Supplementary Information for Realtime Glasses-Free 3D Display with Seamless Ultrawide Viewing Range using Deep Learning

Weijie Ma[1,2,3], Zhangrui Zhao[1,4], Wanli Ouyang[1,5*], Han-Sen Zhong[1,3*]

[1]Shanghai Artificial Intelligence Laboratory, Shanghai, China.
[2]School of Computer Science, Fudan University, Shanghai, China.
[3]Shanghai Innovation Institute, Shanghai, China.
[4]School of Computer Science and Engineering, Beihang University, Beijing, China.
[5]Department of Information Engineering, The Chinese University of Hong Kong, Hong Kong SAR & HKG, China.

Corresponding author(s). E-mail(s): wlouyang@ie.cuhk.edu.hk; zhonghansen@pjlab.org.cn;

Contents

List of Figures

List of Tables

1 Light-Field Display with Multi-Layer LCDs

This section provides a detailed description of the light-field display with multi-layer liquid crystal displays (LCDs).

1.1 Thin-film transistor liquid crystal display

In our system, we utilize Thin-Film Transistor Liquid Crystal Displays (TFT-LCDs), a sophisticated evolution of traditional LCD technology. TFT-LCDs are distinguished by their incorporation of thin-film transistors (TFTs), which enhance the display's performance by enabling precise control over individual pixels. The architecture of a TFT-LCD is composed of several critical components (Supplementary Fig. S1), including two glass substrates, a liquid crystal layer, a matrix of TFTs, pixel electrodes, color filters, common electrodes, polarizers, and a backlight system.

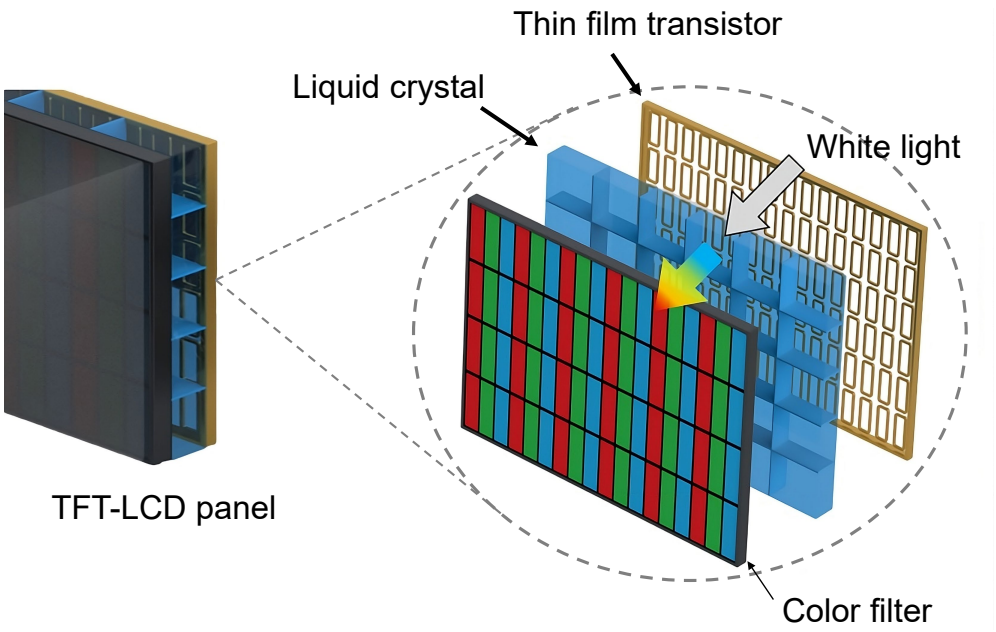
The two glass substrates serve as the foundational layers of the display, enclosing the liquid crystal layer. On one of these substrates, a matrix of TFTs and pixel electrodes is meticulously arranged. These TFTs function as switches, allowing for the application of exact voltages to the liquid crystal molecules within each pixel. This voltage control is crucial because it determines the orientation of the liquid crystal molecules, thereby modulating the amount of light that passes through each pixel and ultimately creating the desired image on the screen. On the opposite substrate, color filters and common electrodes are positioned. The color filters are configured to create red, green, and blue sub-pixels, which are the fundamental building blocks of the full color spectrum displayed on the screen. By combining these sub-pixels in varying intensities, the TFT-LCD can produce a wide range of colors, rendering images with high fidelity and vibrant hues. To manage the polarization of light as it enters and exits the display, polarizers are affixed to the outer surfaces of the glass substrates. These polarizers are essential for ensuring that the light passing through the liquid crystal layer is appropriately aligned, contributing to the display's contrast and clarity.

1.2 Optical mechanism and Malus' law

In TFT-LCD's liquid crystal layer, the orientation of the liquid crystal molecules controlled by the electric fields generated by the TFTs. By varying the applied voltage, the alignment of the liquid crystal molecules is adjusted, altering the polarization state of the light and thus the intensity of the light that exits the display. The operation of TFT-LCDs is deeply intertwined with Malus's law, which governs the intensity of polarized light passing through a LCD. According to Malus's law, the intensity of the transmitted light I is a function of the incident light intensity I_0 and the angle θ between the polarization direction of the light and the transmission axis of liquid crystal, which can be formulated by

$$I = I_0 \sin^2(\theta) \tag{1}$$

Here we throw a brief proof for the above equation based on Jones calculation [1]. The more details can be referred to some reviews [2–5]. A single LCD panel can be modeled as a polarization rotator, which is approximated by applying a linear rotation



Supplementary Fig. S1 Structure of a thin-film transistor liquid crystal display (TFT-LCD). This diagram illustrates the key components of a TFT-LCD, including the glass substrates, liquid crystal layer, TFT matrix, color filters, and backlight system. The arrows represent the path of light as it passes through the display: white light enters from the backlight, which is then modulated by the liquid crystal layer based on voltage control from the TFTs. As the light passes through the color filters, it is split into red, green, and blue components, which combine to form the full-color image displayed on the screen.

to the incident polarization state. The Jones matrix $R(\theta)$, representing a rotation by θ, is expressed as:

$$R(\theta) = \begin{pmatrix} \cos(\theta) & \sin(\theta) \\ -\sin(\theta) & \cos(\theta) \end{pmatrix} \tag{2}$$

Additionally, the Jones matrix $J(\xi)$, which corresponds to a counterclockwise rotation of an optical element by an angle ξ, is given by the transformation $R(-\xi)JR(\xi)$, where J is the Jones matrix of the unrotated element. Utilizing these principles, we derive the expression for the normalized intensity $I_{R1}(\phi, \xi)$ for a single polarization rotator enclosed by two linear polarizers, as a function of the polarization state rotation angle ϕ and the angle ξ between the axes of the front and rear polarizers:

$$I_{R1}(\phi, \xi) = I_0 \left\| \left(R(-\xi) \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix} R(\xi) \right) R(-\phi) \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right\|^2 = I_0 \cos^2(\phi - \xi) \tag{3}$$

Here the vector norm is l_2 norm. In this context, the rear polarizer, closest to the backlight, is assumed to be horizontally aligned, while the front polarizer is allowed to rotate freely. By convention, we assume the polarization rotator induces a counterclockwise rotation. For crossed polarizers such as a horizontal polarizer and a vertical polarizer (i.e., $\xi = \pi/2$), Equation 3 simplifies to

$$I_{R1}\left(\phi, \frac{\pi}{2}\right) = I_0 \sin^2(\phi) \tag{4}$$

thereby confirming Malus' law as illustrated in Equation 1.

1.3 Polarization field display based on phase aggregation

Then we apply the Jones calculus to analyze multilayer LCDs as multilayer polarization rotators. The Jones matrix $J_{RN}(\phi_1, \phi_2, \cdots, \phi_N)$, representing a composition of N polarization rotators, is given by:

$$\begin{aligned} J_{RN}(\phi_1, \phi_2, \cdots, \phi_N) &= R(-\phi_N)R(-\phi_{N-1}) \cdots R(-\phi_1) \\ &= \begin{pmatrix} \cos(\sum_{n=1}^{N} \phi_n & -\sin \sum_{n=1}^{N} \phi_n \\ \sin(\sum_{n=1}^{N} \phi_n & \cos \sum_{n=1}^{N} \phi_n \end{pmatrix} \end{aligned} \tag{5}$$

where $\{\phi_1, \phi_2, \cdots, \phi_N\}$ represent the incremental polarization state rotations induced at each layer. A comparison of Equation 5 with Equation 2 demonstrates that a sequence of polarization rotators effectively applies a counterclockwise rotation to the incident polarization state by an angle

$$\theta_{seq} = \sum_{n=1}^{N} \phi_n \tag{6}$$

Consequently, the normalized intensity $I_{RN}(\phi_1, \phi_2, \cdots, \phi_N)$ for a N-layer polarization rotator enclosed by crossed linear polarizers is expressed as:

$$I_{RN}(\phi_1, \phi_2, \cdots, \phi_N) = I_0 \left\| (0\ 1) \left(\prod_{n=1}^{N} R(-\phi_{N-n+1}) \right) \begin{pmatrix} 1 \\ 0 \end{pmatrix} \right\|^2 = I_0 \sin^2 \left(\sum_{n=1}^{N} \phi_n \right) \tag{7}$$

This confirms Equation 1 from the main text.

1.4 Focal parallax formulation in multi-layer LCDs and VAC-avoidance analysis

Focal parallax is a depth-related visual phenomenon that occurs when the eye or camera focuses on an object at a specific depth, causing objects at other depths to become blurred. This effect results from the way light rays from different depths interact with the imaging system. When the eye or lens focuses on one point, light rays from that point converge sharply onto the image plane or retina, rendering the point in sharp focus. However, light rays from objects at different depths do not converge in the same way, leading to their appearance as defocused or blurred, often seen as larger, soft spots. This is due to the varying angular divergences of light from these objects, which cause differential projections on the image plane. In optical systems, such as cameras or the human eye, this effect is more pronounced when viewed through a small aperture, where the focal point becomes distinct while points at other depths spread out as blur.

When viewing a 3D scene, the brain uses differences between the retinal images of each eye to infer the depth and spatial positions of objects, as illustrated in Supplementary Fig. S2a. Mathematically, consider two eye cameras placed in parallel, separated by a baseline distance B, with their optical axes aligned, as shown in Supplementary Fig. S2b. Each camera has a focal length f, and the pixel coordinates of a point in 3D space in the left and right image planes are denoted as (x_L, y_L) and (x_R, y_R), respectively. The disparity d is defined as the horizontal difference in the pixel coordinates of corresponding points in the left and right images:

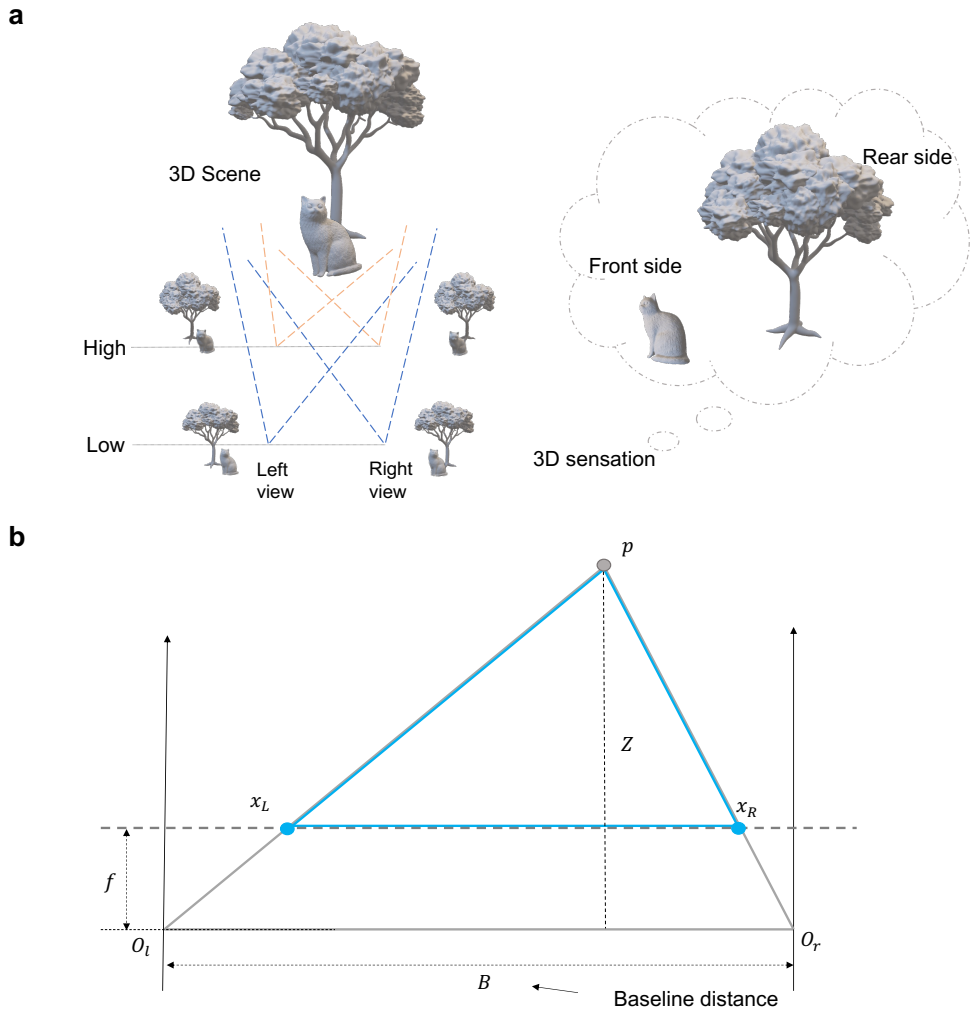
$$d = x_L - x_R \tag{8}$$

Given this disparity, the depth Z of the 3D point can be computed using the triangulation formula:

$$Z = \frac{f \cdot B}{d} \tag{9}$$

This equation shows that the depth Z is inversely proportional to the disparity d, where f is the focal length of the cameras and B is the baseline distance between the two cameras.

To enable the camera or human eyes to freely select the focus position, it is essential that light emitted from points at different depths in the displayed content exhibits varying degrees of divergence. In the context of light field displays, this requirement translates to the presence of continuous parallax in the light field entering the camera



Supplementary Fig. S2 Illustration of binocular parallax function. a, Stereo perception from binocular parallax. The left and right eye views of a 3D scene show both horizontal and vertical shifts for objects at different depths and heights. The brain combines these shifts from the left and right eye views to perceive depth and spatial arrangement. **b**, Schematic of binocular vision geometry for depth estimation. Two cameras are placed in parallel with a baseline distance B, with their optical axes aligned at points O_l and O_r. The 3D point p is projected onto the left and right image planes at coordinates x_L and x_R. The disparity d is the horizontal difference in pixel positions, and f is the focal length. The depth Z can be derived by similar triangle relation.

or eye. Continuous parallax ensures that each position across the lens or pupil receives a slightly different perspective of the scene, mimicking the way light behaves in the real world, as illustrated in Supplementary Fig. S3. This capability is fundamentally unattainable with discrete light field displays, which are restricted to producing light rays corresponding to a finite set of pre-defined views within a spatial domain. These systems are inherently limited to presenting a single focal depth, typically aligned with the central view, thereby precluding the possibility of true focal parallax.

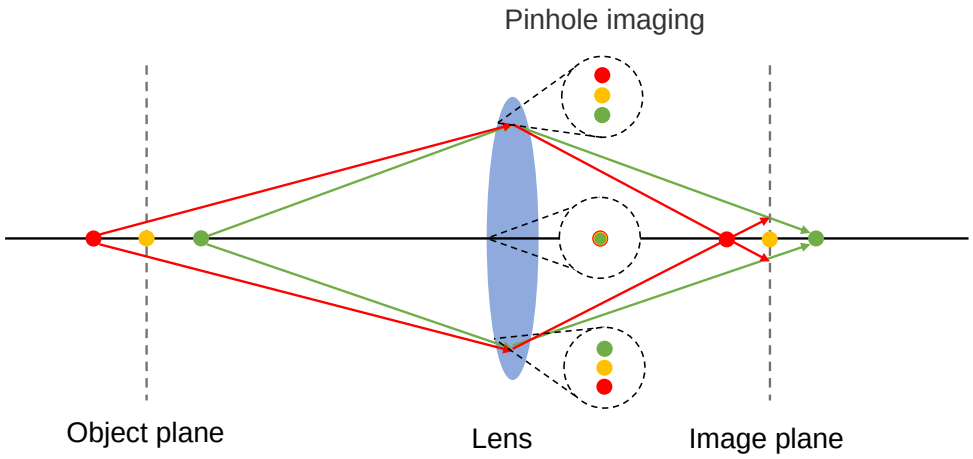
In contrast, our system achieves robust continuous parallax, enabling a realistic simulation of focal parallax by providing a seamless range of perspectives across the visual field. This not only facilitates natural focus adjustments but also addresses a critical limitation of traditional discrete light field displays: the vergence-accommodation conflict (VAC). VAC arises when the brain is forced to reconcile depth cues from the convergence of the eyes with conflicting focus cues, as all content in traditional displays appears on a single, fixed-depth focal plane. This misalignment between perceived and actual depth can lead to severe visual discomfort, including dizziness, headaches, and nausea, especially for physiologically sensitive viewers.

Our light field system overcomes these limitations by generating a continuous light field with natural, seamless focal parallax. This allows the human visual system to focus on multiple depth planes presented in the content, creating a more accurate and intuitive match between the brain's perception and the actual depth structure of the displayed scene. The presence of multiple, well-defined focal planes provides a high degree of perceptual tolerance, enabling the viewer to experience a vivid and realistic sense of 3D immersion. Moreover, this design effectively avoids VAC by accommodating the natural vergence and accommodation responses of the human visual system, thereby delivering a comfortable and convincing stereoscopic experience free from the adverse effects associated with traditional discrete systems.

2 Eye-Lightfield Standardization and Calibration

2.1 Eye coordinate system setup

The eye-camera model described in the main text is foundational for simulating the retinal imaging process using a light field coordinate system. This model provides a geometric framework to understand how the human eye perceives objects within a defined 3D space, replicating the actual position and gaze direction of the eye in the real world. The choice of a pinhole camera model stems from its simplicity and effectiveness in approximating the human eye's operation, mapping 3D points to 2D planes (retina) in an optically plausible manner. By aligning the center of the screen with the center of the light field and setting the eye's default gaze towards the origin, we create a standardized system that is easy to analyze and implement for simulating different viewing conditions. In the eye coordinate system model, the z-axis of the camera model is defined as opposite to the gaze direction, ensuring a consistent representation of the observer's viewpoint relative to the world. This approach mirrors real-world behavior, where the visual plane extends outward from the eye into the space it is observing. The x-axis parallel to the ground ensures that objects lying on a flat surface (e.g., a table or floor) are viewed similarly by both the eye and the camera model,



Supplementary Fig. S3 A schematic diagram illustrates the imaging process with focal parallax. When the focus point overlap with the yellow point, the green and red points become defocused, spreading into larger spots on the camera sensor. This phenomenon arises due to points with distinct depth having different perspective relationships. This effect can be observed when imaging these three points through a small aperture within the optical system.

thereby maintaining a realistic orientation of the visual scene. Additionally, setting the y-axis perpendicular to the plane formed by the z and x axes creates a right-handed coordinate system. This ensures that the 3D transformations applied to points and vectors within this system are consistent with common geometric conventions, making mathematical operations such as rotation and translation straightforward. Below is the mathematical derivation of the projection matrix. To convert from the eye-camera coordinate system to the light field coordinate system, the transformation can be expressed as a homogeneous matrix $\mathbf{M}_e = [\mathbf{R}_e|\mathbf{t}_e]$, where $\mathbf{R}_e$ represents the rotation matrix and $\mathbf{t}_e$ represents the translation vector. This matrix maps the eye's position and orientation within the light field, enabling the simulation of retinal imaging in the light field system. The rotation matrix $\mathbf{R}_e$ is derived based on three orthogonal vectors: $\mathbf{r}_x$, $\mathbf{r}_y$, and $\mathbf{r}_z$, which correspond to the x, y, and z axes of the eye camera coordinate system, respectively. The vector $\mathbf{r}_z$ is defined as the vector from the origin of the light field coordinate system to the eye position, representing the gaze direction. The vector $\mathbf{r}_x$ is perpendicular to $\mathbf{r}_z$ and lies in the plane parallel to the ground (i.e., the xOy plane of the light field coordinate system). This vector can be computed using the cross product of $\mathbf{r}_z$ and the z-axis vector of the light field system:

$$\begin{cases} \mathbf{r}_x \parallel xOy \\ \overrightarrow{Oz} \perp xOy \end{cases} \implies \mathbf{r}_x \perp \overrightarrow{Oz} \tag{10}$$

Based on the above, then

$$\begin{cases} \mathbf{r}_x \perp \overrightarrow{Oz} \\ \mathbf{r}_x \perp \mathbf{r}_z \end{cases} \implies \mathbf{r}_x = \overrightarrow{Oz} \times \mathbf{r}_z \tag{11}$$

Finally, the vector $\mathbf{r}_y$ is orthogonal to both $\mathbf{r}_z$ and $\mathbf{r}_x$, ensuring the creation of an orthonormal basis for the rotation matrix:

$$\begin{cases} \mathbf{r}_x \perp \mathbf{r}_y \\ \mathbf{r}_y \perp \mathbf{r}_z \end{cases} \implies \mathbf{r}_y = \mathbf{r}_z \times \mathbf{r}_x \tag{12}$$

These vectors are normalized to ensure unit length, resulting in the rotation matrix $\mathbf{R}_e$:

$$\mathbf{R}_e = \left[\frac{\mathbf{r}_x}{\|\mathbf{r}_x\|_2}, \frac{\mathbf{r}_y}{\|\mathbf{r}_y\|_2}, \frac{\mathbf{r}_z}{\|\mathbf{r}_z\|_2} \right]^T \tag{13}$$

Here $\|\cdot\|_p$ denotes the l_p vector norm applied on these trivial vectors. This matrix describes the orientation of the eye-camera system relative to the light field, enabling accurate projection of 3D points onto the 2D retinal plane. The translation vector $\mathbf{t}_e$ represents the position of the eye in the light field coordinate system. As mentioned earlier, this vector is simply the vector OP_e, which is equivalent to $\mathbf{r}_z$, the gaze direction:

$$\mathbf{t}_e = \overrightarrow{OP_e} = \mathbf{r}_z \tag{14}$$

Thus, the projection matrix $\mathbf{M}_e = [\mathbf{R}_e|\mathbf{t}_e]$ fully captures both the position and orientation of the eye relative to the light field, allowing for the simulation of the eye's perspective on the virtual scene.

2.2 Eye-lightfield least squares calibration

In this section, we provide a detailed explanation of the mathematical foundation and the step-by-step process involved in solving for the projection matrix $\mathbf{M}_c$ using least squares regression, as described in the main text. The goal is to minimize the alignment error between the captured camera coordinates and the corresponding light field world coordinates of calibration points, denoted as $\mathbf{c}_i$ and $\mathbf{w}_i$, respectively.

Least squares regression is a widely used optimization method for finding the best-fitting solution to an overdetermined system of linear equations. In our case, the objective is to solve for the rotation matrix $\mathbf{R}_c \in SO(3)$ and the translation vector $\mathbf{t}_c \in \mathbb{R}^3$ that define the projection matrix $\mathbf{M}_c = [\mathbf{R}_c|\mathbf{t}_c]$, such that the transformation from the camera coordinate system to the light field world coordinate system minimizes the discrepancy between the transformed camera coordinates and the true world coordinates of the calibration points. Given a set of K calibration pairs $\{(\mathbf{c}_i, \mathbf{w}_i)\}_K$, the least squares problem can be formulated as minimizing the Euclidean distance between the transformed camera coordinates $(\mathbf{R}_c\mathbf{c}_i^T + \mathbf{t}_c)$ and the corresponding world coordinates $\mathbf{w}_i^T$. Mathematically, this is expressed as:

$$\mathbf{R}_c, \mathbf{t}_c = \underset{\mathbf{R}_c \in SO(3),\ \mathbf{t}_c \in \mathbb{R}^3}{\arg\min} \sum_{i=1}^{N} \|(\mathbf{R}_c\mathbf{c}_i^T + \mathbf{t}_c) - \mathbf{w}_i^T\|_2^2 \tag{15}$$

where $\|\cdot\|_2$ denotes the l_2 norm, representing the Euclidean distance between the predicted and measured coordinates. This formulation allows us to quantify the error for each calibration pair and sum the squared errors across all pairs to form a single objective function. By minimizing this objective function, we can find the optimal projection matrix that minimizes the alignment error.

We model the transformation from the camera coordinate system to the light field coordinate system using a rigid body transformation. The transformation is described by the rotation matrix $\mathbf{R}_c \in SO(3)$, which captures the orientation, and the translation vector $\mathbf{t}_c \in \mathbb{R}^3$, which accounts for the relative displacement. For each calibration point $\mathbf{c}_i \in \mathbb{R}^3$ in the camera coordinate system, the corresponding world coordinates $\mathbf{w}_i \in \mathbb{R}^3$ can be approximated as:

$$\mathbf{w}_i^T \approx \mathbf{R}_c\mathbf{c}_i^T + \mathbf{t}_c \tag{16}$$

The error term for each calibration point is defined as the difference between the transformed camera coordinates and the true world coordinates, expressed as:

$$\mathbf{e}_i = (\mathbf{R}_c\mathbf{c}_i^T + \mathbf{t}_c) - \mathbf{w}_i^T \tag{17}$$

The goal is to minimize the sum of the squared errors over all calibration points:

$$\epsilon = \sum_{i=1}^{K} \|\mathbf{e}_i\|_2^2 = \sum_{i=1}^{N} \|(\mathbf{R}_c\mathbf{c}_i^T + \mathbf{t}_c) - \mathbf{w}_i^T\|_2^2 \tag{18}$$

To solve the least squares problem with scale estimation, we rewrite the transformation model in matrix form. We stack all the calibration points into matrices:

$$\begin{aligned} \mathbf{C} &= [\mathbf{c}_1, \mathbf{c}_2, \cdots, \mathbf{c}_K] \in \mathbb{R}^{K\times 3} \\ \mathbf{W} &= [\mathbf{w}_1, \mathbf{w}_2, \cdots, \mathbf{w}_K] \in \mathbb{R}^{K\times 3} \end{aligned} \tag{19}$$

Considering that the camera coordinate system and world coordinate system might be measured in different units or scales (e.g., camera coordinates in millimeters while world coordinates in meters), we introduce a scale factor s into the transformation model:

$$\epsilon = \|\mathbf{W} - (s\mathbf{C}\mathbf{R}_c^T + \mathbf{1}_K \mathbf{t}_c^T)\|_F^2 \tag{20}$$

where s is the scale factor, $\|\cdot\|_F$ represents the Frobenius norm, and $\mathbf{1}_K$ is a $K \times 1$ column vector of ones.

First, we compute the centroids of both point sets:

$$\overline{\mathbf{c}} = \frac{1}{K}\sum_{i=1}^{K} \mathbf{c}_i \ , \quad \overline{\mathbf{w}} = \frac{1}{K}\sum_{i=1}^{K} \mathbf{w}_i \tag{21}$$

To estimate the scale factor, we center both point sets and compute their respective average distances from their centroids:

$$\begin{aligned} \mathbf{C}' &= \mathbf{C} - \mathbf{1}_K \overline{\mathbf{c}}^T \\ \mathbf{W}' &= \mathbf{W} - \mathbf{1}_K \overline{\mathbf{w}}^T \\ s_c &= \sqrt{\frac{1}{K}\sum_{i=1}^{K} \|\mathbf{c}_i'\|^2} \\ s_w &= \sqrt{\frac{1}{K}\sum_{i=1}^{K} \|\mathbf{w}_i'\|^2} \\ s &= \frac{s_w}{s_c} \end{aligned} \tag{22}$$

The covariance matrix is computed using the scaled centered coordinates:

$$\mathbf{H} = (s\mathbf{C}')^T \mathbf{W}' \tag{23}$$

Using SVD:

$$\mathbf{H} = \mathbf{U}\Sigma\mathbf{V}^T \tag{24}$$

The optimal rotation matrix is computed with determinant check:

$$\mathbf{R}_c = \mathbf{V}\mathbf{S}\mathbf{U}^T \tag{25}$$

where $\mathbf{S}$ is a diagonal matrix with $\text{diag}(1, 1, \det(\mathbf{V}\mathbf{U}^T))$. The translation vector is then computed as:

$$\mathbf{t}_c = \overline{\mathbf{w}} - s\mathbf{R}_c\overline{\mathbf{c}} \tag{26}$$

Finally, we construct the transformation matrix that incorporates the scale factor:

$$\mathbf{M}_c = \begin{bmatrix} s\mathbf{R}_c & \mathbf{t}_c \\ \mathbf{0}^T & 1 \end{bmatrix} \tag{27}$$

This matrix is estimated to map any point from the camera coordinate system to the light field world coordinate system with different scales. By employing least squares regression, we minimize the calibration error and ensure that the transformation between the RGB-D sensor and the light field coordinate system is accurate and robust. This enables the precise alignment of captured eye coordinates with the light field display, enhancing the autostereoscopic effect for the viewer. Based on the solved projection matrix $\mathbf{M}_c$, we can obtain the position of the eye in the light field coordinate system $\mathbf{P}_e \in \mathbb{R}^3$ as below:

$$\begin{bmatrix} \mathbf{P}_e \\ 1 \end{bmatrix} = \mathbf{M}_c \begin{bmatrix} x_c \\ y_c \\ z_c \\ 1 \end{bmatrix} \tag{28}$$

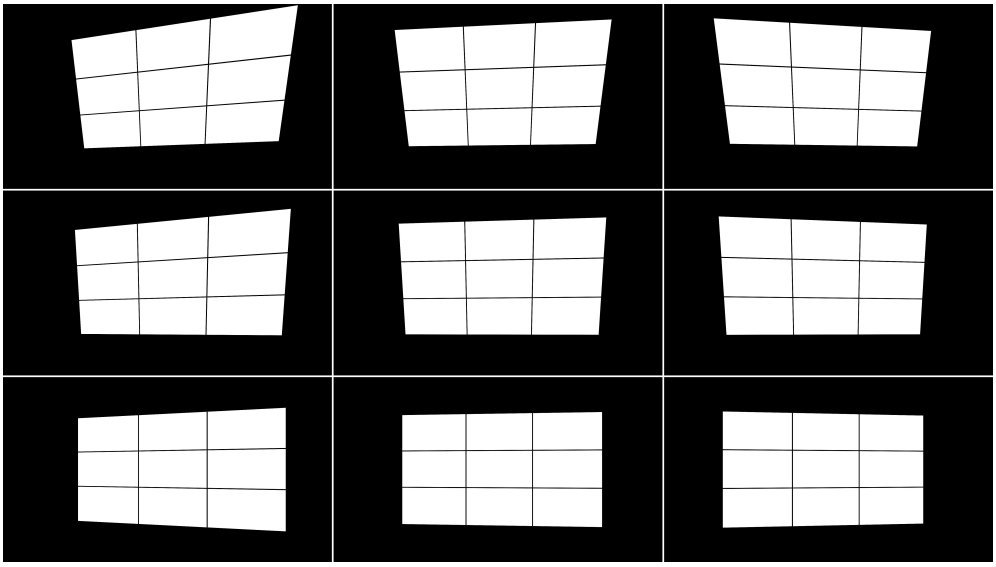
Here x_c, y_c, z_c represent the ocular coordinates in the RGB-D camera coordinate system.

2.3 Calibration experimental details and results

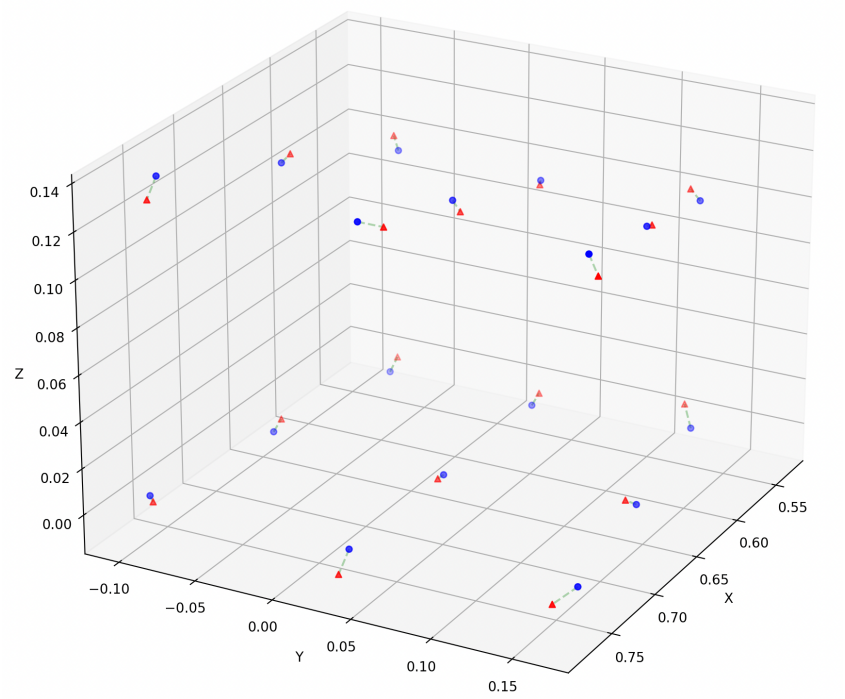
The calibration experiment was conducted to validate the theoretical framework for mapping eye-camera coordinates to light field world coordinates. To rigorously evaluate the proposed framework, synthetic data were generated using a physically accurate model to simulate retinal imaging under diverse geometric configurations. This approach ensured controlled conditions for assessing the robustness and accuracy of the coordinate transformation.

The experimental setup was designed to systematically explore the parameter space, with a focus on key geometric variables that influence the mapping process. Specifically, radial distances $R \in \{40, 50, 60\}$ cm were selected to represent the distance from the physical world viewpoint to the world center, while polar angles $\phi \in \{70°, 80°, 90°\}$ and azimuthal angles $\theta \in \{80°, 90°, 100°\}$ were chosen to define the orientation of the viewpoint relative to the z-axis and y-axis, respectively. These parameter ranges were carefully selected to cover a broad spectrum of potential viewpoints, ensuring comprehensive validation of the framework under varying observational conditions.

To facilitate the observer's judgment of the 3D effect, we designed a specialized calibration image. This image featured a 10-pixel black line drawn on a pure white background, positioned at the intersection of lines dividing the width and height into approximately three equal parts. As shown in Supplementary Fig. S4, the first layer of calibration charts was generated for $R = 40$ cm, with ϕ values of $70°, 80°, 90°$ and θ values of $80°, 90°, 100°$. Using the model, we obtained calibration charts corresponding



Supplementary Fig. S4 Calibration charts for viewpoint match. Here are the front-depth right-eye calibration charts generated for $R = 40$ cm, with polar angles $\phi \in \{70°, 80°, 90°\}$ and azimuthal angles $\theta \in \{80°, 90°, 100°\}$. These calibration charts are used to validate the correspondence between physical world coordinates $\mathbf{c}_i$ and pixel world coordinates $\mathbf{w}_i$. The calibration image consists of a 10-pixel black line on a pure white background, positioned at the intersection of lines dividing the width and height into approximately three equal parts. This design facilitates the observer's judgment of the 3D effect.



Supplementary Fig. S5 The visualization of calibration error. The plot illustrates the reprojection errors in both X, Y, and Z coordinates (in meters) across different calibration points. Here the errors consistently within acceptable limits for practical applications (<5cm), showcasing the simplicity but effectiveness of the proposed calibration method.

to different viewpoint coordinates. When these calibration charts were displayed on the screen, observers perceived a clear three-dimensional effect at the physical world coordinates $\mathbf{c}_i$.

Since the pixel world coordinates $\mathbf{w}_i$ of the viewpoint were known, we employed the least squares method to solve for the transformation matrix $\mathbf{M}_c$ from the physical world to the pixel world. This approach provided an approximate solution for $\mathbf{M}_c$. Despite the simplicity of the proposed calibration method, the results demonstrated remarkable accuracy. As shown in Supplementary Fig. S5, the reprojection errors were consistently low across all calibration points, and the angular deviations in the rotation component were negligible.

2.4 RGB-D sensor details and configuration

The RGB-D sensor utilized in our research is the Kinect for Windows v2 sensor (Kinect v2), developed by Microsoft as the second-generation Kinect device. Kinect v2 consists of a color camera, depth camera, and infrared (IR) emitter. These components work together to capture RGB (color), depth, and infrared data from a scene. The color camera of Kinect v2 features a resolution of 1920×1080 pixels, providing high fidelity and color accuracy. The field of view is relatively wide, with a horizontal angle of 70° and a vertical angle of 60°, enabling the device to cover larger areas in a single frame. Additionally, Kinect v2 supports high-speed data acquisition, capturing depth information at up to 60 frames per second. Compared to the original Kinect, the Kinect v2 offers three times the depth fidelity, allowing for more detailed representation of objects in the scene. The reduced motion blur and higher dynamic range in depth data further enhance its ability to capture accurate representations, particularly for moving objects.

Kinect v2's depth sensing mechanism is based on Time-of-Flight (ToF) technology. ToF systems operate by emitting modulated light, typically in the infrared (IR) spectrum, towards a scene. The emitted light interacts with the objects in the scene and reflects back to the sensor. The sensor then measures the time delay (or phase shift) between the emitted and reflected light. Since the speed of light is known, the time delay can be used to calculate the distance to each point in the scene. The basic principle of ToF can be expressed using the following formula:

$$d = \frac{c \cdot t}{2} \tag{29}$$

where d is the distance between the sensor and the object, c is the speed of light in air, approximately 3×10^8 meters per second, t is the round-trip time of the light (the time it takes for the light to travel from the sensor to the object and back). The factor of 2 in the denominator accounts for the fact that the measured time t includes the trip to the object and the return trip to the sensor.

In practical applications, however, the distance measurement is often performed using phase detection rather than directly measuring the travel time. The Kinect v2, for example, modulates the light source at an MHz-level frequency and detects the phase difference between the emitted and reflected light waves. This phase difference

is directly related to the distance of the object from the sensor, as described in the equation below:

$$d = \frac{c \cdot \phi_s}{2\pi f} \tag{30}$$

where ϕ_s is the phase shift between the emitted and received light, f is the modulation frequency of the light (in the case of Kinect v2, 80 MHz).

3 Details of Eye Camera Imaging and Warping

This section provides a detailed description of eye camera imaging and relevant perspective warping.

3.1 Preliminaries: Pinhole camera model and its imaging geometry

The human eye can be effectively modeled using a camera due to its optical similarities in terms of image formation. Both systems involve light entering through an aperture, being focused by a lens, and forming an image on a photosensitive surface. In the eye, the cornea and lens focus incoming light onto the retina, much like a camera lens focuses light onto a film or sensor. The retina functions similarly to a camera's sensor, converting light into electrical signals that the brain processes to produce visual perception. One particularly effective and simplified model of the eye is the pinhole camera model. This approximation works because, under generic conditions, the complex lens system of the eye can be abstracted to a single point through which light rays pass, forming an image. In essence, the pinhole camera model neglects the lens's refractive properties and instead focuses on the geometric path of light, which is sufficient for understanding basic image formation processes. This model is especially useful in situations where the precise focusing provided by the lens is less critical, or when analyzing the overall image formation without the complexities introduced by lens aberrations. The pinhole camera model assumes a small aperture (the pinhole) that allows light from different points of an object to project onto a surface (such as the retina in the eye or the image plane in a camera).

In the pinhole camera model, the imaging process can be described mathematically using a simple coordinate system. Assume that the pinhole is located at the origin of the coordinate system, with the image plane located at a distance behind the pinhole along the z-axis, where is the focal length, analogous to the distance from the pinhole to the imaging surface. To develop a comprehensive mathematical model for the process of projecting a three-dimensional world scene onto a two-dimensional image, we begin by adopting the camera's coordinate system as the reference frame. This choice allows us to establish the mathematical relationships governing the projection process with precision. The camera-centered reference system is defined as a right-handed coordinate system, where the origin of the coordinates is positioned at the camera's optical center, commonly referred to as the pinhole in the pinhole camera model. In this coordinate system, the z-axis corresponds to the camera's central optical axis, which is a straight line extending from the optical center and perpendicular to the image plane. In this framework, the image plane is mathematically described by the

equation $Z = f$, where f represents the focal length of the camera. As depicted in the left portion of Supplementary Fig. S6, we denote the camera coordinate system by $O - x - y - z$, where O indicates the optical center. Let $[X, Y, Z]^T$ represent the coordinates of a point P in the real world, and $[X', Y', Z']^T$ denote the coordinates of the corresponding image point P'. By applying the principle of similar triangles, illustrated in the right portion of Supplementary Fig. S6, we derive the following relationship:

$$\frac{Z}{f} = -\frac{X}{X'} = -\frac{Y}{Y'} \tag{31}$$

Here, the negative sign indicates the direction of the coordinate axis, signifying that the image appears inverted relative to the object. To simplify the analysis and eliminate the negative sign, we relocate the imaging plane from behind the camera to the front. With this adjustment, the relationship becomes:

$$\frac{Z}{f} = \frac{X}{X'} = \frac{Y}{Y'} \tag{32}$$

This leads to the solution for the coordinates of P', given by:

$$\begin{cases} X' = f\dfrac{X}{Z} \\ Y' = f\dfrac{Y}{Z} \end{cases} \tag{33}$$

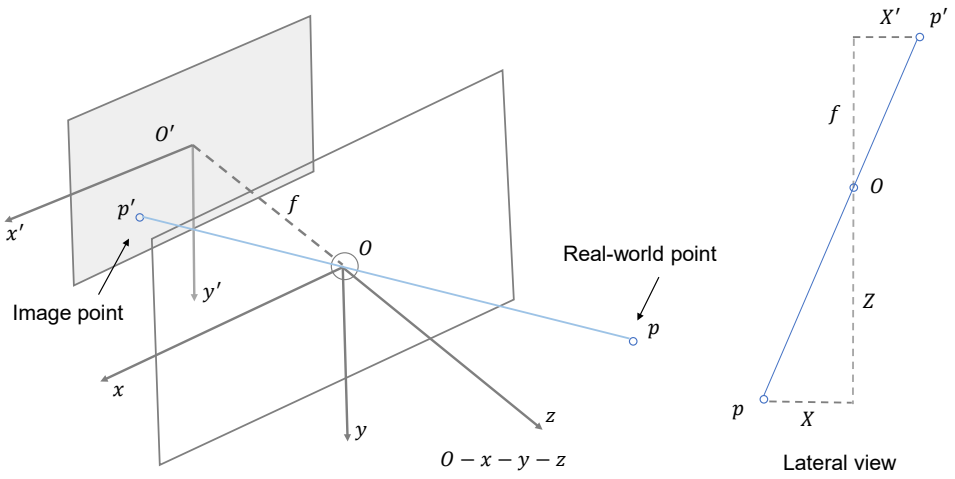
Next, we map the calculated coordinates on the image plane to the pixel coordinate system. The pixel plane, denoted by $O - u - v$, is fixed on the physical imaging plane. Assume that the pixel-level coordinates of P' are $[u, v]^T$. We further scale the coordinates by factors of α along the u-axis and β along the v-axis. Additionally, the origin of the pixel coordinate system is shifted by $[c_x, c_y]^T$. The relationship between the coordinates of P' and the pixel coordinates is then expressed as:

$$\begin{aligned} u &= \alpha X' + c_x \\ v &= \beta Y' + c_y \end{aligned} \tag{34}$$

By substituting the relationship between P and P' derived earlier, i.e.,

$$\begin{aligned} u &= \alpha f\frac{X'}{Z} + c_x = f_x\frac{X}{Z} + c_x \\ v &= \beta f\frac{Y'}{Z} + c_x = f_y\frac{Y}{Z} + c_y \end{aligned} \tag{35}$$

Here, we replace αf and βf with f_x and f_y respectively, where f_x and f_y represent the focal lengths in pixels along the u and v axes. Using homogeneous coordinates, we



Supplementary Fig. S6 The geometrical imaging process of pinhole camera model. The left panel depicts the 3D camera-centered coordinate system $O-x-y-z$, where the pinhole (optical center) is at O, and the image plane is located at $Z = f$ (focal length). A real-world point P is projected onto the image plane as p' via rays passing through O. The dashed lines show the projection geometry. The right panel provides a lateral view, showing the proportional relationship among p, p' and the focal plane through similar triangles.

can write the above equations in matrix form:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \frac{1}{Z} \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \frac{1}{Z}\mathbf{KP} \tag{36}$$

Z can also be written to the left of the above equation , which is transformed into:

$$Z \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \mathbf{KP} \tag{37}$$

In this equation, $\mathbf{K}$ represents the camera's intrinsic matrix, encapsulating the internal parameters of the camera.

To represent the transformation from any coordinate system to the image pixel coordinate system, we first transform the point P_ω in the world coordinate system to the corresponding point P in the camera coordinate system. This transformation is achieved through the camera pose, described by the rotation matrix $\mathbf{R}$ and the translation vector $\mathbf{t}$. These two components can be combined into a single transformation matrix $\mathbf{T} = [\mathbf{R}|\mathbf{t}]$, which belongs to the Special Euclidean group $SE(3)$. The $SE(3)$ group is defined as follows:

$$SE(3) = \left\{ \mathbf{A} \;\middle|\; \mathbf{A} = \begin{bmatrix} \mathbf{R} & \mathbf{r} \\ \mathbf{0}_{1\times 3} & 1 \end{bmatrix}, \mathbf{R} \in \mathbf{R}^{3\times 3}, \mathbf{r} \in \mathbf{R}^3, \mathbf{R}^T\mathbf{R} = \mathbf{R}\mathbf{R}^T = \mathbf{I}, |\mathbf{R}| = 1 \right\}$$

Here, $\mathbf{R}$ is a 3×3 orthogonal matrix representing the rotation of the camera, and $\mathbf{t}$ is a 3×1 vector representing the translation. The properties $\mathbf{R}^T\mathbf{R} = \mathbf{I}$ and $|\mathbf{R}| = 1$ ensure that the rotation matrix is valid, preserving the lengths and angles during transformation. The relationship between the point P_ω in the world coordinate system and the point P in the camera coordinate system can be expressed as:

$$\mathbf{P} = \mathbf{R}\mathbf{P}_\omega + \mathbf{t} \quad \Leftrightarrow \quad \mathbf{P}|_{homo} = \mathbf{T}\mathbf{P}_\omega|_{homo} \tag{38}$$

In this equation, P and P_ω denote the column vectors of points P and P_ω. $P|_homo$ and $P_\omega|_homo$ represent their homogeneous coordinates respectively. By substituting this relationship into the earlier derived equation for the imaging process, we obtain the complete formulation for mapping a point P_ω in the world coordinate system to its corresponding point in the pixel coordinate system:

$$Z\mathbf{P}_{uv}|_{homo} = \mathbf{K}(\mathbf{R}\mathbf{P}_\omega + \mathbf{t}) \tag{39}$$

3.2 Reverse perspective uniform for binocular pose encoding

In our system, we encode the pose information of binocular views using a Reverse Perspective Uniform (RPU). To achieve a unified encoding, the RPU employs reverse perspective transformation to standardize each image in binocular 6D poses onto the

screen plane at each depth as normalized warpings (Supplementary Fig. S7). The perspective transformation maps a three-dimensional scene onto a two-dimensional plane, simulating depth and spatial relationships within images. This transformation enables the image plane (viewing plane) to rotate about the perspective axis by a specific angle. This rotation modifies the original projection rays while preserving the geometric integrity of the shapes projected onto the image plane.

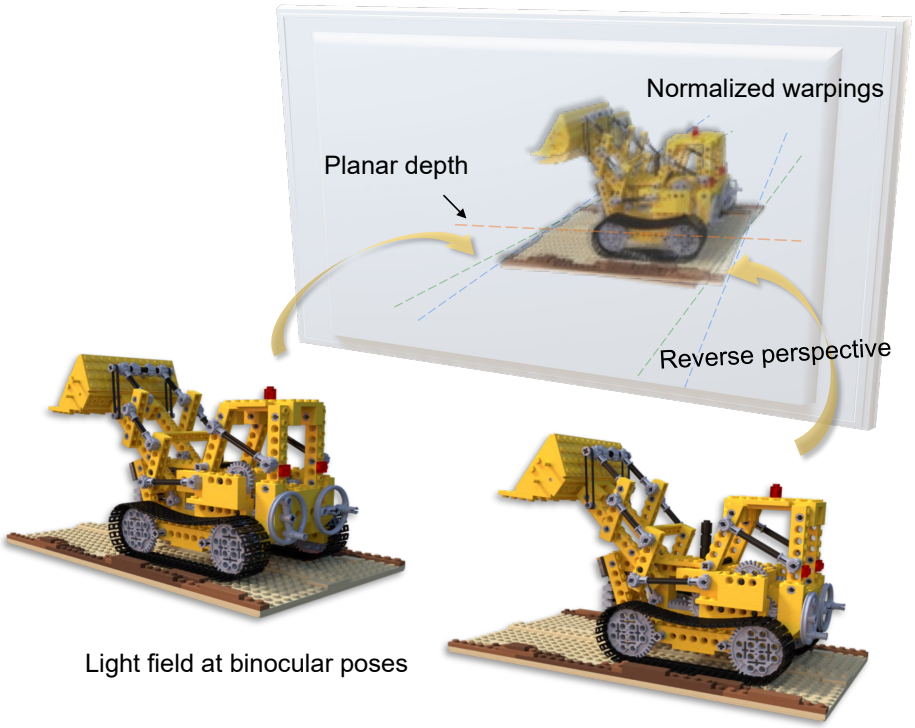
Mathematically, perspective transformation is typically represented by a homography matrix, a 3×3 matrix used to map points from the original image coordinates (j, k) to the transformed image coordinates (x, y). This transformation matrix can be decomposed into four components, each representing a different linear transformation, such as scaling, shearing, rotation, and translation, which collectively produce the perspective effect. These linear transformations can be considered special cases of perspective transformation, where the projection plane and the original plane are parallel, resulting in a simpler affine transformation. The general transformation equation for perspective transformation is expressed as:

$$\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} j \\ k \\ 1 \end{bmatrix} \tag{40}$$

Here, (j, k) are the coordinates in the original image, and (x', y') are the coordinates in the transformed image. The w' is a scale factor that accounts for the depth of the point relative to the camera. To obtain the final 2D coordinates (x, y), we normalize by dividing x' and y' by w':

$$\begin{cases} x = \dfrac{x'}{w'} \\ y = \dfrac{y'}{w'} \end{cases} \tag{41}$$

To determine the transformation matrix, it is necessary to have at least four corresponding point pairs between the original and the transformed images. Since the homography matrix has eight degrees of freedom (excluding the scale factor), these four point pairs provide eight linear equations, allowing the unique determination of the matrix elements. Consider the specific case of transforming a square into an arbitrary quadrilateral. Let the four corners of the square be mapped to four points on the quadrilateral. These points can be denoted as (j_1, k_1), (j_2, k_2), (j_3, k_3), and (j_4, k_4) in the original image, and (x_1, y_1), (x_2, y_2), (x_3, y_3), and (x_4, y_4) in the transformed



Supplementary Fig. S7 Illustration of the proposed Reverse Perspective Uniform (RPU) for encoding binocular pose information. The RPU applies reverse perspective transformations to project the binocular 6D poses onto the screen plane as normalized warpings. This transformation standardizes images from binocular camera coordinates onto the light field plane at a unified depth, preserving geometric integrity.

image. The transformation equations for these points are:

$$\begin{cases} x_1 = \dfrac{h_{11}j_1 + h_{12}k_1 + h13}{h_{31}j_1 + h_{32}k_1 + h33} \\ y_1 = \dfrac{h_{21}j_1 + h_{22}k_1 + h23}{h_{31}j_1 + h_{32}k_1 + h33} \\ \vdots \\ \vdots \\ x_4 = \dfrac{h_{11}j_4 + h_{12}k_4 + h13}{h_{31}j_4 + h_{32}k_4 + h33} \\ y_4 = \dfrac{h_{21}j_4 + h_{22}k_4 + h23}{h_{31}j_4 + h_{32}k_4 + h33} \end{cases} \tag{42}$$

In the light field display system discussed in the main text, the light field is oriented towards the eyes, thereby forming a retinal image. Here, the coordinates (j_i, k_i) correspond to the pixel coordinates on the light field plane, while the coordinates (x_i, y_i) correspond to the pixel coordinates on the eye camera. Specifically, (j_i, k_i) is denoted as $\mathbb{Q}_n := \{(0,0),(W,0),(W,H),(0,H)\}$ in the main text, while (x_i, y_i) is represented as $\mathbb{Q}'_n := \{(u'_i, v'_i)\}|_{i=1}^{4}$ in the main text. As aforementioned, here we exclude the scale factor, and set $h_{33} = 1$ for convenience. Substituting all these into Equation 42 yields:

$$\begin{aligned} u'_0 &= h_{13} \\ v'_0 &= h_{23} \\ u'_1 &= \frac{h_{11}W + h_{13}}{h_{31}W + 1} \\ v'_1 &= \frac{h_{11}W + h_{13}}{h_{31}W + 1} \\ u'_2 &= \frac{h_{11}W + h_{12}H + h_{13}}{h_{31}W + h_{32}H + 1} \\ v'_2 &= \frac{h_{21}W + h_{22}H + h_{23}}{h_{31}W + h_{32}H + 1} \\ u'_3 &= \frac{h_{12}H + h_{13}}{h_{32}H + 1} \\ v'_3 &= \frac{h_{22}H + h_{23}}{h_{32}H + 1} \end{aligned} \tag{43}$$

Rearranging the above equation gives:

$$\begin{aligned}
h_{13} &= u'_0 \\
h_{23} &= v'_0 \\
Wh_{11} + h_{13} - u'_1 h_{31} - u'_1 &= 0 \\
Wh_{21} + h_{23} - v'_1 h_{31} - v'_1 &= 0 \\
Wh_{11} + Hh_{12} + h_{13} - Wu'_2 h_{31} - Hu'_2 h_{32} - u'_2 &= 0 \\
Wh_{21} + Hh_{22} + h_{23} - Wv'_2 h_{32} - Hv'_2 h_{32} - v'_2 &= 0 \\
Hh_{12} + h_{13} - u'_3 h_{32} - u'_3 &= 0 \\
Hh_{22} + h_{23} - v'_3 h_{32} - v'_3 &= 0
\end{aligned} \tag{44}$$

To simplify the solution and make the relationships more compact, auxiliary variables can be introduced. With the auxiliary variables, the equations governing the transformation can be reduced, making it easier to understand and solve.

$$\begin{aligned}
\Delta u'_1 &= u'_1 - u'_2 \\
\Delta v'_1 &= v'_1 - v'_2 \\
\Delta u'_2 &= W(u'_3 - u'_2) \\
\Delta v'_2 &= H(v'_3 - v'_2) \\
\Delta u'_3 &= u'_0 - u'_1 + u'_2 - u'_3 \\
\Delta v'_3 &= v'_0 - v'_1 + v'_2 - v'_3
\end{aligned} \tag{45}$$

When these auxiliary variables are set to zero, the transformation plane becomes parallel to the original plane. This effectively reduces the transformation to an affine transformation, which is a simpler type that does not account for perspective effects, where the size of objects changes based on their distance from the viewer. When they are non-zero, the result is a full perspective transformation, and the size, shape, and position of objects in the transformed image will vary based on their relative position to the viewer. Continuing to substitute these auxiliary variables and simplify the above

equation can yield:

$$
\begin{aligned}
h_{11} &= \frac{u_1' - u_0' + h_{31}u_1'}{W} \\
h_{21} &= \frac{v_1' - v_0' + h_{31}v_1'}{W} \\
h_{31} &= \frac{\begin{vmatrix} \Delta u_3' & \Delta u_2' \\ \Delta v_3' & \Delta v_2' \end{vmatrix}}{\begin{vmatrix} \Delta u_1' & \Delta u_2' \\ \Delta v_1' & \Delta v_2' \end{vmatrix}} \\
h_{12} &= \frac{u_3' - u_0' + h_{32}u_3'}{H} \\
h_{22} &= \frac{v_3' - v_0' + h_{32}v_3'}{H} \\
h_{32} &= \frac{\begin{vmatrix} \Delta u_1' & \Delta u_3' \\ \Delta v_1' & \Delta v_3' \end{vmatrix}}{\begin{vmatrix} \Delta u_1' & \Delta u_2' \\ \Delta v_1' & \Delta v_2' \end{vmatrix}} \\
h_{13} &= u_0' \\
h_{23} &= v_0' \\
h_{33} &= 1
\end{aligned}
\tag{46}
$$

The above derives matrix elements of the forward perspective process. For the sake of binocular pose encoding, i.e., mapping the eye pose to the light field coordinate system, we can get the reverse perspective transformation matrix M_{RPU} from its inverse matrix given $\mathbb{Q}_n'$ of one eye:

$$
M_{RPU} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix}^{-1} \tag{47}
$$

4 Specific Settings of Lightfield Data Establishment

This section presents a comprehensive overview of the experimental configurations employed in establishing our light field datasets. As detailed in Supplementary Table S1 and Supplementary Table S2, we systematically document the critical parameters for each constructed light field dataset to ensure reproducibility and facilitate comparative analysis. The key specifications include: the scaling factor , which governs the transformation between physical world coordinates and digital light field representations; the compensating SE(3) transformation matrix, explicitly decomposed into its rotational and translational components to provide complete spatial transformation information; and the total depth in physical dimensions d_{thick}, which characterizes the volumetric extent of the scene.

LightField Dataset	r_{p2d}	d_{thick}(cm)
Metal Materials	0.03	12
MatrixCity	0.23	5
Desk & Bookshelf	0.03	6
Hot Dog	0.06	10
Orchids	0.04	100
Lego Bulldozer	0.08	6
Ficus	0.05	10
Chair	0.06	9.1
Traffic	0.04	8.75
Street	0.02	100
Minecraft	2.00	74.07
Lego Entity	0.08	11
Truck	0.08	6
Shoe Rack	0.08	12
Room Floor	0.02	46.7
Wukang Mansion	0.07	6
China Art Museum	0.03	25

Supplementary Table S1 Configuration parameters of light field datasets. The table presents the scaling factor r_{p2d} for physical-to-digital transformation and the corresponding depth d_{thick} in centimeters for each dataset.

LightField Dataset	Rotation	Translation
Metal Materials	[0, 0, 0]	[0, 0, 0]
MatrixCity	[0, 0, 0]	[0, -1.5, 0]
Desk & Bookshelf	[-114, 0, 90]	[4.8, -2.5, 0]
Hot Dog	[0, -92, 14]	[2.9, 0.04, 0.16]
Orchids	[-90, 0, 90]	[19.16, -1.78, 0]
Lego Bulldozer	[0, 0, 0]	[0, 0, 0]
Ficus	[0, 0, 0]	[0, 0, 0]
Chair	[0, 0, 0]	[0, 0, 0]
Traffic	[216, -1, -28]	[-0.36, 0, 2.54]
Street	[-116, 0, 114]	[2.56, 0, 1.46]
Minecraft	[0, 0, 0]	[0, 0, 0]
Lego Entity	[0, 0, 0]	[0, 0, 0]
Truck	[80, 182, -92]	[0.5, -0.24, 0.1]
Shoe Rack	[234, 0, 122]	[2.6, -0.24, 3.1]
Room Floor	[-20, -90, 86]	[0, 0.13, -3.07]
Wukang Mansion	[194, 23, -31]	[0.34, -0.02, 2.02]
China Art Museum	[234, 4, -61]	[0.54, 0.33, 0.72]

Supplementary Table S2 SE(3) transformation parameters for light field datasets. The table lists the rotation (in degrees) and translation (in meters) components of the compensating SE(3) matrix for each dataset.

References

[1] Clark Jones, R.: A new calculus for the treatment of optical systems. Journal of the Optical Society of America (1942)

[2] Collett, E.: Field guide to polarization (2005). Spie Bellingham

[3] Goodman, J.W.: Introduction to Fourier Optics. McGraw-Hill physical and quantum electronics series, (2005)

[4] Lanman, D., Wetzstein, G., Hirsch, M., Heidrich, W., Raskar, R.: Polarization fields: dynamic light field display using multi-layer lcds. In: Proceedings of the 2011 SIGGRAPH Asia Conference, pp. 1–10 (2011)

[5] Yeh, P., Gu, C.: Optics of liquid crystal displays **67** (2009)