**Appendix 3:** The method for generating simulated image sequences.

In Figure 2(b), the set of image points {*Pk*|*k*=1→*K*} represents the center points of blurred circles with varying diameters. These correspond to a set of pixels in the image sequence {*Ik*|*k*=1→*K*} that are aligned along the optical axis and positioned accordingly. The set of their grayscale values is denoted as {*fk*}, where the image frame with the most focused pixel has index *k*=*p*, with . *Pp*​ represents the most focused pixel, with a grayscale value of *fp*​, and its corresponding *Z*-axis coordinate is *Zp*​. The set {*fk*} is simulated and generated using a method divided into three modules: the first module initializes *p*, *Zp*​, and *fp*​; the second module generates {*fk*}; and the third module creates the simulated image sequence {*Ik*}.

***Module* 1:** The initialization method for *p*, *Zp*​, and *fp*​:

*Step* 1: Create a surface model and discretize it. In the software platform, define a virtual coordinate system *O-XYZ*, in which the three-dimensional surface model *U* is constructed. The dimensions of the minimum circumscribed cuboid of *U* are represented as *L*×*W*×*H*. The circumscribed cuboid space is divided into *M*×*N*×*K* small cuboid grids, with the set of center points of these grids denoted as {*Qk*(*m*, *n*)| *m*=1→*M*, *n*=1→*N*, *k*=1→*K*}. The set {*Qk*(*m*, *n*)| *k*=1→*K*} represents the trajectory of the object points along the *Z*-axis, while *M*×*N* corresponds to the resolution of a single image frame.

*Step* 2: The values of *p*, *Zp*, and *fp*. If the trajectory of the object point *G* on the surface model is {*Qk*}, the corresponding set of *Z*-axis coordinates is {*Zk*}, then in *Step* 1, the index *p* of the cuboid grid containing point *G* and its corresponding *Z*-axis coordinate *Zp*​ can be determined. The initial grayscale value *fp* of the focused pixel corresponding to point *G* can then be expressed as:

*fp*=*g*1×(*p*-1)/(*K-*1)+ *g*0 (A3-1)

Where *g*0 represents the desired minimum grayscale value, while *g*1 indicates the maximum grayscale difference in the image. The grayscale range of the simulated image is [*g*0, *g*0 + *g*1].

*Step* 3: Simulate the all-focus image *I*0. The set {*Ik*} contains *M*×*N* focused pixels, which construct a frame of the all-focus image *I*0. Formula (A3-1) is used to calculate and simulate the grayscale values of the pixels in *I*0, which are then utilized to determine the grayscale values of the blurred pixels in ***Module* 2**.

***Module* 2:** The calculation of {*fk*}

*Step* 1: The calculation of the radius {*rk*} of the blurred circle corresponding to {*Pk*}. The radius of the blurred circle, *rk*​, can be estimated using the thin lens model as follows:

(A3-2)

Where *fo*​ is the focal length of the optical system, and *F* is the aperture factor.

*Step* 2: Calculate the convolution window radius {}​ corresponding to the parameter set {*Pk*}​. The radius ​ is calculated by *rk*​, and the formula is as follows:

(A3-3)

Where *a* is a tuning parameter, for example, *a*=0.5. ​ determines the size of the convolution window, which affects the degree of pixel blurring.

*Step* 3: Calculate the blur kernel {}. ***hk***​ represents the blur kernel used in the calculation of *fk*​. It is a matrix with dimensions , and its elements are calculated using the Gaussian function:

(A3-4)

Where σ represents the standard deviation.

*Step* 4: Compute {*fk*} based on the Gaussian blur principle. {*fk*} is given by the following equation:

(A3-5)

Where represents the convolution operation, and ​ is a sub-image taken from *I*0​, with its center pixel at *Pp*​, and its size matches that of ***hk***​.

***Module* 3:** Generate Image Sequence {*Ik*}

The grayscale simulation method from ***Module 2*** applied to the single-pixel sequence {*Pk*} is extended to all pixel sequences to generate the simulated image sequence {*Ik*}. To minimize the discrepancy between the simulated and real image sequences, a certain proportion of Gaussian noise is added to {*Ik*}, thus producing a more realistic simulated image sequence.