

Article

# Automated Calibration Method for Eye-Trackable Autostereoscopic Display

Hyoseok Hwang 

Department of Software, Gachon University, Seongnam, Gyeonggi-do 13120, Korea; hshwang@gachon.ac.kr;  
Tel.: +82-31-750-8920

Received: 27 July 2018; Accepted: 7 August 2018; Published: 9 August 2018



**Abstract:** In this paper, we propose an automated calibration system for an eye-trackable autostereoscopic display (ETAD). Instead of calibrating each device sequentially and individually, our method calibrates all parameters of the devices at the same time in a fixed environment. To achieve this, we first identify and classify all parameters by establishing a physical model of the ETAD and describe a rendering method based on a viewer's eye position. Then, we propose a calibration method that estimates all parameters at the same time using two images. To automate the proposed method, we use a calibration module of our own design. Consequently, the calibration process is performed by analyzing two images captured by onboard camera of the ETAD and the external camera of the calibration module. For validation, we conducted two types of experiments, one with simulation for quantitative evaluation, and the other with a real prototype ETAD device for qualitative assessment. Experimental results demonstrate the crosstalk of the ETAD was improved to 8.32%. The visual quality was also improved to 30.44% in the peak-signal-to-noise ratio (PSNR) and 40.14% in the structural similarity (SSIM) indexes when the proposed calibration method is applied. The whole calibration process was carried out within 1.5 s without any external manipulation.

**Keywords:** autostereoscopic display; automated calibration; camera calibration; eye-tracking

## 1. Introduction

An autostereoscopic display is a device that allows three-dimensional (3D) perception by providing different images to each one of a person's eyes. The biggest advantage of the autostereoscopic display is that users do not need to wear special equipment such as glasses. During the past decade, various methods have been studied to implement autostereoscopic display [1,2]. Among them, an autostereoscopic display with a conventional two-dimensional (2D) flat panel display and an additional optical layer, which is located in front or at the back of the display panel, has gained the most commercial success due to its low cost and ease of fabrication [3]. A typical method of stereoscopic display, which employs an additional optical layer, endows directionality to each pixel by blocking light paths (for parallax barriers) or by refracting light rays (for lenticular lenses). This method limits the visibility of pixels to only a specific range of view positions or angles. Located at different view positions, the left and the right eye see different sets of pixels.

Despite these advantages, several drawbacks prevent the autostereoscopic display from becoming more prevalent [4,5]. A typical method of stereoscopic display, in which the left and the right views are repeated alternately, is restricted to a specific viewing area. In the case of two views, the margin of the correct viewing area is approximately the distance between the eyes. Even if a viewer moves slightly from the fixed location, the view will transform into a depth-reversed image, referred to as a pseudoscopic image [6]. Increasing the angular resolution, i.e., the number of views, has been a promising method to overcome this problem [7]. The concept of the multiview method is based on showing the user two of several consecutive views, which have disparity over each other, instead of

the left and the right views. In this case, however, the spatial resolution of each view is inevitably lowered due to the finite number of pixels. Hence, a tradeoff arises between the angular resolution and the spatial resolution. In addition, complete separation between all views is challenging, so there is a crosstalk where several views overlap with each other.

To solve these problems, an eye-tracked autostereoscopic display (ETAD) has been proposed [8–11]. Instead of increasing the number of views, the ETAD uses the left and right views only. To prevent the user's eyes from moving out of the correct viewing area, the ETAD employs an additional device, e.g., a camera or a depth sensor, for detecting and tracking the eye position of a viewer. Using the eye tracking device, the ETAD ensures that the user's eyes are always in the correct viewing zones by moving the optical element mechanically [12], by moving slits on an additional LC display [13,14], or by merging viewing zones into two views around the viewer's eyes [6].

To date, there has been various studies to improve the visual quality of the ETAD, which mainly focus on minimizing crosstalk [4], reducing visual fatigue [15], or estimating the accurate position of a viewer [16]. These studies implicitly assumed that all parameters of the ETAD are known accurately or are the same as the designed parameters. In practice, however, parameters of the ETAD are likely to differ from the designed values for many reasons, especially due to inaccuracy in the production and the assembly processes. Moreover, the ETAD system, which employs additional devices and processes, requires parameters that are more accurate because the accumulation of errors from each device significantly degrades the visual quality. Therefore, parameters of the ETAD must be calibrated prior to the operation.

Several studies have investigated the calibration of the ETAD. Sandin et al. [17] described a calibration method of Varrier<sup>TM</sup>, which consists of multiple autostereoscopic displays with a stereo camera for user's head tracking. To calibrate each optical element of the displays, they used external stereo cameras. However, they did not describe the method for calibrating the pose between the head tracking camera and the coordinates of each device. Bailer et al. [18] proposed a calibration method for a moving barrier type ETAD system. In the calibration process, they mapped each pixel on a captured camera image to the pixels of a display panel. This method is simple to implement; however, the calibration process should be conducted in a dark room. Moreover, the mapping pixels between the image and the panel are only valid when the viewer is at a certain distance. In the case of Nintendo<sup>TM</sup> 3DS, which is commercially available, the calibration is carried out by the user [19]. A user continuously changes parameters while looking at the screen until the optimum visual quality is achieved.

Thus far, there has been no method to calibrate all parameters of the devices consisting the ETAD. In calibrating the ETAD, therefore, the most suitable method to date is to calibrate each device individually and sequentially. Below, we present a calibration example for an ETAD consisting of an onboard camera for eye tracking, a 2D display panel, and an optical element such as lenticular lenses or parallax barriers. First, we calibrated an onboard camera by analyzing several images, which capture patterns of various postures in front of the camera [20]. Then, we calibrated the optical element by the visual pattern analysis method [21]. This method uses two images of the display, thus requiring an external camera setup. Finally, the extrinsic pose between the onboard camera and the display panel should be calibrated. There are few studies on calibrating the extrinsic pose of these two devices. The second-best method is to apply a similar approach as in [22], in which the extrinsic pose of the camera and the pattern on a robot are calibrated. Even though we employed this method, problems still exist. To calibrate the pose of the camera and the pattern looking in the same direction, a mirror is needed. Furthermore, several movements of the device are required to capture images at various angles. Applying this approach to the ETAD can lead to a serious problem: parameters of the device may be changed by repeated movements of the device. In addition, individual calibration methods have a serious problem. The calibration errors of each device are accumulated and finally affect to the visual quality. The sequential approach described above is also not suitable for automated calibration. For automated calibration, an auxiliary manipulator is needed to move calibration patterns or the

ETAD itself. In terms of mass production, this method increases both cost and processing time of the device calibration.

For automated calibration of the ETAD, several problems should be solved. It has not been clearly established which parameters of the ETAD need to be calibrated. A novel methodology would be to calibrate all of these parameters not individually but simultaneously for cost reduction. To alleviate parameter change during the calibration process, the method should minimize manipulation or even calibrate in a fixed environment. In this paper, we introduce an automated calibration method for the ETAD. The main contributions of this study are as follows:

- We establish the parameters that require calibration. To achieve this, we firstly address the process of the ETAD from eye tracking to 3D image rendering. To the best of our knowledge, the whole process of the ETAD has never been explored before.
- We propose a novel methodology, which calibrates all devices of the ETAD. Our calibration approach is mainly focused on both the automated and the simultaneous calibration. This method calibrates all parameters at the same time, not sequentially. The proposed method also does not require the movement of the ETAD or the calibration device, but instead is performed in a fixed environment. To achieve this goal, we designed and implemented a calibration module that consists of a 3D pattern and an external camera. The module enabled us to calibrate an ETAD by simply setting it in front of the calibration module.

The rest of the paper is organized as follows. In Section 2, we establish the parameters of the ETAD by addressing its physical model and rendering process. Section 3 presents a calibration method, which estimates all parameters automatically and simultaneously. Then, we experimentally validate the proposed method in Section 4. Finally, Section 5 concludes the paper.

## 2. Parameters Establishment

In this section, we establish the parameters of the ETAD that need to be calibrated in two stages. First, we describe the parameters of the devices that comprise a conventional ETAD. Here, the conventional ETAD refers to an autostereoscopic device having a camera for eye tracking and an additional optical layer, e.g., lenticular lenses or parallax barriers, on a 2D flat panel. Note that parameters are common even though the types of optical layers are different. Then, we describe the rendering process of the ETAD in order to identify the parameters.

### 2.1. Physical Model of the ETAD

The parameters of the conventional ETAD devices are represented in Figure 1. The conventional ETAD consists of three modules: a two-dimensional flat panel, an optical layer, and an eye-tracking sensor. The two-dimensional flat panel is a collection of pixels and each pixel has a width  $\mu_w$  and height  $\mu_h$ . Most of optical layers are lenticular lenses or parallax barriers [23]. In addition, these two optical layers have common parameters. A pitch  $p$  represents the distance between the lens in the case of lenticular lenses and between the slits in the case of parallax barriers. The slanted angle  $\theta$  represents the degree of tilting of the optical layer. The gap  $\tau$  is the distance of the slit of the barrier or the focal plane of the lens from the flat panel. The offset parameter  $\eta$  represents the horizontal translation of the optical layer from the origin of the display coordinate system. An onboard camera is attached at the top of the display panel for tracking a viewer's eyes. To reconstruct the three-dimensional position from a point on an image, intrinsic parameters and distortion parameters of the camera are used. The intrinsic parameters include focal lengths  $f_x, f_y$  and principal points  $c_x, c_y$ . The parameters  $k_1, k_2, k_3$ , and  $k_4$  denote distortion coefficients. There are two coordinate systems: the onboard camera coordinate system and the display panel coordinate system.

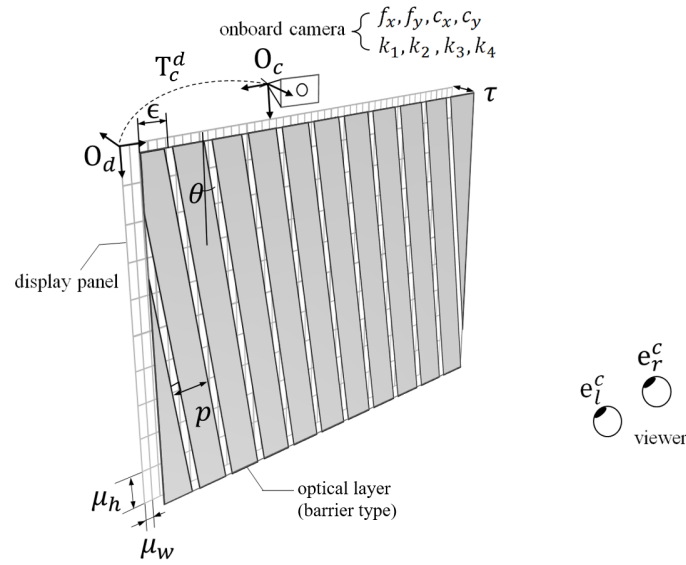


Figure 1. Parameters of an Eye-Tracked Autostereoscopic Display (ETAD).

2.2. Rendering Process

Basically, the rendering process for the ETAD is considered as an iterative process comprising two steps: the eye position estimation step and the image-multiplexing step. In the eye position estimation step, the system estimates the 3D position of the eyes from the image captured by the onboard camera. Note that the eye detection method is beyond the scope of this study. In the image-multiplexing step, the system generates a multiplexing image according to the eye positions. Let us describe the iterative process systematically.

**Eye Position Estimation Step:** Given the eye positions on a captured image from the onboard camera, they are derived in the camera coordinate system as

$$\begin{bmatrix} x_\mu \\ y_\mu \end{bmatrix} = \begin{bmatrix} \frac{s_\mu - c_x}{f_x} \\ \frac{t_\mu - c_y}{f_y} \end{bmatrix}, \tag{1}$$

where  $s_\mu$  and  $t_\mu$  are eye positions on the captured image and  $x_\mu, y_\mu$  are the corresponding positions on the onboard camera coordinate system. Note that the subscript  $\mu$  is either left or right ( $\mu \in \{l, r\}$ ) and the equation containing  $\mu$  is the same in the left and right cases. The parameters  $f_x, f_y$  are focal lengths in the  $x$  and  $y$  directions, respectively, and  $c_x, c_y$  are principal points of the onboard camera. Commercial camera lenses conventionally have distortion, known as radial and tangential distortion [24]. We define  $k_1, k_2$  as the parameters for radial distortion, and  $k_3, k_4$  as the parameters for tangential distortion of the camera lens. Using these distortion parameters, the undistorted position of each eye  $x'_\mu, y'_\mu$  is derived as

$$\begin{bmatrix} x'_\mu \\ y'_\mu \end{bmatrix} = (1 + k_1 r^2 + k_2 r^4) \begin{bmatrix} x_\mu \\ y_\mu \end{bmatrix} + \begin{bmatrix} k_3(2x_\mu y_\mu) + k_4(r^2 + 2x_\mu^2) \\ k_3(r^2 + 2y_\mu^2) + k_4(2x_\mu y_\mu) \end{bmatrix}, \tag{2}$$

where  $r_\mu = \sqrt{x_\mu^2 + y_\mu^2}$ . Then, we can obtain the ray vectors, which have direction information of both eyes from the origin of the camera coordinate system. The ray vector  $\mathbf{v}_\mu$  which passes through  $x'_\mu, y'_\mu$  from the origin of the camera coordinate system  $\mathbf{O}_c$  can be represented in homogeneous coordinates as

$$\mathbf{v}_\mu = \begin{bmatrix} x'_\mu \\ y'_\mu \\ 1 \end{bmatrix}. \quad (3)$$

Given these back-projected vectors, we finally obtain the three dimensional eye positions  $\mathbf{e}_l^c, \mathbf{e}_r^c$  as

$$\mathbf{e}_l^c = \frac{\psi}{\|\mathbf{v}_l - \mathbf{v}_r/\alpha\|} \mathbf{v}_l, \quad (4)$$

$$\mathbf{e}_r^c = \frac{\psi}{\|\alpha\mathbf{v}_l - \mathbf{v}_r\|} \mathbf{v}_r, \quad (5)$$

where  $\alpha$  is defined as

$$\alpha = \frac{\mathbf{v}_r^T \phi}{\mathbf{v}_l^T \phi}, \quad (6)$$

Here, we assume that interpupillary distance (IPD)  $\psi$  is known and the direction of the face normal vector  $\phi$  faces the onboard camera. The proof for Equations (4) and (5) are given in Appendix A at the end of the paper.

**Image Multiplexing Step:** In our rendering approach, we can simplify the method of generating a multiplex image as a problem of assigning labels (left or right) to each pixel. Note, a pixel generally consists of several sub-pixels. However, here we define the pixels in the smallest unit, i.e., sub-pixels, as “pixels”. The multiplexed image is generated from the number of viewpoint images, e.g., a two-view display requires two images: the left and the right image. As described above, all pixels have directionality by the additional optical layer. Pixels are assigned the “left” label to show the same intensity value of the left source image if their direction is toward the left eye of the viewer; the same condition is applicable to the “right” case. On the contrary, we can assign a label to each pixel by determining whether the pixel’s position is close to the visible pixels of the left eye or those of the right eye. The slits of the barrier or principle points of a lenticular lens can be expressed as a set of lines; therefore, the pixels passing through the slit and directed to the left or right eye are also represented by lines. Here, we would like to define these sets of lines as the visible pixels of the left eyes and those of the right eyes. Then, we assign a label value that corresponds to the nearest visible line for the all pixels, not visible directly.

The 3D eye position  $\mathbf{e}_\mu^c$  obtained from the captured image is based on the camera coordinate system; we can convert it into the display coordinate system as

$$\begin{bmatrix} \mathbf{e}_\mu^d \\ 1 \end{bmatrix} = \mathbf{T}_c^d \begin{bmatrix} \mathbf{e}_\mu^c \\ 1 \end{bmatrix} \quad (7)$$

where  $\mathbf{T}_c^d$  is a homogeneous transform matrix from an onboard camera coordinate system to a display coordinate system. The transform matrix can be represented as

$$\mathbf{T}_c^d = \begin{bmatrix} \mathbf{R}_c^d & \mathbf{t}_c^d \\ 0 & 1 \end{bmatrix}, \quad (8)$$

which consisting of the transform matrix  $\mathbf{t}_c^d = [t_x \ t_y \ t_z]^\top$  and the rotation matrix  $\mathbf{R}_c^d$ . The rotation matrix is defined as

$$\mathbf{R}_c^d = \begin{bmatrix} \cos r_z \cos r_y & \cos r_z \sin r_y \sin r_x - \sin r_z \cos r_x & \cos r_y \sin r_y \cos r_x + \sin r_z \sin r_x \\ \sin r_z \cos r_y & \sin r_z \sin r_y \sin r_x + \cos r_z \cos r_x & \sin r_x \sin r_y \cos r_z - \cos r_z \cos r_x \\ -\sin r_y & \cos r_y \sin r_x & \cos r_y \cos r_x \end{bmatrix}, \quad (9)$$

where  $r_x, r_y, r_z$  are Euler angles in radians.

We define the  $n$ th line of the slits of an optical element as  $\mathbf{S}_n$ ; the vector equation is represented as

$$\mathbf{S}_n = \begin{bmatrix} \epsilon + \frac{p}{\cos \theta} n \\ 0 \\ \tau \end{bmatrix} + \delta \begin{bmatrix} \tan \theta \\ 1 \\ 0 \end{bmatrix}, \quad (10)$$

where  $\epsilon$  is the offset,  $\theta$  is the slanted angle,  $p/\cos \theta$  is the horizontal pitch, and  $\tau$  is the gap of the optical layer. The left term of this equation represents the origins of slits, and the right term the directional vector with an arbitrary real number  $\delta$ . We obtain the positions of the visible pixel  $\mathbf{s}'_{n,\mu}$  which is represented as the projected vectors on the panel using the eye positions and the slit vectors as

$$\mathbf{s}'_{n,\mu} = \frac{1}{z_\mu - \tau} \left( z_\mu \begin{bmatrix} \epsilon + \frac{p}{\cos \theta} n \\ 0 \\ \tau \end{bmatrix} - \tau \mathbf{e}_\mu^d \right) + \delta \begin{bmatrix} \tan \theta \\ 1 \\ 0 \end{bmatrix}, \quad (11)$$

where  $z_\mu (\mu \in l, r)$  is the distance to the left or the right eye. Then, we also express a pixel position of  $u, v$  on the display panel as

$$\mathbf{P}_{u,v} = \begin{bmatrix} u\lambda_w \\ v\lambda_h \\ 0 \end{bmatrix}, \quad (12)$$

where  $\lambda_w, \lambda_h$  are the width and height of a pixel, respectively.

We obtain the distances between a pixel  $\mathbf{P}$  and all visible pixels  $\mathbf{s}'_{n,\mu}$  by the function  $D$ , which is given by

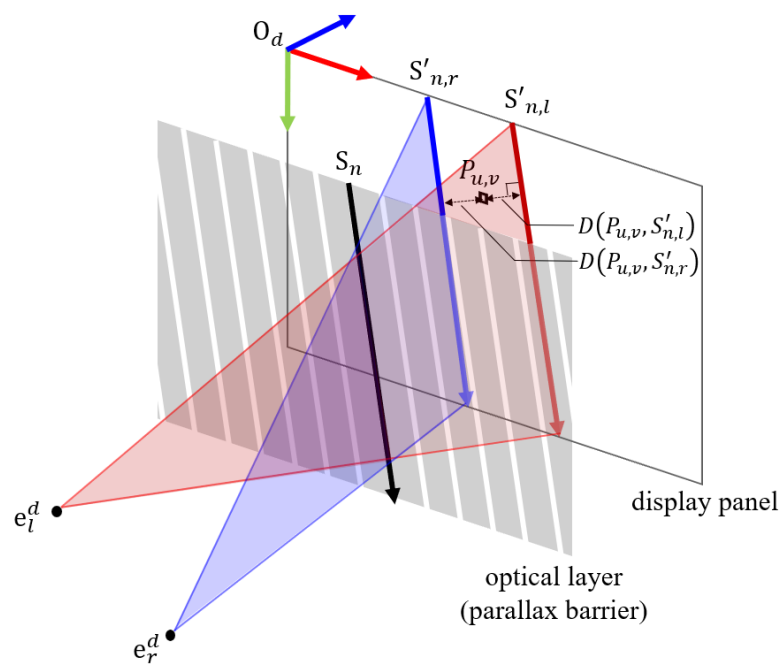
$$D(\mathbf{P}_{u,v}, \mathbf{s}'_{n,\mu}) = \frac{\left( \mathbf{P}_{u,v} - \frac{1}{z_\mu - \tau} \left( z_\mu \begin{bmatrix} \epsilon + \frac{p}{\cos \theta} n \\ 0 \\ \tau \end{bmatrix} - \tau \mathbf{e}_\mu^d \right) \right) \times \begin{bmatrix} \tan \theta \\ 1 \\ 0 \end{bmatrix}}{\left\| \begin{bmatrix} \tan \theta \\ 1 \\ 0 \end{bmatrix} \right\|}, \quad (13)$$

where  $\times$  represents the cross product and  $\|\cdot\|$  denotes the two-norm.

Finally, we determine the label of the pixel by comparing it with the minimum distances to the left and the right visible pixels as

$$L(u, v) = \begin{cases} \text{"left"}, & \min(D(\mathbf{P}_{u,v}, \mathbf{s}'_{n,l})) < \min(D(\mathbf{P}_{u,v}, \mathbf{s}'_{n,r})) \\ \text{"right"}, & \text{otherwise} \end{cases}. \quad (14)$$

The pixels with labels "left" have the pixel intensity of those positions on the left source image; the same applies for the "right" label case. An overview of the calibration process is presented in Figure 2.



**Figure 2.** Overview of calibration process using the calibration module. The calibration module consists of a three-dimensional calibration pattern and an external camera.

### 2.3. Parameter Classification

As described above, various parameters are required to generate a correct multiplex image for a certain viewing position. This means that accumulation of small parameter errors can cause visual quality degradation; therefore, accurate calibration is needed for the ETAD. The parameters are listed in Table 1. We classify these parameters into two categories: constant and variable parameters. Constant parameters have absolute values, determined when manufacturing the devices and there is little variation between the devices. These constant parameters also hardly change with movement. Therefore, it is sufficient to use designed parameters directly for the constant parameters. The width and height of the pixel  $\mu_w$ ,  $\mu_h$ , and the pitch of an optical layer  $p$  belong to this category.

**Table 1.** Device parameters for the ETAD rendering process.

Device	Parameters	Symbols	Category
onboard camera	focal length	$f_x, f_y$	variable
	principal point	$c_x, c_y$	variable
	distortion coefficient	$k_1, k_2, k_3, k_4$	variable
display panel	pixel width	$\mu_w$	constant
	pixel height	$\mu_h$	constant
optical element	pitch	$p$	constant
	slanted angle	$\theta$	variable
	gap (thickness)	$\tau$	variable
	offset	$\epsilon$	variable
onboard camera pose	rotation	$r_x, r_y, r_z$	variable
	translation	$t_x, t_y, t_z$	variable

On the other hand, some parameters, classified as variable parameters, are not only different from the design value, but even have different values for each ETAD. These errors mainly occur during the production process of each device. These errors can even increase if the device is not fixed during the

calibration process. For onboard cameras, focal lengths and principal points, which are dependent on the pose of the image sensor and the lens, vary from device to device. The radial distortions of cameras also have different values between each other. Some parameters of the additional optical layers, such as the slanted angle, the gap, and the offset, are determined in the assembly process with the display panel, because those parameters are relative to the display coordinates. The relative pose between the onboard camera coordinate system and the display coordinate system also depends on how the onboard camera is attached to the display. In our study, therefore, we focus on the calibration of those “variable parameters”. Based on the classification of parameters, we describe the calibration method in the next section.

### 3. Calibration

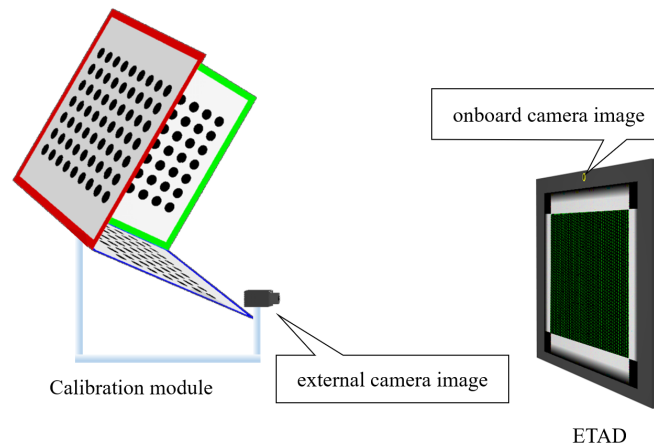
#### 3.1. Calibration Process

The proposed method uses two images to calibrate all the parameters of the ETAD. One image is for estimating the parameters of the onboard camera and the other image is for parameters of the optical layer. To estimate camera parameters, i.e., intrinsic parameters and distortion coefficients, the onboard camera takes a picture of the calibration pattern. The same applies to calibrating the optical layer. In this case, the optical layer is a kind of pattern to be captured. Because the viewing angle of the onboard camera does not include the optical layer, an additional camera is required to capture the optical layer. In summary, we use two different cameras: the onboard camera and the external camera; hence, we can capture two images at the same time. To estimate the pose between the onboard camera and the display panel, we introduce an indirect pose estimation method. As mentioned above, it is not possible to directly capture the display panel or the optical layer with only the onboard camera. Some methods use equipment, e.g., mirrors, to capture objects to estimate the pose. These methods require several images, so a robot or a manipulator that can move the mirror is needed. Alternatively, the object in front of the mirror should be moved, which would cause parameter changes. Our method estimates the final pose by combining three poses. Two poses are estimated from two captured images and have different values for each calibration. The other pose is estimated once and is fixed.

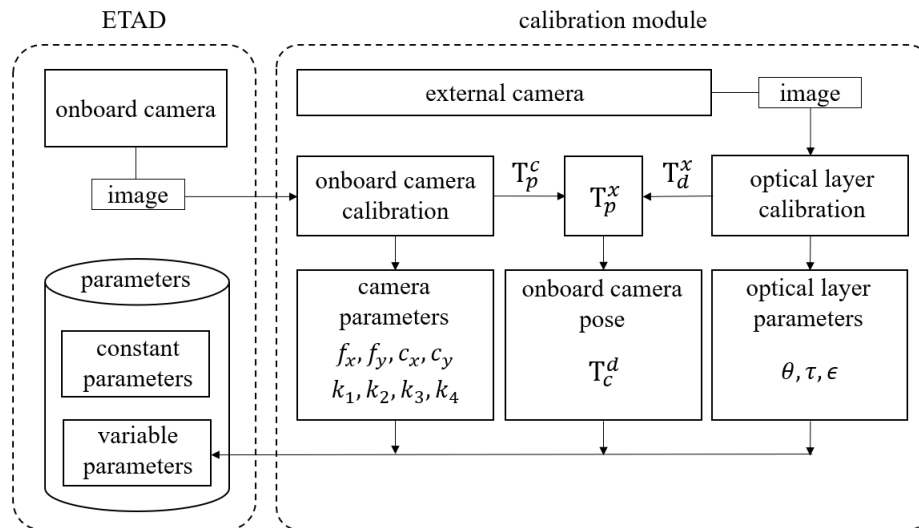
With the proposed calibration method, we can calibrate the ETAD automatically and simultaneously; however, additional elements are needed: the calibration pattern for the onboard camera and the external camera to capture the display panel. We designed a calibration module that consists of these two devices: the calibration pattern and the additional camera, which has a common structure with the ETAD (see Figure 3). As with the ETAD, the calibration pattern is not included in the viewing angle of the external camera of the calibration module. Therefore, we placed the ETAD in front of the calibration module for the calibration process. The onboard camera of the ETAD captures an image of the calibration pattern of the calibration module. Oppositely, the external camera of the calibration pattern captures an image of the optical layer with the display pattern. Here, we can estimate the parameters of the ETAD devices, i.e., the onboard camera and the optical layer, because we know in advance the parameters of the devices constituting the calibration module.

The block diagram of the calibration process with two images is represented in Figure 4. When the calibration begins, the onboard camera and the external camera simultaneously capture the patterns on the opposite side. Using the onboard camera image, the system calibrates the onboard camera parameters, i.e.,  $f_x, f_y, c_x, c_y, k_1, k_2, k_3, k_4$ . At this time, it also estimates the transformation matrix  $\mathbf{T}_c^p$  from the calibration pattern coordinate system to the onboard camera coordinate system. Simultaneously, the system calibrates the optical element parameters, i.e.,  $\theta, \tau, \epsilon$ , using the external camera image. The onboard camera pose, i.e.,  $r_x, r_y, r_z, t_x, t_y, t_z$ , which also can be represented as the transformation matrix  $\mathbf{T}_c^d$  is finally calibrated using two estimated transformation matrix  $\mathbf{T}_c^p, \mathbf{T}_d^x$  and  $\mathbf{T}_x^p$  calibrated in advance. The details of the calibration method are presented in the following section.





**Figure 3.** Overview of the calibration process using the calibration module. The calibration module consists of a 3D calibration pattern and an external camera.



**Figure 4.** Block diagram of the calibration process.

### 3.2. Calibration Method

**Onboard camera calibration:** The onboard camera parameters are calculated by analyzing the pattern image of the 3D calibration pattern of the calibration module. We employ Zhang's camera calibration method [20] to estimate both intrinsic parameters and distortion parameters of the onboard camera. The calibration pattern of the calibration module is built such that the three 2D patterns are orthogonal to each other. Hence, once an image is captured, the system detects a 3D pattern on the image and split it into three 2D patterns. However, three images are not enough to estimate accurate parameters using [20]; therefore, we add an orthogonal constraint among the estimated poses of the three planes. Using the known 3D points and the corresponding points in the captured image, we estimate the onboard camera parameters and the poses of the plane patterns by minimizing the following functional:

$$\sum_{i=1}^3 \sum_{j=1}^m \|m_{ij} - \tilde{m}(f_x, f_y, c_x, c_y, k_1, k_2, k_3, k_4, \mathbf{R}_i, \mathbf{t}_i, M_j)\|^2 - \lambda \|\hat{\mathbf{R}}^T \hat{\mathbf{R}} - \mathbf{I}\| \quad (15)$$

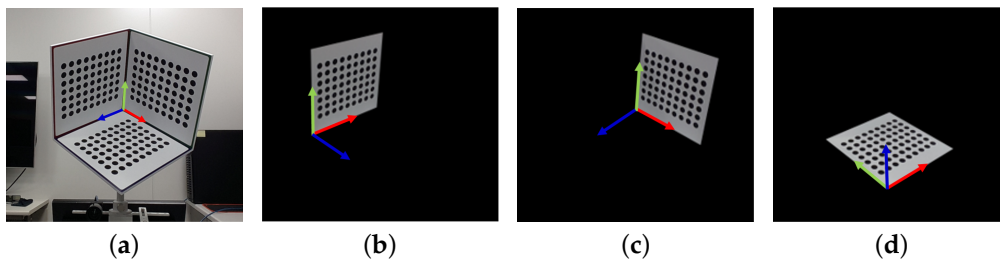
where the left term denotes reprojection error and the right term denotes the orthogonal constraint term. In the left term,  $j$  represents the index of the points and  $i$  represents the index of the 2D patterns. The function  $\tilde{m}$  projects the 3D point  $M_j$  of the  $i$ th pattern, and  $m_{ij}$  corresponds to the points on the image. In the right term,  $\lambda$  is a weight factor; we set this term to 0.01. The matrix  $\hat{\mathbf{R}}$  is defined by a composition of vectors, which are the basis vectors of  $\mathbf{R}_1, \mathbf{R}_2, \mathbf{R}_3$  and can be represented as

$$\hat{\mathbf{R}} = \mathbf{R}_1 \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \mathbf{R}_2 \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} + \mathbf{R}_3 \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & -1 \\ 0 & 0 & 0 \end{bmatrix}. \quad (16)$$

Because the three planes of the calibration pattern are orthogonal, the matrix  $\hat{\mathbf{R}}$  also has to satisfy orthogonality. This means that the product of  $\hat{\mathbf{R}}$  and the transpose of  $\hat{\mathbf{R}}$  is the identity matrix  $\mathbf{I}$ . The optimization is solved using Levenberg–Marquardt nonlinear optimization [25]. The transformation matrix  $\mathbf{T}_x^c$  can be estimated as

$$\mathbf{T}_x^c = \begin{bmatrix} \mathbf{R}_2 & \mathbf{t}_2 \\ 0 & 1 \end{bmatrix}, \quad (17)$$

because the coordinate system of the second pattern is the same as the 3D calibration pattern coordinate system (see Figure 5).



**Figure 5.** Onboard camera parameter calibration using a 3D calibration pattern of the calibration module: (a) onboard camera image; (b) the first 2D calibration pattern; (c) the second 2D calibration pattern; and (d) the third 2D calibration pattern. The red, green, and blue arrows represent the  $x$ -,  $y$ -, and  $z$ -axis, respectively.

**Optical layer calibration:** The system calibrates the parameters of the optical layer such as the pitch, the slanted angle, the gap, and the offset. Our method is mainly based on the visual pattern analysis method [21], which uses two captured images by an external camera. First, a pattern is displayed on the panel. This pattern is composed of two vertically striped patterns to avoid ambiguity between the slanted angle and the pitch. The two vertical stripe patterns have different distances and different colors, e.g., blue and green, to separate them later. However, the ambiguity between the gap and the pitch remains. To solve this problem, Hwang et al. [21] captured the pattern images twice at different distances. They first capture the pattern image at a close distance, then move the camera and capture the image from a long distance. It is not suitable to apply this method directly to our system because of the movement of the camera. We propose a revised method to calibrate the parameters of the optical layer in a fixed environment. The key idea of our method is that we only take into account three parameters of the optical layer, excluding the pitch because it was classified as a constant parameter. With the revised method, ambiguities are solved and only one capture image of the pattern is required to calibrate parameters.

We first displayed the pattern image on the display panel and capture the image using the external camera. We employed a state-of-the-art method to estimate camera pose called homography decomposition [26]. We detected positions of the four corners of the display in the captured image

and found homography between the external camera coordinates and the image coordinates. The homography can be decomposed into the intrinsic parameters of the external camera and the camera pose matrix, which is denoted as  $\mathbf{T}_x^d$  in this paper. We assumed the intrinsic parameters of the extrinsic camera are calibrated in advanced to this process, thus the camera pose matrix  $\mathbf{T}_x^d$  can be computed by multiplying the inverse of the intrinsic matrix to the homography. Even if we found the four corners of the pattern area, it is necessary to restore the pattern to its original shape because the shape of the pattern in the captured image may appear distorted due to the camera rotation. To calibrate the system with the captured image, a rectification process is needed. We used a simpler scheme that warps the four corners to the vertices of a rectangle of the display pattern [24].

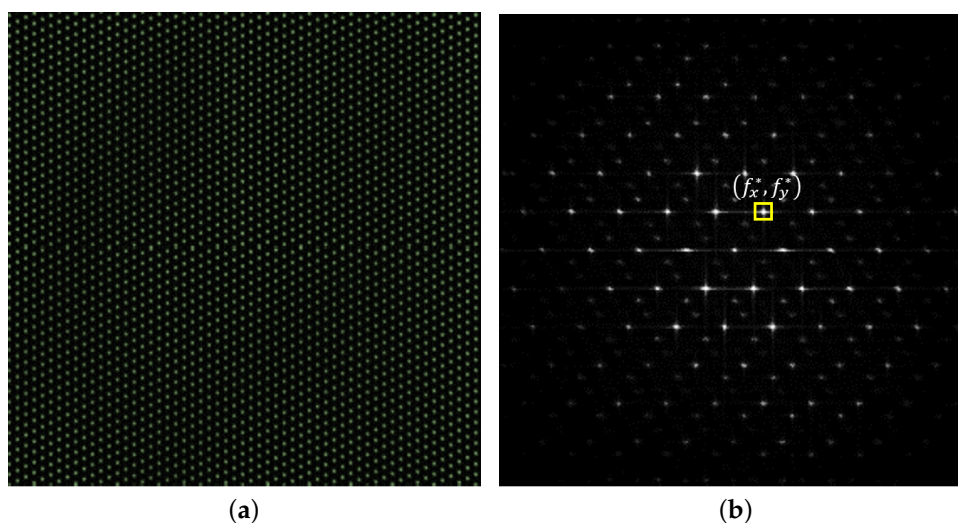
In the captured image, the original stripe pattern is shown as a lattice pattern (see Figure 6a). This lattice pattern is generated by the intersection of the two stripes: the stripes of the display pattern and the stripes of the optical layers. The lattice pattern  $L$  can be represented as

$$L(x, y) = \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \left\{ \delta(x - m\beta) \cdot \delta\left(y - \frac{1}{\tan\theta} \left(m\beta - n \frac{dp}{(d-\tau)\cos\theta}\right)\right) \right\}, \quad (18)$$

where  $\delta$  denotes Dirac's delta function. In Equation (18),  $x, y$  represent the positions on the lattice pattern, and  $\beta$  denotes the horizontal space of the display pattern. The distance of the external camera from the display pattern is represented as  $d$ . Note, the parameter *beta* is fixed and the distance  $d$  was obtained when estimating the external camera pose. Therefore, we can estimate parameters of the optical layer  $p, \tau$  by analyzing the positions of  $x, y$  and then determining optimal values of  $m, n$ . Generally, a 2D lattice in a spatial domain is mapped to another 2D lattice in the frequency domain [27]. In the frequency domain, Equation (18) is transformed into

$$\hat{L}(f_x, f_y) = C \sum_{m=-\infty}^{\infty} \sum_{n=-\infty}^{\infty} \left\{ \delta\left(f_x - m \frac{1}{\beta} + n \frac{(d-\tau)\cos\theta}{dp}\right) \cdot \delta\left(f_y - n \tan\theta \frac{(d-\tau)\cos\theta}{dp}\right) \right\}, \quad (19)$$

where  $C$  denotes a constant scale factor. Estimating a peak point  $(f_x, f_y)$  of the lattice in the frequency domain instead of  $(x, y)$  in the spatial domain has several advantages. The signal-to-noise ratio (SNR) around the peak is quite high in the frequency domain; therefore, the position of the peak point is estimated more accurately. Moreover, noise is well suppressed in the frequency domain [21].



**Figure 6.** The captured lattice pattern is represented in: (a) spatial domain; and (b) frequency domain.  $(f_x^*, f_y^*)$  on the frequency domain is the peak point.

For estimating the parameters of the optical layers, we first transformed the warped pattern image  $I(x, y)$ , which is in the form of a 2D lattice in a spatial domain to a frequency domain image  $\hat{I}(f_x, f_y)$  using the discrete Fourier transform. Then, we estimated a peak point of the lattices in the frequency domain. To estimate the accurate peak position, the paraboloid fitting was exploited. The parameters concerned with the display pattern are already known, thus the parameters of the optical layer can be estimated by decoding the lattice pattern. After detecting the peak point  $f_x^*, f_y^*$  in the frequency domain, the number of stripes  $m$  and the number of slits  $n$  are estimated as

$$(m^*, n^*) = \arg \min_{m, n} \left| p_0 - \frac{d - \tau_0}{d} \left( \frac{n}{m/\beta - f_x^*} \right) \cos \left( \arctan \frac{f_y^*}{m/\beta - f_x^*} \right) \right|, \quad (20)$$

where  $p_0$  and  $\tau_0$  mean the designed pitch and gap, respectively. The selected  $m$  and  $n$  bring the estimated pitch closer to the designed one. We determined  $m^*, n^*$  by exhaustive search within the range 1–20. The slanted angle  $\theta$  and the gap  $\tau$  are then derived using the peak points  $f_x, f_y$  and the selected numbers of  $m^*, n^*$  as

$$\theta = \arctan \frac{f_y^*}{m^*/\beta - f_x^*}, \quad (21)$$

$$\tau = z \left( 1 - \frac{p}{\cos \theta} \frac{m^*/\beta - f_x^*}{n^*} \right). \quad (22)$$

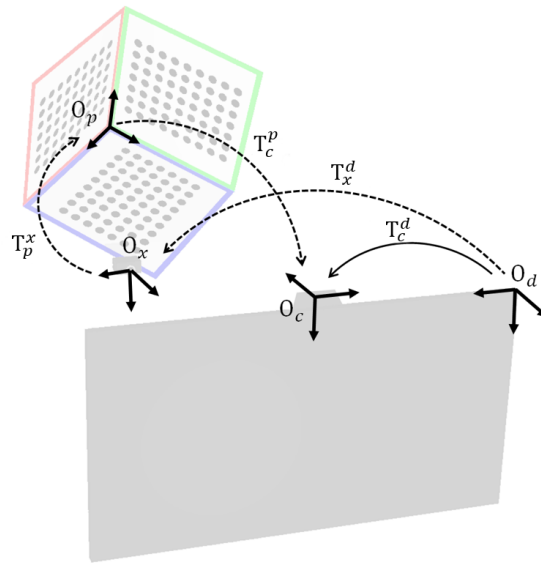
Finally, we generated a lattice pattern using these parameters with zero offset. The offset  $\epsilon$  of the optical element was estimated by measuring the vertical shift between the generated lattice pattern and the captured pattern.

**Onboard camera pose calibration:** The onboard camera pose plays an important role because it converts the three-dimensional position of the viewer's eyes from the camera coordinate system to the display coordinate system. For calibrating the onboard camera pose, we propose an indirect pose estimation method instead of direct pose estimation. As mentioned earlier, the direct pose estimation has several problems and is not suitable for our method, which aims toward an automated and simultaneous calibration process. With the indirect pose estimation method, we estimated the pose of the onboard camera  $\mathbf{T}_c^d$  by multiplying several poses as

$$\mathbf{T}_c^d = \mathbf{T}_x^d \mathbf{T}_p^x \mathbf{T}_c^p, \quad (23)$$

where  $\mathbf{T}_x^d$  and  $\mathbf{T}_c^p$  are the inverse matrix of  $\mathbf{T}_d^x$  and  $\mathbf{T}_p^c$ .  $\mathbf{T}_p^x$  denotes the pose between the external camera and the calibration pattern on the calibration module (see Figure 7). Therefore, once the  $\mathbf{T}_p^x$  is estimated in advanced to the calibration process, it can be used as a fixed value each time the ETAD is calibrated. The other poses, such as  $\mathbf{T}_x^d$  and  $x_p \mathbf{T}_c^p$ , have different values for each calibration process of the ETAD. Note, however, that the estimation of these poses is not needed because they are already estimated in the process of calibrating other devices.  $\mathbf{T}_c^p$ , which denotes the pose between the onboard camera and the calibration pattern, is estimated in the process of the onboard camera calibration. Likewise, the pose between the display coordinate system and the calibration pattern coordinate system  $\mathbf{T}_x^d$  is estimated in the optical layer calibration process. Therefore, there are no additional processes for estimating the onboard camera pose without multiplying the already known pose matrix.

The proposed indirect method has two advantages. First, the calibration process performed in this method is carried out under a fixed environment. Second, no additional devices or processes are required for estimating the onboard camera calibration, which significantly reduces the calibration time of the ETAD.

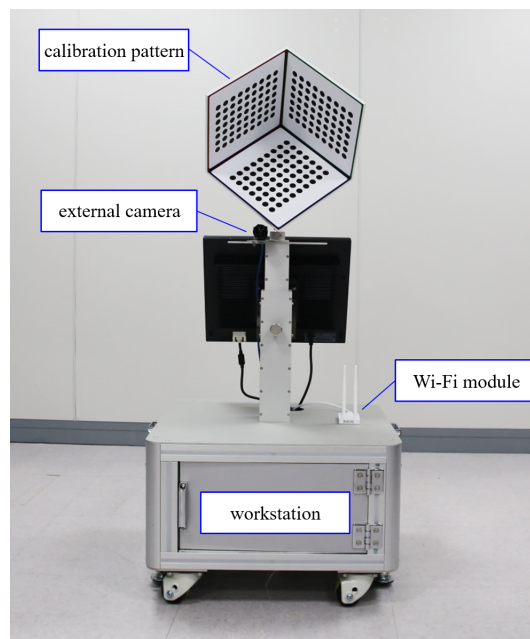


**Figure 7.** Coordinate systems with respect to each device of the ETAD. The arrow with solid line is the direct pose estimation. The arrows with dashed lines are paths of indirect pose estimation.

#### 4. Experiments and Discussion

We conducted two types of experiments to evaluate the performance of the proposed calibration scheme. First, we evaluated our method on a synthetic dataset by using a simulation tool. The simulation provides a controlled environment, i.e., we set up the system and obtained images with various parameters. In this case, the ground-truth parameters are available for the synthetic dataset; therefore, we could quantitatively measure the estimation error. We also conducted the experiment in the same manner by adding noise to the original dataset and evaluated the robustness of the proposed method.

Next, we applied the calibration scheme to real-life displays. We used the calibration module that we designed and built as described in the previous section (see Figure 8).



**Figure 8.** Calibration module.

The calibration module contains not only a calibration pattern and an external camera but also a workstation and a Wi-Fi module. Through Wi-Fi, the ETAD transmits the image of the onboard camera to the workstation and receives the calibration results. The proposed calibration algorithms were implemented on a workstation equipped with an Intel Core(TM) i7-4960 processor and 64 GB of memory. In the case of the real-life ETAD, the actual parameters were relatively different from the designed ones. However, we could not measure the errors directly because the ground-truth parameters were unknown. We evaluated the performance in the visual aspect of the observed images. We prepared a measuring device that mimics a person: a fake face was attached on the stereo cameras. The ETAD continuously rendered 3D images according to the position of the viewer's eyes. The stereo camera captured the left image and the right image. With the captured images, we assessed the visual quality with two types of evaluation: the first was the measurement of the crosstalk using red-blue images and the second was the image quality measurement using PSNR and SSIM index [28].

#### 4.1. Simulation Environment

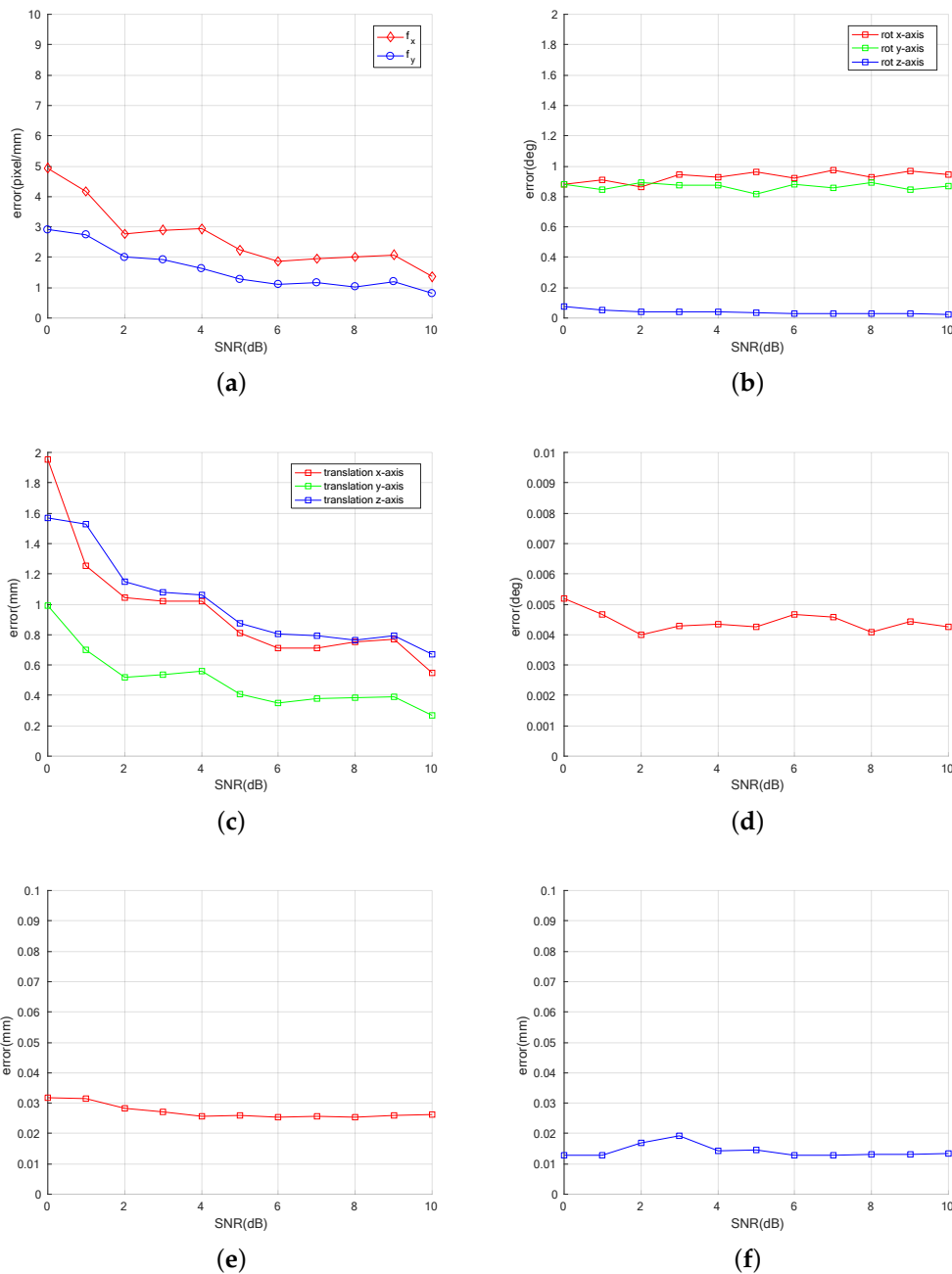
For the synthetic dataset, we built a simulation environment based on a ray-tracing tool (POV-RAY [29]). The first system was built using the designed parameters listed in Table 2. We generated 100 datasets by adding or subtracting random values from the designed ones, except for the pixel width, pixel height, and pitch parameters, which are classified as constant. We also used constant values for principal points and distortion coefficients for the onboard camera because they cannot be handled with the simulation tool. In the calibration evaluation, however, few attempts were made to measure errors of parameters directly instead of measuring the standard deviation or reprojection error. Thus, we used fixed values for the principal points of the onboard camera ( $c_x = 960, c_y = 480$ ) and set all of the distortion coefficients ( $k_1-k_4$ ) to zero. We defined the resolution of the display panel as  $1920 \times 1.080$  and both pixel sizes  $\mu_x, \mu_y$  as 0.0846 mm. For each case of the datasets, we captured two images, one from the position of the onboard camera and the other from positions of the external camera of the calibration module. Given the two captured images, we first simultaneously estimated the onboard camera parameters and pose of the onboard camera, as well as the optical layer parameters and the pose of the display panel. Then, we finally estimated the onboard camera pose parameters from the origin of the display panel. The total processing time was 0.8 s. The estimation results are presented in Table 2. The accuracy of the focal length is comparable to that of the existing camera calibration method [30], by comparing the standard deviation.

**Table 2.** Designed parameters and experimental results in the simulation environment.

Device	Parameter (Unit)	Designed		Estimation Errors	
		Designed Value	Variation	Mean of Absolute Error	Standard Deviation
onboard camera	$f_x$ (pixel/mm)	1506.9	$\pm 40.0$	0.9167	0.7122
	$f_y$ (pixel/mm)	1506.9	$\pm 40.0$	0.6045	0.4178
	$c_x$ (pixel)	960	-	-	-
	$c_y$ (pixel)	480	-	-	-
optical layer	$p$ (mm)	0.01237	-	-	-
	$\theta$ ( $^\circ$ )	12.5288	$\pm 1.0$	0.0041	0.0031
	$\tau$ (mm)	0.5	$\pm 0.1$	0.0252	0.0175
	$\epsilon$ (mm)	0.2	$\pm 0.1$	0.0135	0.0087
onboard camera pose	$r_x$ ( $^\circ$ )	0	$\pm 1.0$	0.9362	0.5986
	$r_y$ ( $^\circ$ )	0	$\pm 1.0$	0.8596	0.6038
	$r_z$ ( $^\circ$ )	0	$\pm 1.0$	0.0160	0.0124
	$t_x$ (mm)	204.48	$\pm 5.0$	0.4171	0.3226
	$t_y$ (mm)	-15.0	$\pm 5.0$	0.1804	0.1364
	$t_z$ (mm)	2.0	$\pm 5.0$	0.6030	0.3814

We also evaluated the same dataset with additional noise to verify the robustness of the proposed algorithm in a noisy environment. We added Gaussian noise from signal-noise-ratio (SNR) 0–10 to all the generated datasets. Given the 2200 images, we also evaluated the errors of the parameters.

The result of parameter estimation errors is described in Figure 9. Note, error magnitudes of the parameters related to the optical element are quite low even with a significant amount of noise. On the other hand, the translation error of the pose tends to be affected by noise levels up to values greater than 5 dB of the SNR.

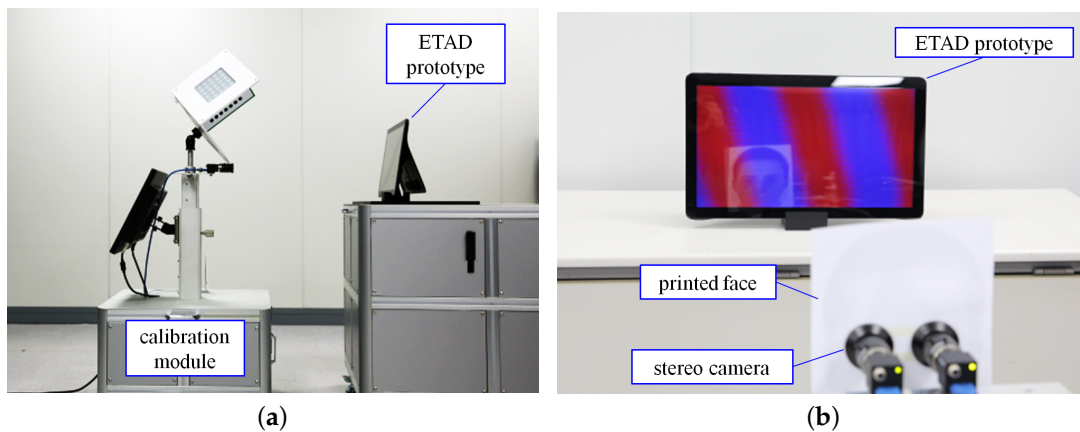


**Figure 9.** Mean absolute error for parameter estimation in noisy settings: (a) focal length  $f_x, f_y$  of the onboard camera; (b) rotational error  $r_x, r_y, r_z$ ; (c) translational error  $t_x, t_y, t_z$  of the pose; (d) angle error  $\theta$ ; (e) gap error  $\tau$ ; and (f) offset error  $\epsilon$  of the optical layer.

#### 4.2. Real System

We also applied the proposed method to a real-life ETAD. The proposed method was applied to a prototype of the ETAD system that included an 18.5-inch LCD panel with a resolution of  $1920 \times 1080$ . We inserted an additional parallax barrier in front of the display panel. The parameters concerned

with the calibration module, i.e., external camera parameters and the pose  $T_x^p$  were obtained before beginning the calibration process. Because our calibration process is fully automated, all we had to do was place the ETAD in front of the calibration module (see Figure 10a). The ETAD was connected to the calibration module through Wi-Fi. Then, the images were simultaneously captured from the onboard camera and the external camera. The image of the onboard camera was transmitted to the workstation in the calibration module. Using the PC in the module, all parameters were estimated and the results were finally transmitted to the prototype. The overall processing time from connection to writing parameters onto the tablet was less than 1.46 s. The reason behind the overall processing time being longer than the simulation processing time is that it takes into consideration the waiting time after command and image transmission.



**Figure 10.** Calibration process: (a) calibration process using the calibration module; and (b) verification process using a stereo camera with a fake facial image.

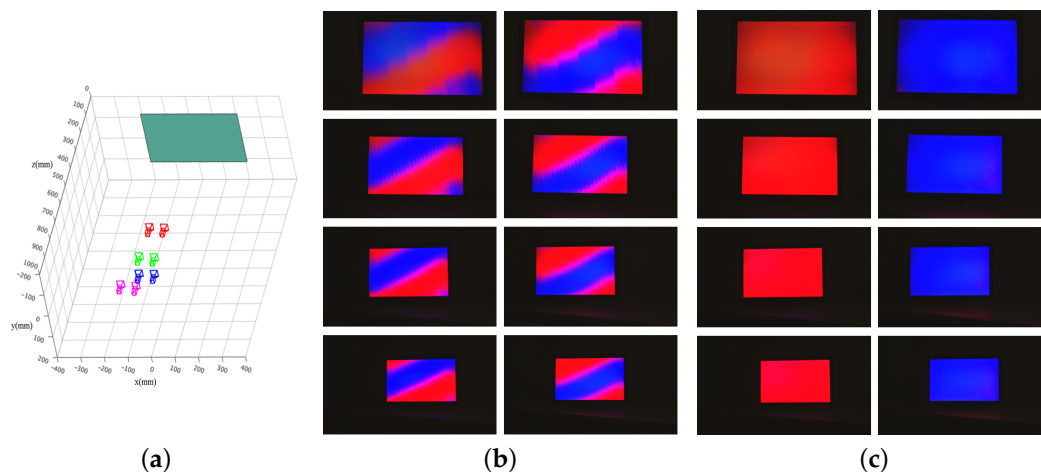
After completing the calibration, we performed the visual quality evaluation using the calibrated parameters. To measure the view of the user, we set up stereo cameras, approximately 65 mm apart, and attach a printed image of a human face onto the lens shown in Figure 10b. The onboard camera continuously detected the eyes from the face-mimicking image, and a multiplex image was generated from a set of left and right images. We set the stereo camera system at different positions with z-axis values ranging 300–600 mm and performed the above-mentioned process. For every captured image, we adjusted the color to compensate for the difference of RGB spectra between the display and the camera.

For the measurement of the crosstalk, we encoded the views based on color, assigning red to the left, and blue to the right. Ideally, only red pixels are visible on the left image and blue pixels are visible on the right images. However, if errors of parameters are large, the views are not sufficiently separated, and a mix of both images is displayed. This effect is referred to as extrinsic crosstalk. We measured the extrinsic crosstalk according to the scheme described in [31]:

$$\text{Extrinsic crosstalk(\%)} = \frac{\text{Incorrect view luminance}}{\text{Correct view luminance}} \times 100. \quad (24)$$

We set four positions of different distances between 300 mm and 600 mm. Then, we captured 100 images at each position and analyzed the crosstalk. Figure 11 shows the left and right captured images at different camera positions. In the case of no calibration, the left view (red) and the right view (blue) appear mixed. On the other hand, the left and right views are sufficiently separated when using the parameters from the proposed method. The results are summarized in Table 3. The average external crosstalk value was 8.32%, which is within an acceptable range for a two-view autostereoscopic display [32]. Note, the crosstalks of the no-calibration case are so large that it may be difficult for users to perceive scenes in 3D.





**Figure 11.** Examples of the two-view image captured at different distances: (a) captured positions of the stereo camera with a fake facial image; (b) captured images of no-calibration; and (c) captured images using the proposed method. The images in Rows 1–4 were captured in the positions at the test positions #1–#4, which are colored in red, green, blue, and magenta in (a), respectively.

**Table 3.** Crosstalk evaluation of the ETAD using red-blue images.

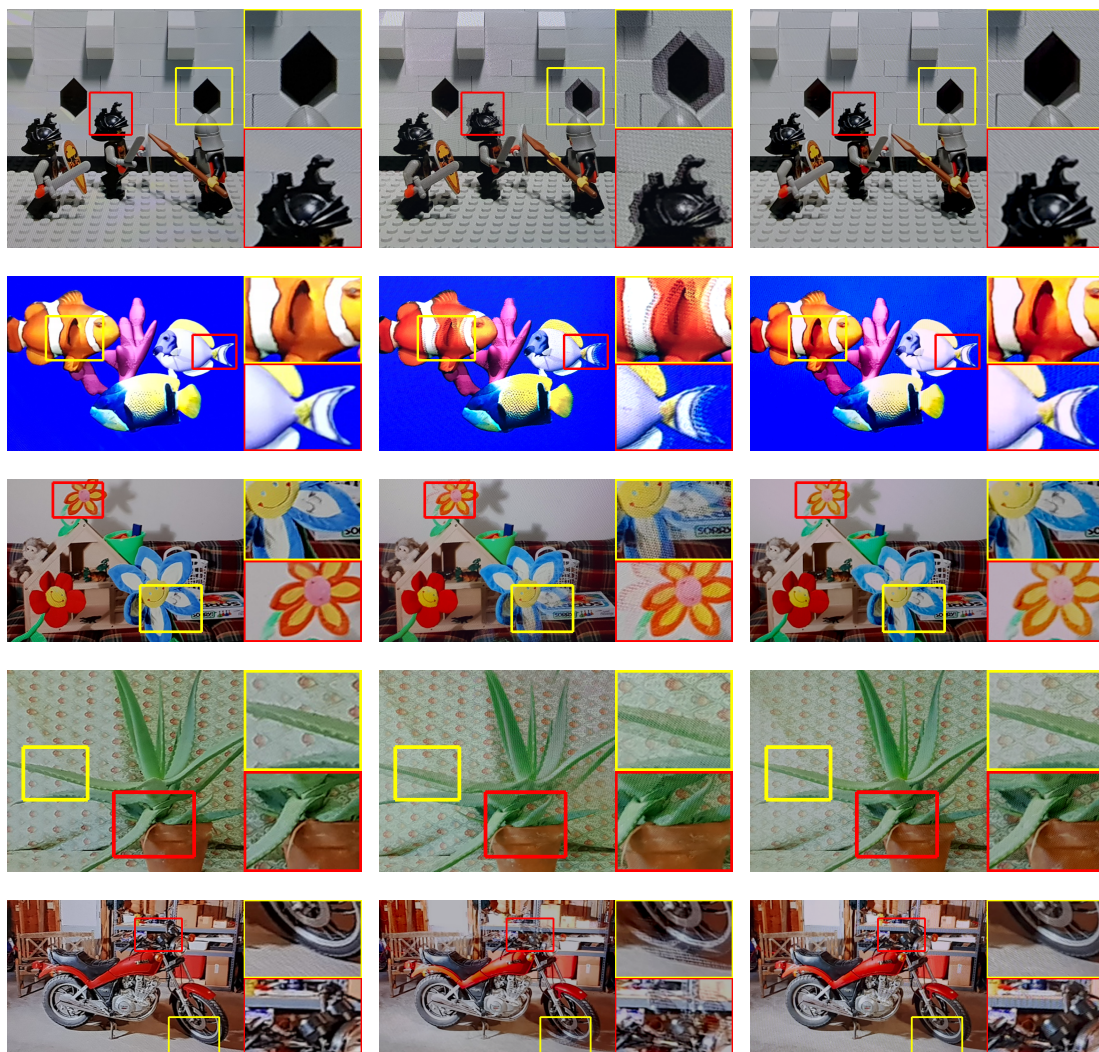
Distance	No Calibration	Proposed Method
test position #1	77.5889	7.9029
test position #2	85.7855	8.0671
test position #3	91.0957	8.5187
test position #4	88.1954	8.7885
mean	85.6664	8.3193

We also evaluated the visual quality in terms of PSNR and SSIM index. In this case, the two views of red–blue are no longer suitable because both metrics need a reference image, which has various colors and structure. We employed de facto standard datasets to evaluate stereo and multi-view image quality such as Middlebury stereo dataset [33,34], The Stanford Light Field Archive [35], MIT Synthetic Light Field Archive [36], and Disney Light Field Archive [37]. From the datasets, we selected eight images, i.e., Lego Knights, Fish, Flower, Aloe, Motorcycle, Baby3, Couch, and Bicycle1. To obtain the reference images of the selected ones, we followed the same method proposed in [21]: We first identified the view corresponding to the camera position. We then displayed the viewpoint image, which was one of the left or the right image, on the panel. This means that a 2D image was displayed on the ETAD. Then, we captured the image, which was free from artifacts caused by the other view image. The evaluation was performed in the same manner, except that we used a multiplexed image rather than a 2D image.

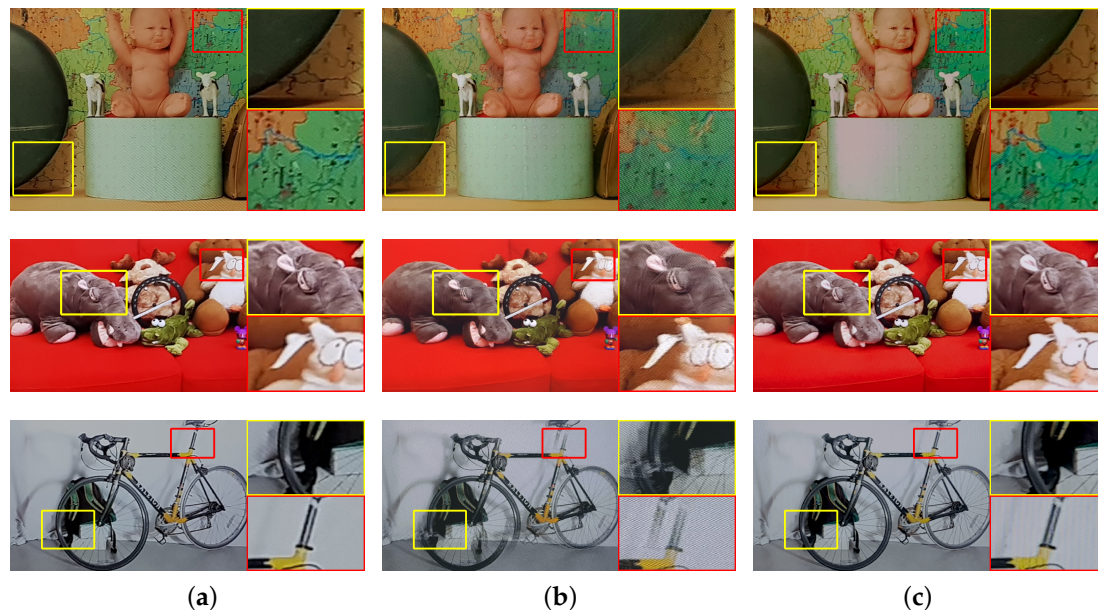
The results of the visual quality assessment are listed in Table 4. The gains of using the proposed method over no calibration are large (5.62 dB in PSNR, 0.2 in SSIM index on average). Figure 12 also shows more details of the results. Similar to the crosstalk evaluation results, two views are mixed in a view when rendering without calibration. In the images of the no-calibration cases, the edges in the wall bricks, the fish’s tail, the flower’s petal, the leaf of aloe, the front wheel of the motorcycle, the edge of ball, the hippo’s eye, and the front wheel of the bicycle are exposed in duplicate. However, we can verify that those artifacts almost disappear when the proposed method is used.

**Table 4.** Visual quality assessment. The shown numbers are the mean and maximum deviations (in parenthesis) of the PSNR and the SSIM index over four display instances.

Name	PSNR (Standard Deviation)		SSIM (Standard Deviation)	
	No Calibration	Proposed	No Calibration	Proposed
Lego Knights	18.51 (2.46)	26.25 (0.46)	0.44 (0.02)	0.65 (0.03)
Fish	17.83 (0.64)	22.75 (0.10)	0.56 (0.02)	0.80 (0.03)
Flower	23.62 (0.36)	30.18 (0.30)	0.68 (0.01)	0.92 (0.04)
Aloe	20.55 (1.05)	29.39 (0.59)	0.48 (0.04)	0.80 (0.03)
Motorcycle	19.67 (0.19)	26.05 (0.25)	0.72 (0.01)	0.82 (0.06)
Baby3	24.06 (0.08)	27.61 (0.49)	0.47 (0.03)	0.73 (0.12)
Couch	23.34 (0.15)	29.45 (1.02)	0.67 (0.03)	0.91 (0.17)
Bicycle1	20.07 (0.25)	28.76 (0.89)	0.34 (0.12)	0.88 (0.10)



**Figure 12.** Cont.



**Figure 12.** Example images observed on a real-life ETAD display. The images Lego Knights, Fish, Flower, Aloe, Motorcycle, Baby3, Couch, and Bicycle1 (from top to bottom) were rendered using the parameters from our designed method and the calibration results: (a) reference; (b) no calibration; and (c) calibration using the proposed method.

## 5. Conclusions

Despite the ability of the ETAD to fix issues faced by the autostereoscopic displays, the parameters of the ETAD become more complex and need higher levels of accuracy. In terms of mass production, the calibration process is essential for devices whose parameters deviate from designed ones. In this paper, we propose an efficient calibration method to estimate all variable parameters of the ETAD.

First, we determined the required parameters of the ETAD system and classified the parameters based on their ability to be misaligned during the assembly process. To fix the parameter errors, we proposed a simultaneous estimation algorithm using two images. For the automated process, we designed a calibration module comprising a pattern and an external camera. By using the module, the entire calibration process can be carried out in fixed environments, thus avoiding the need of any external manipulation. Experimental results obtained by evaluating the proposed method using a synthesis dataset show that the proposed method minimized the estimated parameter errors to within 1 s. Using a real ETAD prototype, rendering with the calibrated parameters limited the crosstalk to under 8.5%. The visual quality was also improved to 30.44% in PSNR and 40.14% in SSIM index by using the proposed method.

**Author Contributions:** H.H. proposed the idea of the paper, implemented the algorithm, and designed and performed the experiments.

**Funding:** This research received no external funding.

**Acknowledgments:** The authors would like to appreciate H. S. Chang for his generous advice.

**Conflicts of Interest:** The authors declare no conflict of interest.

## Appendix A.

### Appendix A.1. Eye Position Estimation

Given the unit vectors  $\mathbf{v}_l = [x'_l, y'_l, 1]^T$  and  $\mathbf{v}_r = [x'_r, y'_r, 1]^T$ , we can define the 3D positions of the left and right eye  $\mathbf{e}_l^c, \mathbf{e}_r^c$  as

$$\mathbf{e}_l^c = \kappa_l \mathbf{v}_l, \quad (\text{A1})$$

$$\mathbf{e}_r^c = \kappa_r \mathbf{v}_r, \quad (\text{A2})$$

where  $\kappa$  is distance to each eye. The distance of two eyes is equal to IPD  $\psi$  and defined as

$$\psi = \|\kappa_l \mathbf{v}_l - \kappa_r \mathbf{v}_r\|. \quad (\text{A3})$$

The vector passing through the two eyes is orthogonal to face normal vector  $\phi$  and is represented as

$$(\kappa_l \mathbf{v}_l - \kappa_r \mathbf{v}_r) \cdot \phi = 0, \quad (\text{A4})$$

where  $\cdot$  represents the inner product. We assumed that the viewer is watching at the center of the camera, therefore, the face normal vector has the inverse direction of the  $\mathbf{v}_l, \mathbf{v}_r$  as

$$\phi = \frac{\mathbf{v}_l + \mathbf{v}_r}{\|\mathbf{v}_l + \mathbf{v}_r\|}. \quad (\text{A5})$$

From Equation (A6), we obtain

$$\kappa_l = \frac{\mathbf{v}_r \Phi}{\mathbf{v}_l \Phi} \kappa_r. \quad (\text{A6})$$

By replacing with  $\alpha$ , which is defined in Equation (6), Equation (A6) can be rewritten as

$$\kappa_l = \alpha \kappa_r, \quad (\text{A7})$$

$$\kappa_r = \kappa_l / \alpha. \quad (\text{A8})$$

We insert this into Equation (A3) to obtain the distance parameter as

$$\kappa_l = \frac{\Psi}{\|\mathbf{v}_l - \mathbf{v}_r / \alpha\|}, \quad (\text{A9})$$

$$\kappa_r = \frac{\Psi}{\|\alpha \mathbf{v}_l - \mathbf{v}_r\|}. \quad (\text{A10})$$

finally, we can rewrite the eye positions (A1) and (A3) as follows,

$$\mathbf{e}_l^c = \frac{\Psi}{\|\mathbf{v}_l - \mathbf{v}_r / \alpha\|} \mathbf{v}_l, \quad (\text{A11})$$

$$\mathbf{e}_r^c = \frac{\Psi}{\|\alpha \mathbf{v}_l - \mathbf{v}_r\|} \mathbf{v}_r. \quad (\text{A12})$$

## References

1. Dodgson, N.A. Optical devices: 3D without the glasses. *Nature* **2013**, *495*, 316–317. [[CrossRef](#)] [[PubMed](#)]
2. Urey, H.; Chellappan, K.V.; Erden, E.; Surman, P. State of the art in stereoscopic and autostereoscopic displays. *Proc. IEEE* **2011**, *99*, 540–555. [[CrossRef](#)]
3. Takaki, Y. Multi-view 3-D display employing a flat-panel display with slanted pixel arrangement. *J. Soc. Inf. Disp.* **2010**, *18*, 476–482. [[CrossRef](#)]
4. Kim, S.K.; Yoon, S.K.; Yoon, K.H. Crosstalk minimization in autostereoscopic multiveiw 3D display by eye tracking and fusion (overlapping) of viewing zones. In *SPIE Defense, Security, and Sensing*; International Society for Optics and Photonics: Bellingham, WA, USA, 2012; p. 838410.
5. Hanhart, P.; di Nolfo, C.; Ebrahimi, T. Active crosstalk reduction system for multiview autostereoscopic displays. In *Proceedings of the 2015 IEEE International Conference on Multimedia and Expo (ICME)*, Turin, Italy, 29 June–3 July 2015; IEEE: Biloxi, MS, USA, 2015; pp. 1–6.
6. Kim, S.K.; Yoon, K.H.; Yoon, S.K.; Ju, H. Defragmented image based autostereoscopic 3D displays with dynamic eye tracking. *Opt. Commun.* **2015**, *357*, 185–192. [[CrossRef](#)]

7. Van Berkel, C. Image preparation for 3D LCD. In *Stereoscopic Displays and Virtual Reality Systems VI*; International Society for Optics and Photonics: Bellingham, WA, USA, 1999; Volume 3639, pp. 84–92.
8. Woodgate, G.J.; Ezra, D.; Harrold, J.; Holliman, N.S.; Jones, G.R.; Moseley, R.R. Autostereoscopic 3D display systems with observer tracking. *Signal Process. Image Commun.* **1998**, *14*, 131–145. [[CrossRef](#)]
9. Borner, R.; Duckstein, B.; Machui, O.; Roder, H.; Sinnig, T.; Sikora, T. A family of single-user autostereoscopic displays with head-tracking capabilities. *IEEE Trans. Circuits Syst. Video Technol.* **2000**, *10*, 234–243. [[CrossRef](#)]
10. Kooima, R.L.; Peterka, T.; Girado, J.I.; Ge, J.; Sandin, D.J.; DeFanti, T.A. A gpu sub-pixel algorithm for autostereoscopic virtual reality. In Proceedings of the 2007 Virtual Reality Conference, Charlotte, NC, USA, 10–14 March 2007; IEEE: Biloxi, MS, USA, 2007; pp. 131–137.
11. Andiel, M.; Hentschke, S.; Elle, T.; Fuchs, E. Eye tracking for autostereoscopic displays using web cams. In *Electronic Imaging 2002*; International Society for Optics and Photonics: Bellingham, WA, USA, 2002; pp. 200–206.
12. Hopf, K.; Chojecki, P.; Neumann, F.; Przewozny, D. Novel autostereoscopic single-user displays with user interaction. In *Optics East 2006*; International Society for Optics and Photonics: Bellingham, WA, USA, 2006; p. 639207.
13. Peterka, T.; Kooima, R.L.; Girado, J.I.; Ge, J.; Sandin, D.J.; Johnson, A.; Leigh, J.; Schulze, J.; DeFanti, T.A. Dynallax: Solid state dynamic parallax barrier autostereoscopic VR display. In Proceedings of the 2007 Virtual Reality Conference, Charlotte, NC, USA, 10–14 March 2007; IEEE: Biloxi, MS, USA, 2007; pp. 155–162.
14. Yi, S.Y.; Chae, H.B.; Lee, S.H. Moving parallax barrier design for eye-tracking autostereoscopic displays. In Proceedings of the 2008 3DTV Conference: The True Vision-Capture, Transmission and Display of 3D Video, Istanbul, Turkey, 28–30 May 2008; IEEE: Biloxi, MS, USA, 2008; pp. 165–168.
15. Chappuis, A.; Rerabek, M.; Hanhart, P.; Ebrahimi, T. Subjective evaluation of an active crosstalk reduction system for mobile autostereoscopic displays. In *IS&T/SPIE Electronic Imaging*; International Society for Optics and Photonics: Bellingham, WA, USA, 2014; p. 90110S.
16. Leroy, L.; Fuchs, P.; Moreau, G. Visual fatigue reduction for immersive stereoscopic displays by disparity, content, and focus-point adapted blur. *IEEE Trans. Ind. Electron.* **2012**, *59*, 3998–4004. [[CrossRef](#)]
17. Sandin, D.J.; Margolis, T.; Ge, J.; Girado, J.; Peterka, T.; DeFanti, T.A. The Varrier TM autostereoscopic virtual reality display. *ACM Trans. Graph.* **2005**, *24*, 894–903. [[CrossRef](#)]
18. Bailer, C.; Henriques, J.; Schmitz, N.; Stricker, D. A Simple Real-Time Eye Tracking and Calibration Approach for Autostereoscopic 3D Displays. In Proceedings of the 2015 VISAPP: 10th International Conference on Computer Vision Theory and Applications, Berlin, Germany, 11–14 March 2015; Volume 3, pp. 123–129.
19. Lawton, G. 3D displays without glasses: Coming to a screen near you. *IEEE Comput. Soc.* **2011**, *44*, 17–19. [[CrossRef](#)]
20. Zhang, Z. A flexible new technique for camera calibration. *IEEE Trans. Pattern Anal. Mach. Intell.* **2000**, *22*, 1330–1334. [[CrossRef](#)]
21. Hwang, H.; Chang, H.S.; Nam, D.; Kweon, I.S. 3D Display Calibration by Visual Pattern Analysis. *IEEE Trans. Image Process.* **2017**, *26*, 2090–2102. [[CrossRef](#)] [[PubMed](#)]
22. Zhang, X.; Fang, Y.; Li, B.; Wang, J. Visual servoing of nonholonomic mobile robots with uncalibrated camera-to-robot parameters. *IEEE Trans. Ind. Electron.* **2017**, *64*, 390–400. [[CrossRef](#)]
23. Dodgson, N.A. Autostereoscopic 3D displays. *Computer* **2005**, *38*, 31–36. [[CrossRef](#)]
24. Hartley, R.; Zisserman, A. *Multiple View Geometry in Computer Vision*; Cambridge University Press: Cambridge, UK, 2003.
25. Moré, J.J. The Levenberg-Marquardt algorithm: Implementation and theory. In *Numerical Analysis*; Springer: Berlin, Germany, 1978; pp. 105–116.
26. Malis, E.; Vargas, M. Deeper Understanding of the Homography Decomposition for Vision-Based Control. Ph.D. Thesis, INRIA, Rocquencourt, France, 2007.
27. Konrad, J.; Agniel, P. Subsampling models and anti-alias filters for 3-D automultiscopic displays. *IEEE Trans. Image Process.* **2006**, *15*, 128–140. [[CrossRef](#)] [[PubMed](#)]
28. Wang, Z.; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P. Image quality assessment: From error visibility to structural similarity. *IEEE Trans. Image Process.* **2004**, *13*, 600–612. [[CrossRef](#)] [[PubMed](#)]
29. Buck, D.K.; Collins, A.A. POV-Ray—The Persistence of Vision Raytracer. Available online: <http://www.povray.org/> (accessed on 26 January 2018).

30. Tan, L.; Wang, Y.; Yu, H.; Zhu, J. Automatic camera calibration using active displays of a virtual pattern. *Sensors* **2017**, *17*, 685. [[CrossRef](#)] [[PubMed](#)]
31. Hwang, H.; Chang, H.S.; Kweon, I.S. Local deformation calibration for autostereoscopic 3D display. *Opt. Express* **2017**, *25*, 10801–10814. [[CrossRef](#)] [[PubMed](#)]
32. Huang, K.C.; Chou, Y.H.; Lin, L.C.; Lin, H.Y.; Chen, F.H.; Liao, C.C.; Chen, Y.H.; Lee, K.; Hsu, W.H. Investigation of designated eye position and viewing zone for a two-view autostereoscopic display. *Opt. Express* **2014**, *22*, 4751–4767. [[CrossRef](#)] [[PubMed](#)]
33. Hirschmuller, H.; Scharstein, D. Evaluation of cost functions for stereo matching. In Proceedings of the 2007 Computer Vision and Pattern Recognition, Minneapolis, MN, USA, 17–22 June 2007; IEEE: Biloxi, MS, USA, 2007; pp. 1–8.
34. Scharstein, D.; Hirschmüller, H.; Kitajima, Y.; Krathwohl, G.; Nešić, N.; Wang, X.; Westling, P. High-resolution stereo datasets with subpixel-accurate ground truth. In *German Conference on Pattern Recognition*; Springer: Berlin, Germany, 2014; pp. 31–42.
35. Stanford. The (New) Stanford Light Field Archive. Available online: <http://lightfield.stanford.edu> (accessed on 26 January 2018).
36. MIT. Synthetic Light Field Archive. Available online: <http://web.media.mit.edu/~simonw/SyntheticLightFields> (accessed on 26 January 2018).
37. Kim, C.; Zimmer, H.; Pritch, Y.; Sorkine-Hornung, A.; Gross, M.H. Scene reconstruction from high spatio-angular resolution light fields. *ACM Trans. Graph.* **2013**, *32*, 73. [[CrossRef](#)]



© 2018 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<http://creativecommons.org/licenses/by/4.0/>).