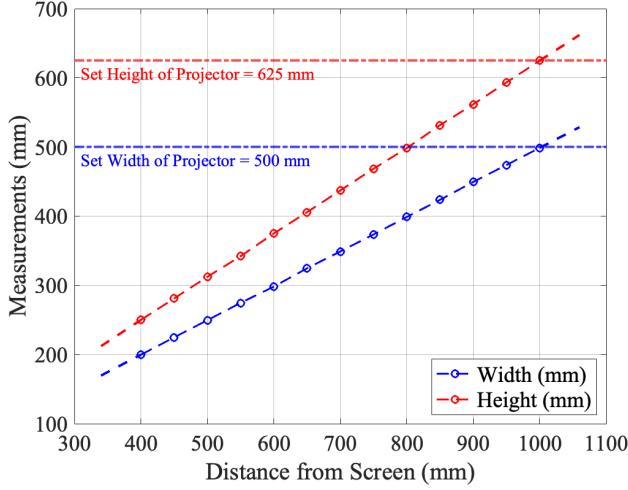


Fig. 10: Measured projected fringe width and height at various distances. The mean absolute errors (MAEs) between measured and theoretical values are 1.7014 mm for width and 1.4170 mm for height.

Having comprehensively validated our mathematical formulation using the virtual system, we proceeded to implement the digital twin of our calibrated real-world FPP system. The calibrated real-world FPP system employed for this validation is illustrated in Figure 11.

The intrinsic and extrinsic matrices obtained from the physical system calibration were applied to Equation 22 to determine the appropriate projector dimensions for simulation. Using this mathematical framework, we calculated the metric dimensions of the projected fringe image:

$$\begin{bmatrix} x_w \\ y_w \end{bmatrix} = \begin{bmatrix} 202.7 \\ 323.3 \end{bmatrix} \quad (25)$$

where the values are expressed in meters. To validate the accuracy of our digital twin implementation, we conducted comparative measurements between the physical FPP system and its simulated counterpart at a screen distance of 0.4 meters from the camera-projector system. The projected fringe pattern dimensions from both systems were measured and compared to assess the real-to-sim transfer accuracy. The experimental results confirm excellent agreement between the physical and simulation formulation shown above systems, as demonstrated in Figure 12.

However, when setting the virtual projector width and height attributes to these calculated dimensions, we observed a dimensional mismatch at the specified screen distance of 0.4 meters. Through systematic experimentation, we discovered that setting the projector's width and height attributes to the values obtained from Equation 22 with  $Z = 1$  meter resulted in accurate dimensional correspondence of the projected fringe

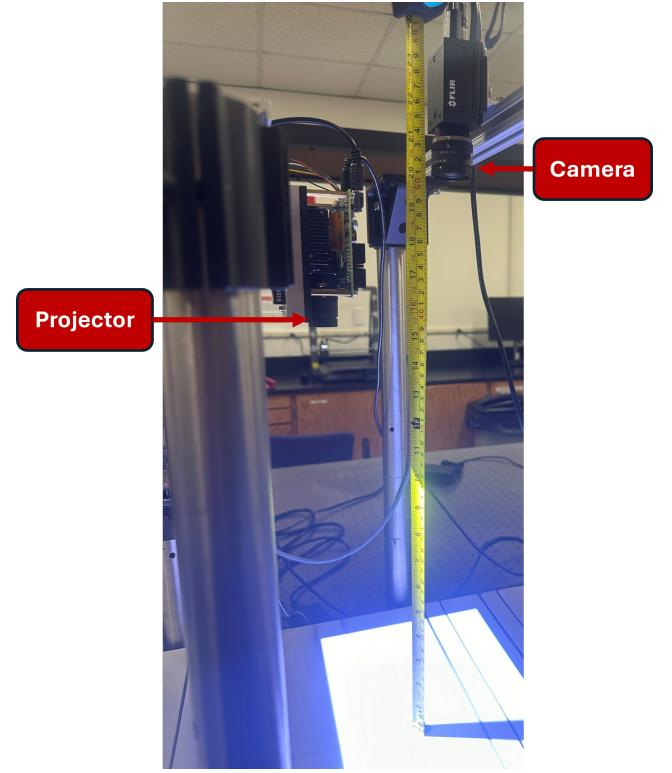


Fig. 11: Physical fringe projection profilometry system used for digital twin validation, with measuring tape indicating the 400 mm distance to the projection screen.

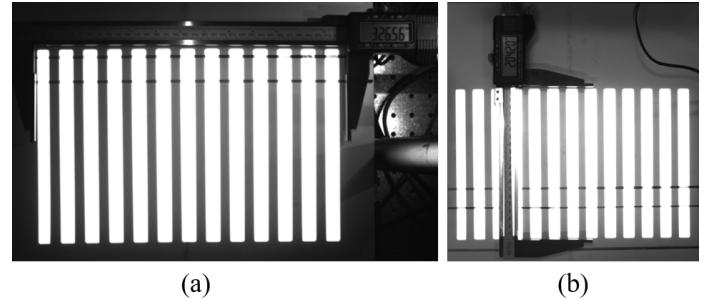


Fig. 12: The measured dimensions of the real-world fringes at 0.4 m are illustrated in this figure. The dimensions are as follows: (a) Height = 326.56 mm; (b) Width = 204.20 mm

width and height with the projected fringe dimension measurements from the real-world system and the theoretical model.

We attribute this phenomenon to the inherent computational implementation of the `RectLight` primitive within NVIDIA Isaac Sim's rendering pipeline, where we believe 1 m may be defined as the set reference distance where the projected fringe width and height corresponds exactly to the set projector width and height attributes. Despite this scaling artifact, our empirically validated approach successfully establishes the requisite fidelity for digital twin implementation. This demonstrates the capability of VIRTUS-FPP to accurately replicate real-world FPP system behavior, providing a robust foundation for virtual fringe projection profilometry applications.

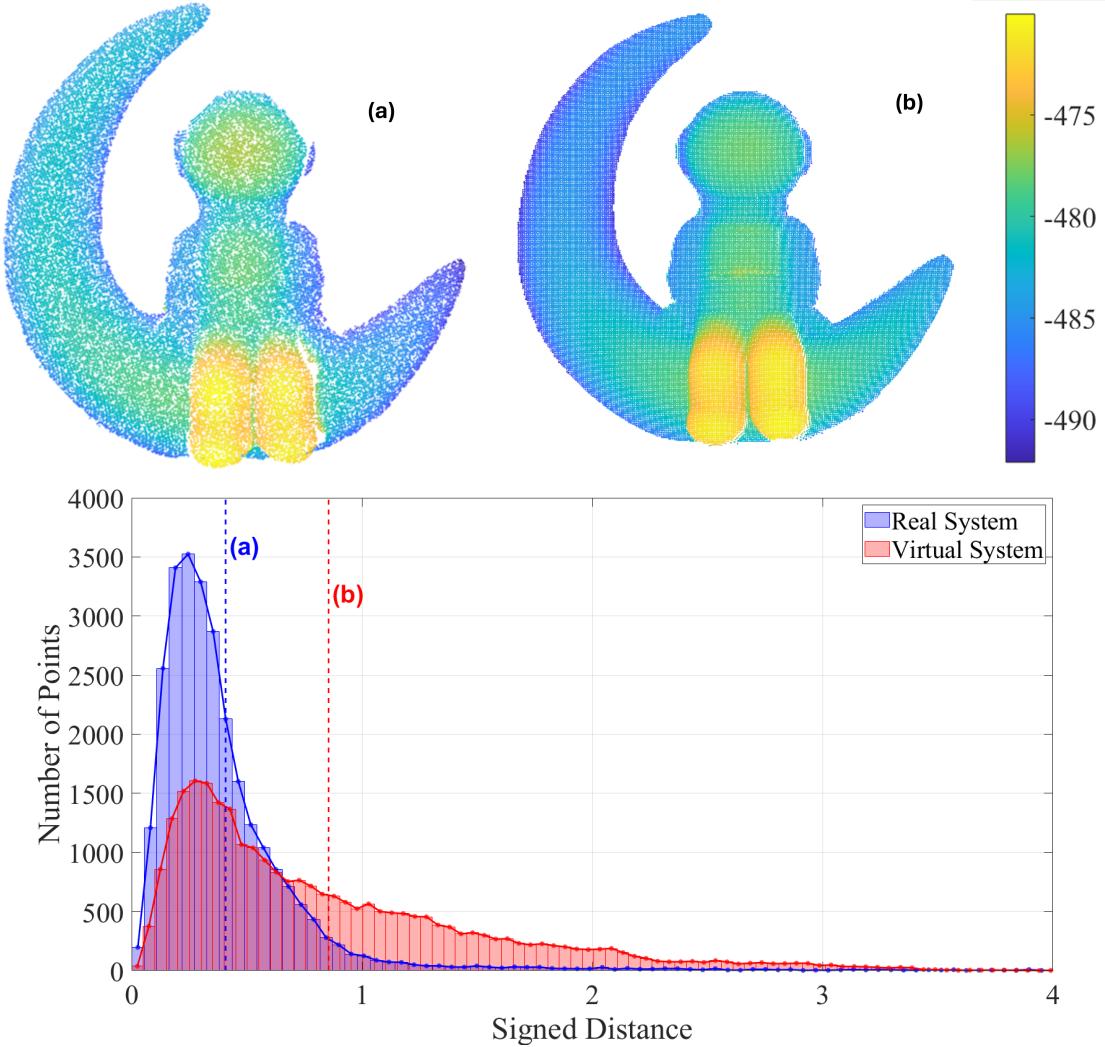


Fig. 13: Reconstruction comparison between (a) virtual system and (b) physical system on identical astronaut figurine targets. The histograms of the C2M distances (c) for the real system (a) and virtual system (b) are left-skewed distributions peaking between 0-1 mm, validating digital twin fidelity through sub-millimeter accuracy.

#### D. 3D Reconstruction

With both real and virtual systems fully calibrated, we proceeded to compare reconstructions obtained from each system. The test object was a 3D printed astronaut figurine, whose .stl file was imported into the simulation environment and scanned using the digital twin system developed in the previous section.

To quantitatively assess reconstruction accuracy, we compared both real and virtual system reconstructions of the astronaut figurine target with the original .stl file using Cloud-to-Mesh (C2M) distance analysis. This methodology involves aligning the reconstructed point clouds with the reference mesh using Iterative Closest Point (ICP) registration, followed by C2M distance computation to quantify geometric conformance. Lower C2M distances indicate higher fidelity reconstruction, while larger deviations suggest discrepancies due to measurement errors or reconstruction artifacts. The detailed methodology for this analysis is adapted from [70].

The reconstructions and C2M distance histograms presented

in Figure 13 demonstrate high reconstruction fidelity for both FPP systems. Quantitative analysis of the C2M distance distributions reveals that the majority of points for both real and virtual systems fall within the 0-1 mm range, consistent with the established measurement precision of structured light systems. The statistical agreement between real and virtual system performance validates the accuracy of our digital twin implementation and confirms that the virtual FPP system achieves reconstruction precision comparable to its physical counterpart. These results substantiate the effectiveness of our inverse camera model approach for creating high-fidelity digital twins of fringe projection profilometry systems with our VIRTUS-FPP framework.

## VI. DISCUSSION

### A. Key Advantages of Simulation-Based FPP

The VIRTUS-FPP framework, built on NVIDIA Isaac Sim as described in Section II-B, offers several key advantages over physical FPP systems:

- **System Configuration Flexibility:** The programmatic control of camera and projector parameters enables rapid experimentation with different optical configurations without the physical constraints of hardware mounting and adjustment. This allows researchers to optimize system setups virtually before building physical prototypes.
- **Controlled Testing Environment:** The ability to precisely control environmental conditions such as ambient lighting and material properties provides a level of experimental control impossible to achieve in physical setups. This controlled environment is particularly valuable for systematically evaluating the impact of specific environmental factors on reconstruction accuracy.
- **Perfect Ground Truth Availability:** Unlike physical systems where ground truth geometries are often approximated or measured with other instruments, the simulation provides exact geometric ground truth for validation. This enables precise quantification of reconstruction errors under different conditions.

These advantages significantly accelerate the research and development cycle for FPP systems, enabling more thorough exploration of the parameter space and more rigorous validation of reconstruction algorithms.

### B. Limitations and Challenges

Despite the advantages, our VIRTUS-FPP framework faces several challenges and limitations:

- **Computational Requirements:** The photorealistic rendering capabilities discussed in Section II-B require modern GPU hardware for optimal performance. Our implementation requires at minimum a NVIDIA GeForce RTX 3070 with at least 8GB VRAM to achieve real-time performance. Systems with lower VRAM capacity may experience significantly reduced frame rates, reducing the efficiency of the synthetic data acquisition process.
- **Rendering Noise:** The path-traced rendering in NVIDIA Isaac Sim, while physically accurate, can introduce noise that affects phase calculation. We discovered that configuring specific global ray tracing parameters is critical for obtaining clean fringe patterns. In particular, we disabled sampled direct lighting mode under the real-time renderer ray tracing configurations. Without this setting, significant pixel-level artifacts appear in the phase maps that degrade reconstruction accuracy.
- **Simulation-to-Reality Gap:** While NVIDIA Isaac Sim provides highly realistic rendering, discrepancies between simulated and real-world optical phenomena remain. Subtle effects such as lens distortion, sensor noise, and projector gamma response are simplified in the simulation, potentially affecting the direct transferability of algorithms trained or validated solely in simulation.
- **Scaling factor consideration:** The virtual FPP reconstruction exhibits a consistent empirically calculated scaling factor of 1.37 for accurate real-world correspondence. This reproducible scaling behavior preserves all geometric feature relationships while providing a systematic

transformation that has been experimentally determined and integrated into the processing workflow. We believe this scaling is attributed to the simulation environment's optical modeling.

## VII. CONCLUSION AND FUTURE WORK

In this paper, we presented VIRTUS-FPP, a comprehensive physics-based virtual sensor modeling framework for fringe projection profilometry built on NVIDIA Isaac Sim. By leveraging the advanced rendering capabilities, physically accurate material modeling, and programmatic control described in Section II-B, our framework enables end-to-end simulation of the entire FPP pipeline from calibration to 3D reconstruction.

Our key contributions include: (1) the development of a complete virtual camera-projector system that accurately simulates the optical principles of FPP; (2) a calibration methodology that achieves sub-pixel accuracy in a virtual environment; (3) validation of reconstruction accuracy against ground truth geometry; (4) demonstration of the framework's flexibility for experimenting with environmental conditions and material properties; and (5) established a true digital twin framework that accurately models projector intrinsic parameters through the inverse camera model.

VIRTUS-FPP significantly accelerates the development cycle for FPP systems by enabling rapid virtual prototyping and systematic evaluation of system performance under various conditions. The framework's ability to generate large volumes of synthetic data with perfect ground truth annotation also opens new possibilities for training machine learning algorithms for tasks such as phase unwrapping and surface reconstruction.

Future work will focus on extending the framework to support multi-view FPP configurations, integration with robotic manipulation for automated scanning sequences, and development of hybrid physical-virtual workflows that combine real-world calibration with simulated data generation. Additionally, we plan to explore domain adaptation techniques to further reduce the simulation-to-reality gap, enabling more direct transfer of algorithms from virtual to physical systems.

## REFERENCES

- [1] S. Zhang, *High-Speed 3D Imaging with Digital Fringe Projection Techniques*. CRC Press, 1st ed., 2016.
- [2] J. Geng, “Structured-light 3d surface imaging: a tutorial,” *Advances in optics and photonics*, vol. 3, no. 2, pp. 128–160, 2011.
- [3] T. Bell, B. Li, and S. Zhang, “Structured light techniques and applications,” in *Wiley Encyclopedia of Electrical and Electronics Engineering*, pp. 1–24, John Wiley & Sons, Inc., 2016.
- [4] B. Balasubramaniam, J. Li, L. Liu, and B. Li, “3d imaging with fringe projection for food and agricultural applications—a tutorial,” *Electronics*, vol. 12, no. 4, p. 859, 2023.
- [5] J. Salvi, J. Pages, and J. Batlle, “Pattern codification strategies in structured light systems,” *Pattern recognition*, vol. 37, no. 4, pp. 827–849, 2004.
- [6] J. Xu and S. Zhang, “Status, challenges, and future perspectives of fringe projection profilometry,” *Optics and Lasers in Engineering*, vol. 135, p. 106193, 2020.
- [7] G. Sansoni, M. Trebeschi, and F. Docchio, “Three-dimensional vision based on a combination of gray-code and phase-shift light projection: analysis and compensation of the systematic errors,” *Applied optics*, vol. 38, no. 31, pp. 6565–6573, 1999.

- [8] H. Zhang, C. K. P. Vallabh, and X. Zhao, "Machine learning enhanced high dynamic range fringe projection profilometry for in-situ layer-wise surface topography measurement during lpbf additive manufacturing," *Precision Engineering*, vol. 84, pp. 1–14, 2023.
- [9] J. Qian, S. Feng, M. Xu, T. Tao, Y. Shang, Q. Chen, and C. Zuo, "High-resolution real-time 360° 3d surface defect inspection with fringe projection profilometry," *Optics and Lasers in Engineering*, vol. 137, p. 106382, 2021.
- [10] A. Dickins, T. Widjanarko, D. Sims-Waterhouse, A. Thompson, S. Lawes, N. Senin, and R. Leach, "Multi-view fringe projection system for surface topography measurement during metal powder bed fusion," *Journal of the Optical Society of America A*, vol. 37, no. 9, pp. B93–B105, 2020.
- [11] V. Suresh, W. Liu, M. Zheng, and B. Li, "High-resolution structured light 3d vision for fine-scale characterization to assist robotic assembly," in *Dimensional Optical Metrology and Inspection for Practical Applications X*, vol. 11732, p. 1173203, SPIE, April 2021.
- [12] J. Xu, G. Rao, and Z. Chen, "Robotic visual servoing using fringe projection profilometry," in *Optical Metrology and Inspection for Industrial Applications V*, vol. 10819, pp. 134–142, SPIE, 2018.
- [13] K. Bai, L. Zhang, Z. Chen, F. Wan, and J. Zhang, "Close the sim2real gap via physically-based structured light synthetic data simulation," in *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 17035–17041, IEEE, 2024.
- [14] A. Lakshman, F. Delzendehrooy, B. Balasubramaniam, G. E. Kremer, Y. Liao, and B. Li, "Corrosion characterization of engine connecting rods using fringe projection profilometry and unsupervised machine learning," *Measurement Science and Technology*, vol. 35, no. 8, p. 085021, 2024.
- [15] P. S. Huang, F. Jin, and F.-P. Chiang, "Quantitative evaluation of corrosion by a digital fringe projection technique," *Optics and Lasers in Engineering*, vol. 31, no. 5, pp. 371–380, 1999.
- [16] Y. Zhao, A. Pilvar, A. Tank, H. Peterson, J. Jiang, J. C. Aster, J. P. Dumas, M. C. Pierce, and D. Roblyer, "Shortwave-infrared mesopatterned imaging enables label-free mapping of tissue water and lipid content," *Nature communications*, vol. 11, no. 1, p. 5355, 2020.
- [17] K. Genovese and C. Pappalettere, "Whole 3d shape reconstruction of vascular segments under pressure via fringe projection techniques," *Optics and lasers in engineering*, vol. 44, no. 12, pp. 1311–1323, 2006.
- [18] Y. Zheng and B. Li, "Uniaxial high-speed microscale three-dimensional surface topographical measurements using fringe projection," *Journal of Micro-and Nano-Manufacturing*, vol. 8, no. 4, p. 041007, 2020.
- [19] H. Du, X. Chen, J. Xi, C. Yu, and B. Zhao, "Development and verification of a novel robot-integrated fringe projection 3d scanning system for large-scale metrology," *Sensors*, vol. 17, no. 12, p. 2886, 2017.
- [20] Y. Zhao, H. Yu, Y. Zheng, Y. Zhang, D. Zheng, and J. Han, "Scene-adaptive pattern coding-based fringe projection profilometry: diffuse surfaces identification and 3-d reconstruction in cluttered scenes," *Optics Express*, vol. 31, no. 20, pp. 32565–32581, 2023.
- [21] S. Feng, C. Zuo, L. Zhang, T. Tao, Y. Hu, W. Yin, J. Qian, and Q. Chen, "Calibration of fringe projection profilometry: A comparative review," *Optics and lasers in engineering*, vol. 143, p. 106622, 2021.
- [22] S. Asamura, Y. Naganawa, and N. Fukushima, "Fast drawing method of circular patterns based on gaussian circles for camera calibration," in *SPIE Future Sensing Technologies 2024*, vol. 13083, pp. 80–86, SPIE, May 2024.
- [23] S. Lv and Q. Kemao, "Modeling the measurement precision of fringe projection profilometry," *Light: Science & Applications*, vol. 12, no. 1, p. 257, 2023.
- [24] S. Zhang, "Absolute phase retrieval methods for digital fringe projection profilometry: A review," *Optics and lasers in engineering*, vol. 107, pp. 28–37, 2018.
- [25] R. Zhang, M. Duan, X. Fan, Y. Zheng, Z. Sun, J. Zheng, and Y. Jin, "Deep learning-enabled anti-ambient light approach for fringe projection profilometry," *Optics Express*, vol. 30, no. 26, pp. 47672–47689, 2022.
- [26] F. Li, D. Stoddart, and I. Zwierzak, "A performance test for a fringe projection scanner in various ambient light conditions," *Procedia CIRP*, vol. 62, pp. 400–404, 2017.
- [27] S. Zhang, "Recent progresses on real-time 3d shape measurement using digital fringe projection techniques," *Optics and lasers in engineering*, vol. 48, no. 2, pp. 149–158, 2010.
- [28] S. Feng, L. Zhang, C. Zuo, T. Tao, Q. Chen, and G. Gu, "High dynamic range 3d measurements with fringe projection profilometry: a review," *Measurement Science and Technology*, vol. 29, no. 12, p. 122001, 2018.
- [29] B. Salahieh, Z. Chen, J. J. Rodriguez, and R. Liang, "Multi-polarization fringe projection imaging for high dynamic range objects," *Optics express*, vol. 22, no. 8, pp. 10064–10071, 2014.
- [30] Y. Wei, J. Xi, Y. Yu, Q. Guo, and Y. Yin, "Experimental study for the influence of surface characteristics on the fringe patterns," in *Optical Metrology and Inspection for Industrial Applications III*, vol. 9276, pp. 177–182, SPIE, 2014.
- [31] Y. Xu, H. Zhao, H. Jiang, and X. Li, "High-accuracy 3d shape measurement of translucent objects by fringe projection profilometry," *Optics express*, vol. 27, no. 13, pp. 18421–18434, 2019.
- [32] X. Liu, Z. Zhang, N. Gao, and Z. Meng, "3d shape measurement of diffused/specular surface by combining fringe projection and direct phase measuring deflectometry," *Optics Express*, vol. 28, no. 19, pp. 27561–27574, 2020.
- [33] Z. Wu, H. Xuan, C. Sun, W. Guan, K. Zhang, and Y. Yan, "Semi-supervised video inpainting with cycle consistency constraints," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 22586–22595, June 2023.
- [34] Y. Li, A. Dai, L. Guibas, and M. Nießner, "Database-assisted object retrieval for real-time 3d reconstruction," in *Computer graphics forum*, vol. 34, pp. 435–446, Wiley Online Library, 2015.
- [35] S. Choi, Q.-Y. Zhou, S. Miller, and V. Koltun, "A large dataset of object scans," *arXiv preprint arXiv:1602.02481*, 2016.
- [36] L. Wright and S. Davidson, "Digital twins for metrology; metrology for digital twins," *Measurement Science and Technology*, vol. 35, p. 051001, feb 2024.
- [37] M. Vlaeyen, H. Haitjema, and W. Dewulf, "Digital twin of an optical measurement system," *Sensors*, vol. 21, no. 19, 2021.
- [38] F. Gomez-Donoso, A. Garcia-Garcia, J. Garcia-Rodriguez, S. Orts-Escalano, and M. Cazorla, "Lonchanet: A sliced-based cnn architecture for real-time 3d object recognition," in *2017 International Joint Conference on Neural Networks (IJCNN)*, pp. 412–418, IEEE, 2017.
- [39] P. Stavroulakis, S. Chen, C. Delorme, P. Bointon, G. Tzimiropoulos, and R. Leach, "Rapid tracking of extrinsic projector parameters in fringe projection using machine learning," *Optics and Lasers in Engineering*, vol. 114, pp. 7–14, 2019.
- [40] C. Zuo, J. Qian, S. Feng, W. Yin, Y. Li, P. Fan, J. Han, K. Qian, and Q. Chen, "Deep learning in optical metrology: a review," *Light: Science & Applications*, vol. 11, no. 1, p. 39, 2022.
- [41] K. Wang, Q. Kemao, J. Di, and J. Zhao, "Deep learning spatial phase unwrapping: a comparative review," *Advanced Photonics Nexus*, vol. 1, no. 1, p. 014001, 2022.
- [42] K. Yan, Y. Yu, C. Huang, L. Sui, K. Qian, and A. Asundi, "Fringe pattern denoising based on deep learning," *Optics Communications*, vol. 437, pp. 148–152, 2019.
- [43] S. I. Nikolenko *et al.*, *Synthetic data for deep learning*, vol. 174. Springer, 2021.
- [44] C. M. De Melo, A. Torralba, L. Guibas, J. DiCarlo, R. Chellappa, and J. Hodgins, "Next-generation deep learning based on simulators and synthetic data," *Trends in cognitive sciences*, vol. 26, no. 2, pp. 174–187, 2022.
- [45] J. Tobin, R. Fong, A. Ray, J. Schneider, W. Zaremba, and P. Abbeel, "Domain randomization for transferring deep neural networks from simulation to the real world," in *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pp. 23–30, IEEE, 2017.
- [46] Y. Zheng, S. Wang, Q. Li, and B. Li, "Fringe projection profilometry by conducting deep learning from its digital twin," *Optics Express*, vol. 28, no. 24, pp. 36568–36583, 2020.
- [47] K. Ueda, K. Ikeda, O. Koyama, and M. Yamada, "Fringe projection profilometry system verification for 3d shape measurement using virtual space of game engine," *Optical Review*, vol. 28, no. 6, pp. 723–729, 2021.
- [48] Q. Zhang, M. Xing, H. Li, X. Li, and T. Wang, "Measurement simulation system of fringe projection profilometry based on ray tracing," *IEEE Access*, vol. 11, pp. 89616–89624, 2023.
- [49] P. Tang, G. Sa, J. Ge, Z. Liu, and J. Tan, "Projection pattern pre-correction method based on projection error decoupling in fringe projection profilometry," *IEEE Transactions on Instrumentation and Measurement*, 2025.
- [50] B. Foundation, "Cycles - blender manual," 2024.
- [51] U. Technologies, "High definition render pipeline," 2024.
- [52] S. G. Parker, J. Bigler, A. Dietrich, H. Friedrich, J. Hoberock, D. Luebke, D. McAllister, M. McGuire, K. Morley, A. Robison, *et al.*, "Optix: a general purpose ray tracing engine," *Acm transactions on graphics (tog)*, vol. 29, no. 4, pp. 1–13, 2010.
- [53] NVIDIA, "What is isaac sim??" <https://docs.omniverse.nvidia.com/isaacsim/>, 2025. Accessed: 2025-05-20.

- [54] A. Haroon, A. Lakshman, M. Mundy, and B. Li, “Autonomous robotic 3d scanning for smart factory planning,” in *Dimensional Optical Metrology and Inspection for Practical Applications XIII*, vol. 13038, pp. 110–118, SPIE, 2024.
- [55] Pixar Animation Studios, “Universal scene description (usd).” <https://openusd.org/>, 2025. Accessed: 2025-05-20.
- [56] NVIDIA, “Physx sdk documentation.” <https://nvidia-omniverse.github.io/PhysX/physx/>, 2025. Accessed: 2025-05-20.
- [57] NVIDIA, “Nvidia optix 9.0.” <https://raytracing-docs.nvidia.com/optix9/index.html>, 2025. Accessed: 2025-05-20.
- [58] J. Pietrzak, K. Kacperski, and M. Cieślar, “Nvidia optix ray-tracing engine as a new tool for modelling medical imaging systems,” in *Medical Imaging 2015: Physics of Medical Imaging*, vol. 9412, pp. 667–672, SPIE, 2015.
- [59] F. Mauch, M. Gronle, and W. Osten, “Gpu-accelerated ray tracing for optical simulation and design,” in *Classical Optics 2014*, p. IW1A.4, Optica Publishing Group, 2014.
- [60] NVIDIA, “Replicator — omniverse extensions.” [https://docs.omniverse.nvidia.com/extensions/latest/ext\\_replicator.html](https://docs.omniverse.nvidia.com/extensions/latest/ext_replicator.html), 2025. Accessed: 2025-05-20.
- [61] T. Reinold, S. Ghosh, and G. Gallego, “Combined physics and event camera simulator for slip detection,” in *Proceedings of the Winter Conference on Applications of Computer Vision*, pp. 935–943, 2025.
- [62] M. Jacinto, J. Pinto, J. Patrikar, J. Keller, R. Cunha, S. Scherer, and A. Pascoal, “Pegasus simulator: An isaac sim framework for multiple aerial vehicles simulation,” *arXiv preprint arXiv:2307.05263*, 2023.
- [63] J. Song, H. Ma, O. Bagoren, A. V. Sethuraman, Y. Zhang, and K. A. Skinner, “Oceansim: A gpu-accelerated underwater robot perception simulation framework,” *arXiv preprint arXiv:2503.01074*, 2025.
- [64] J. Ou, T. Xu, X. Gan, X. He, Y. Li, J. Qu, W. Zhang, and C. Cai, “Comparative analysis on the effect of surface reflectance for laser 3d scanner calibrator,” *Micromachines*, vol. 13, no. 10, 2022.
- [65] P. Zapico, V. Meana, E. Cuesta, and S. Mateos, “Optical characterization of materials for precision reference spheres for use with structured light sensors,” *Materials*, vol. 16, no. 15, 2023.
- [66] P. H. Torr and A. Zisserman, “Mlesac: A new robust estimator with application to estimating image geometry,” *Computer vision and image understanding*, vol. 78, no. 1, pp. 138–156, 2000.
- [67] NVIDIA, “Camera sensors — isaac sim documentation.” [https://docs.isaacsim.omniverse.nvidia.com/4.5.0/sensors/isaacsim\\_sensors\\_camera.html#calibrated-camera-sensors](https://docs.isaacsim.omniverse.nvidia.com/4.5.0/sensors/isaacsim_sensors_camera.html#calibrated-camera-sensors), 2025. Accessed: 2025-07-25.
- [68] I. Martynov, J.-K. Kamarainen, and L. Lensu, “Projector calibration by “inverse camera calibration”,” in *Image Analysis: 17th Scandinavian Conference, SCIA 2011, Ystad, Sweden, May 2011. Proceedings 17*, pp. 536–544, Springer, 2011.
- [69] R. Hartley and A. Zisserman, “Camera models,” in *Multiple View Geometry in Computer Vision*, ch. 6, pp. 153–177, Cambridge, UK: Cambridge University Press, 2nd ed., 2003.
- [70] A. Lakshman, Y. Huang, W. Bussey, L. Liu, and B. Li, “Characterizing the 3-dimensional printability of alginate–gelatin and nanocellulose gels via fringe projection,” *Advanced Devices & Instrumentation*, vol. 6, p. 0116, 2025.