Depth from a Light Field Image with Learning-based Matching Costs

Hae-Gon Jeon, Student Member, IEEE, Jaesik Park, Member, IEEE, Gyeongmin Choe, Student Member, IEEE, Jinsun Park, Student Member, IEEE, Yunsu Bok, Member, IEEE, Yu-Wing Tai, Senior Member, IEEE, In So Kweon, Member, IEEE

Abstract—One of the core applications of light field imaging is depth estimation. To acquire a depth map, existing approaches apply a single photo-consistency measure to an entire light field. However, this is not an optimal choice because of the non-uniform light field degradations produced by limitations in the hardware design. In this paper, we introduce a pipeline that automatically determines the best configuration for photo-consistency measure, which leads to the most reliable depth label from the light field. We analyzed the practical factors affecting degradation in lenslet light field cameras, and designed a learning based framework that can retrieve the best cost measure and optimal depth label. To enhance the reliability of our method, we augmented an existing light field benchmark to simulate realistic source dependent noise, aberrations, and vignetting artifacts. The augmented dataset was used for the training and validation of the proposed approach. Our method was competitive with several state-of-the-art methods for the benchmark and real-world light field datasets.

Index Terms—Computational photography, light field imaging, depth estimation, 3D reconstruction, aberration correction.

1 INTRODUCTION

LIGHT-FIELD cameras collect light coming from different directions. They have broad potential impact because users can alter their point of view or focal plane after shooting. Since the camera array system [1] was introduced in the last decade, significant effort has been spent to develop compact and handy light field cameras. One example is the lenslet-based cameras [2], which utilize a micro-lens array placed in front of an imaging sensor. Because this design can be mass-produced, it has become the most popular design for hand-held light field cameras. Lytro [3], Lytro Illum [4], and Raytrix [5] are the representative cameras in this category.

A decoded lenslet light field is equivalent to numerous pairs of small-baseline stereo images. For this reason, depth map estimation via stereo matching is a natural application for lenslet light field imaging. The stereo matching measures the similarity (known as cost) between two patches extracted from each pair of stereo images. Examples of popular measurement approaches include the sum of absolute difference [6], normalized cross-correlation [7] and Census transform [8] methods.

Despite the success of those measurement techniques, questions remain about applying the global cost measure to the entire light field. The question is especially related to the optical limitations caused by the lenslet array: captured light field images have complicated degradations, including image vignetting, chromatic aberrations, and sensor noises. These characteristics are obvious in a decoded light field, and can even vary, in both the angular and the spatial domains. To the best of our knowledge, most previous studies have been designed with global photo-consistency [9], [10], [11], [12], [13], and validated using noise-free rendered images [14], [15] or with synthetic noise [16].

In this paper, we present a complete pipeline, devised to estimate a high-quality depth map from real light field images. We established a data driven approach that learns the best combination of photo-consistency measures, depending on the input light field. Our pipeline predicts a sub-pixel depth value from the most reliable matching cost. For the training, we augmented an existing light field dataset by reflecting realistic image noise, vignetting, and aberration artifacts. Our learning framework greatly benefited from the augmented dataset, and demonstrated enhanced performance with real light field images. The accuracy of the estimated depth map was comparable to state-of-the-art algorithms on the new depth from the light field benchmark [15].

This work builds upon our previous publication [11] and provides a few distinguishable improvements. In our previous work, matching costs were based on truncated absolute differences in image gradient and color. This work automatically predicts appropriate disparity values from various matching costs in Sec. 4.2. A cascade random forest is applied for this learning-based approach in Sec. 4.3. We create a training dataset by augmenting the light field benchmark [15] in Sec. 4.4. The augmentation step exploits the light-field image generation model [17], [18] and a practical degradation model that considers optical aberrations and the narrow baseline between sub-aperture images in Sec. 3. We demonstrate that the new approach is more reliable using real light field images in Sec. 5.

In addition, our previous approach [11] requires a number of post-processing steps that may depend on the scene.
Our new approach achieves better performance while using a simple weighted median filter. The algorithm was then validated using a recent light field dataset. We also provide a detailed analysis of the new framework, additional experiments on real light field images, and a demonstration of the 3D printing application.

2 RELATED WORKS

Previous works related to depth map (or disparity map) estimation from a light field image are reviewed below. Compared with conventional approaches to stereo matching, lenslet light field images have very narrow baselines. Therefore, previous approaches have been devised for small baseline correspondence matching, handling epipolar images, or adopting other cues and constraints.

Georgiev and Lumsaclaine [19] computed a normalized cross correlation between microlens images in order to estimate the disparity map. Bishop and Favaro [20] introduced an iterative method for a multi-view stereo image for a light field. Wanner and Goldluecke [21] used a structure tensor to compute the vertical and horizontal slopes in the epipolar plane of a light field image, and they formulated the depth map estimation problem as a global optimization approach that was subject to the epipolar constraint. Yu et al. [9] analyzed the 3D geometry of lines in a light field image and computed the disparity maps by line matching between the sub-aperture images. Tao et al. [10] introduced a fusion method that used the correspondences and defocus cues of a light field image to estimate the disparity maps. After the initial estimation, a multi-label optimization was applied in order to refine the estimated disparity map. Heber and Pock [22] estimated disparity maps using low-rank structure regularization to align the sub-aperture images.

More recently, Tao et al. extended their work [10] by adding a shading-based refinement technique, which yielded more accurate depth [12]. Wang et al. [13] proposed a depth estimation method which handles occlusions in the scene, especially pixels with depth discontinuity, by detecting occlusion edges and regularizing.

There have also been several works which target challenging scenes, including non-Lambertian objects with glossy surfaces. Wang et al. [23] presented a BRDF-invariant shape estimation method which uses differential motion theory. However, despite recovering the shape of the glossy surface, an inherent ambiguity still existed for shape and general reflectance. Tao et al. [24] reported a physics-based practical method for estimating the light source color and specularity using a line-consistency. Wiemann and Park [16] proposed a method that is robust to noise and occlusion. It was equipped with novel data costs using an angular entropy metric and adaptive defocus responses. Johannsen et al. [25] presented a technique which used an epipolar plane image patch to compose a dictionary with corresponding known disparity. This method yielded better results with multi-layered scenes.

In addition to the aforementioned, several algorithms have been designed for depth from high-resolution light-field images. For example, Kim et al. [26] utilized a DSLR camera on a linear stage to capture a high definition light field. Their approach simulated the multiple viewpoints of a light field image. Chen et al. [27] introduced a bilateral consistency metric for a surface camera in order to estimate the stereo correspondence in a light field image in the presence of occlusion. However, it should be noted that the baseline of the light field images presented in Kim et al. [26] and Chen et al. [27] were significantly larger than the baseline of the light field images captured using a lenslet light field camera.

Our previous work [11] computed a cost volume that was specially designed for sub-pixel stereo matching. The method used a phase shift theorem when performing the sub-pixel shift of a sub-aperture image. After constructing the cost volume, sequential steps consisting of cost aggregation, multi-label optimization and iterative refinement were applied. With those steps, parameters still needed fine-tuning, depending on scenes.

Here we present a learning-based approach that uses a cascade random forest to predict accurate depth values from various matching costs. This approach is especially effective for the lenslet-based light field as it can determine the most reliable matching cost depending on the angular and the spatial domain of a light field image. We designed and employed an augmented light field dataset for the training. As demonstrated in the experiments, the proposed algorithm is highly effective, and outperformed advanced algorithms.

3 SUB-APERTURE IMAGE ANALYSIS

In this section, we discuss the characteristics of sub-aperture images, and introduce our degradation correction approach. We utilized a Lytro camera for the experiments, as it is one of the most widely used lenslet-based light field cameras.

3.1 Noise and Vignetting

Fig. 1 provides an example of light field images captured with a Lytro camera. Sub-aperture images with slightly different views are extracted from the raw image. The characteristics of a micro-lens array allows it to capture angular information as well as spatial information, by splitting the incoming light into rays from different directions (Fig. 1(b)). At the same time, the micro-lens array hinders the penetration of light and increases photon noise. Also, the vignetting effect causes a radial fall-off in brightness for each micro lens. This is due to the foreshortening of light rays that are at oblique angles to the optical axis, and the obstruction of light by the stop or lens rim. Consequently,
noise distribution and the intensity level varies in the angular and spatial domains.

Previous works involving light field image denoising [28], [29], [30] have assumed additive Gaussian white noise with a signal independent standard deviation. However, this model does not hold for natural images captured by conventional imaging sensors [31], [32]. As lenslet-based light field cameras also use a conventional imaging sensor, the additive white noise model is not the optimal noise model.

In order to determine the appropriate noise model, we captured 15 images of a white planar scene under diffuse and uniform illumination. Based on the captured sequences, the statistics of the intensity of each pixel in the raw images were derived. As shown in Fig. 2, the standard deviation in pixel intensity is proportional to the square root of pixel intensity. This implies that the noise of the Lytro camera is actually signal-dependent, and can be fairly expressed by a linear regression. To generate optimal training data, the estimated slopes in Fig. 2 is used for adding signal-dependent Gaussian noise on the HCI light-field dataset [15].

To estimate vignetting, we generated sub-aperture images using the averaged image of a white plane sequence. Since the sub-aperture images are noisy due to demosaic error and photon noise, we fit the intensity of each sub-aperture image to a convex surface as shown in Fig. 3.

Fig. 2: Standard deviation versus normalized pixel intensity.

In this manner, each sub-aperture image has a look-up table which models the vignetting map. Based on these observations, we were able to model the noise distribution and vignetting of the Lytro camera to generate the training dataset in Sec. 4.4. We note that these look-up tables need to be updated whenever focal lengths, ISO, and zooms etc. are changed.

### 3.2 Narrow Baseline

The disparity range between adjacent sub-aperture views of the Lytro camera is known to be smaller than ±1 pixel [9]. Here, we review the lenslet light field camera projection formula proposed by Bok et al. [18].

\[
\frac{S}{T} = \frac{D}{d}(D + d) \left[ \frac{s/f_x}{t/f_y} \right],
\]

where \( D \) is the distance between the lenslet array and the center of the main lens, \( d \) is the distance between the lenslet array and imaging sensor, and \( f \) is the focal length of the main lens. With the assumption of a uniform focal length (i.e. \( f_x = f_y = f \)), the baseline between two adjacent sub-aperture images is defined as baseline = \( \frac{(D + d)D}{df} \).

Based on this model, we need to shorten \( f \), shorten \( d \), or lengthen \( D \) for a wider baseline. However, \( f \) cannot be too short because it is proportional to the angular resolution of the micro-lenses in the lenslet array. Therefore, the maximum baseline, which is the product of the baseline and angular resolution of the sub-aperture images, remains unchanged even if the value of \( f \) varies. If the physical size of the micro-lenses is too large, the spatial resolution of the sub-aperture images is reduced. Lengthening \( D \) increases the baseline, but the field of view is reduced. Due to these challenges, the scene disparity is often less than one pixel. This motivates us to design an accurate sub-pixel image shifting for cost volume construction in Sec. 4.2.

### 3.3 Distortion Estimation and Correction

From the analyses conducted in this study, it was observed that the lenslet light field images contain optical distortions caused by both the main lens (thin lens model) and micro-lenses (pinhole model). Although the radial distortion of the main lens can be calibrated using conventional methods, it is imperfect, particularly for light rays that have large angular differences from the optical axis. The distortion caused by these rays is called *astigmatism* [33]. Moreover, because the conventional distortion model is based on a pinhole camera model, the rays that do not pass through the center of the main lens cannot be fit well to the model. The distortion caused by these rays is called field curvature [33]. These two distortions are the primary causes of the depth distortion, and they are compensated by the approach described in this section.

During the capture of a light field image of a planar object which is supposed to have constant depth, spatially variant epipolar plane image (EPI) slopes (i.e., non-uniform field cameras also use a conventional imaging sensor, the

1. The 4D parameterization [2], [19], [21] is followed, where the pixel coordinates of a light field image \( I \) are defined using the 4D parameters of \((s, t, x, y)\). Here, \( s = (s, t) \) denotes the discrete index of the angular directions and \( x = (x, y) \) denotes the Cartesian image coordinates of each sub-aperture image.
Because image plane and calibration pattern are in parallel, EPI slopes should be consistent in entire area. (a) shows varying EPI slopes and (b) shows consistent EPI slopes for all area. (c) Our compensation process for a pixel. (d) Zoom-in image of (a) and (b) that displays slope difference between before and after distortion compensation.

To measure and compensate the distortion, we conduct the following experiment. First, a light field camera is mounted on an optical table to be parallel to a planar checkerboard. Second, the EPI slopes of the planar checkerboard are compared with \(\theta_o(\cdot)\). Points with strong gradients (i.e., the checkerboard edges) in the EPI are selected to measure the slope difference \((\theta(\cdot) - \theta_o)\) used in Eq. 2. Without distortions, the slope difference should be zero for every pixel, because we assume that the checkerboard has uniform depth. However, because of distortions, nonzero depth differences exist for the off-center pixels. The difference is compensated by considering distortion map \(G\).

We use a second order polynomial surface model for \(G\):

\[
G(x) = g_{00} + g_{10}x + g_{01}y + g_{11}xy + g_{20}x^2 + g_{02}y^2,
\]

where \(x = (x, y)^T\). Note that \(G\) can cover non-symmetric surfaces although most existing algorithms assume that the distortion is symmetric about its axis of distortion (e.g., image center). The reason is that some unexpected errors such as the misalignment between lens and image sensor can occur during the manufacturing process. This can affect the symmetry of the distortion in an imaging system (i.e., decentering [34]). Although we do not explicitly take those errors into account, our non-symmetric model can deal with such undesirable errors.

Fig. 5 shows an example of the estimated distortion map. The sparse observation of distortion is shown with green dots. This is obtained from the EPI slope of each edge point. The entire distortion map is fitted with Eq. (2) using the sparse observations. After solving Eq. (2), each points EPI slope is compensated using \(\hat{G}\). The pixel for the reference view (i.e., the center view) is set as the pivot of rotation, and the pixels in the other views are rotated according to the amount of distortion (see Fig. 4 (c)). However, due to the astigmatism, the field curvature varies depending on the slice direction.

In order to address this problem, Eq. (2) is solved twice: for the horizontal and the vertical directions. Because the distortion estimation for each direction is performed using individual uncompensated images, the correction order does not affect the compensation result. In order to avoid chromatic aberrations, the distortion parameters are estimated for each color channel. Fig. 4 and Fig. 6 present the EPI and estimated depth map before and after the proposed distortion correction, respectively. The depth distortion is significantly reduced after the compensation.

Strictly speaking, \(G\) is a function of depth because \(\theta_o\) in Eq. (2) depth dependent. However, depth dependency of \(G\) is not obvious in practice if the focal length is fixed as a manufacturer default. From our experiments, at the distance of focal plane, \(G\) is almost unchanged within the distance range for capturing sharp image. As a rule of thumb, we use the default camera focal length for all the experiments conducted in this paper.

Note that the proposed method is classified as a low order approach which targets the astigmatism and field curvature. A more general technique for correcting the aberration has been proposed by Ng and Hanrahan [35], and it is currently used for real products [36]. An approach by Johannsen et al. [37] estimated the micro-lenses distortion by minimizing a reprojection error. The algorithm was designed for a focused light-field camera like Raytrix [5]. Our approach is demonstrated for a light field camera with
a narrower baseline than Raytrix, such as Lytro\textsuperscript{2}.

4 Depth Map Estimation

Using the distortion-corrected sub-aperture images, we then estimate accurate dense depth maps. The proposed depth map estimation algorithm is based on cost volume based stereo matching [6].

The pipeline is tailored with three significant differences. First, instead of traversing the local patches to compute the cost volume, the sub-aperture images are directly shifted using a phase shift theorem to manage the narrow baseline between the sub-aperture images. Second, in order to effectively find correspondences between the sub-aperture images, we build several candidate costs (SAD, ZNCC, Census, GRAD, and their linear combinations). Third, a cascade random forest based framework is proposed to determine the best matching cost among the candidates. The different candidates are automatically chosen for the different angular light field image domains. The subsequent random forest predicts the final depth label.

4.1 Phase Shift based Sub-pixel Displacement

One of the key ideas of our depth estimation algorithm is the matching of the narrow baseline sub-aperture images using sub-pixel displacements. According to the phase shift theorem, if an image \( I \) is shifted by \( \Delta \in \mathbb{R}^2 \), the corresponding phase shift in the 2D Fourier transform is:

\[
\mathcal{F}\{I(x+\Delta)\} = \mathcal{F}\{I(x)\} \exp^{2\pi i \Delta},
\]

where \( \mathcal{F}\{\cdot\} \) denotes the discrete 2D Fourier transform. In Eq. (4), multiplying the exponential term in the frequency domain is the same as convolving a Dirichlet kernel (or periodic sinc) in the spatial domain. According to the Nyquist-Shannon sampling theorem [38], a continuous band-limited signal can be perfectly reconstructed by convolving it with a sinc function. If the centroid of the sinc function deviates from the origin, precisely shifted signals can be obtained. In the same manner, Eq. (4) generates a perfectly shifted image in the spatial domain if the sub-aperture image is band-limited. The sub-pixel shifted image is obtained using:

\[
I(x + \Delta) = \mathcal{F}^{-1}\{\mathcal{F}\{I(x)\} \exp^{2\pi i \Delta}\}.
\]

2. Let’s recall Eq. (1) and assume Lytro and Raytrix has the same focal length of main lens. The two cameras have different intrinsic parameters \( D \). The micro-lens array of Raytrix is at the focal plane \( (D = f) \) while that of Lytro (and Lytro Illum) is in front of the focal plane \( (D < f) \). Therefore, baseline between adjacent aperture image of Raytrix is larger than that of Lytro.

In practice, the light field image is not strictly a band-limited signal. This results from the weak pre-filtering, which fits the light field into the sub-aperture image resolution [39], [40]. However, the artifact is not obvious for the regions where the texture is obtained from the source surface in the scene. For example, a sub-aperture image of a resolution chart captured by a Lytro camera is presented in Fig. 7. This image is shifted by \( \Delta = [2.2345, -1.5938] \) pixels. Compared with the bilinear and bicubic interpolations, the sub-pixel shifted image using the phase shift theorem is sharper and does not contain blurriness. Note that having an accurate reconstruction of sub-pixel shifted images is important for accurate depth map estimations, particularly when the baseline is narrow.

For implementation, a fast Fourier transform with a circular boundary condition was used to manage the non-infinite signals. Note that we do not explicitly generate sub-pixel shifted patches. Instead, we shift the entire sub-aperture image, extract the local region, and consider it to be a sub-pixel shifted local patch. Therefore, the artifacts that result from periodicity problems only appear at the boundary of the sub-pixel shifted sub-aperture image within the width of two pixels, which is negligible.

4.2 Building the Matching Costs

In stereo matching, there are many matching costs that can have advantages in different situations. Based on a study about effective matching costs [41], we exploit four matching costs to find correspondences between sub-aperture images: the sum of absolute differences (SAD), the sum of gradient differences (GRAD), zero-mean normalized cross correlation (ZNCC) and Census transform [8]. In addition to these terms, we define the linear combinations of these costs.

4.2.1 Sum of Absolute Differences (SAD)

The SAD cost \( C_A \) is defined as

\[
C_A(x, l) = \sum_{s \in \mathcal{N}} \sum_{w \in \mathcal{W}} \min(|I_c(w) - I_s(w_\Delta)|, \tau_1),
\]

where \( l \) is sub-pixel depth label, \( \mathcal{N} \) contains the \( st \) coordinate pixels \( s \) except for the center view \( c \). \( \mathcal{W}_s \) is a set of the adjacent pixels of \( x \) in a small rectangular window centered at \( x \). \( I_c \) and \( I_s \) are sub-aperture images of the central view and non-central view, \( \tau_1 \) is a truncation value of a robust function. \( w_\Delta \) in Eq. (6) is the \( xy \) coordinate, and depends on depth label \( l \) and the viewpoint of \( s \):

\[
w_\Delta = w + \Delta(s, l)
\]

The 2D shift vector \( \Delta \) according to \( s, l \) is defined as follows:

\[
\Delta(s, l) = l(k(s - c)),
\]

where \( k \) is the unit of the label in pixels. \( \Delta \) linearly increases with the angular deviation from the reference view.

Equation (6) builds a matching cost by comparing the center sub-aperture image \( I_c \) with the other sub-aperture images \( I_s \) to generate a disparity map from a canonical viewpoint. Summing over a fixed window acts like averaging, which reduces the noise effect for all costs. Equation (4) is used for precise sub-pixel shifting of the images.
4.2.2 Zero-mean Normalized Cross Correlation (ZNCC)
The ZNCC score is calculated by subtracting the mean of the window and computing the normalized cross-correlation. This compensates for differences in both gain and offset within the correlation window, and is defined as:

$$C_Z(x, l) = \frac{\sum_{w \in W_x} N_c(w) N_s(w_{\Delta})}{\sqrt{\sum_{w \in W_x} N_c(w)^2 \sum_{w \in W_x} N_s(w_{\Delta})^2}},$$  

(9)

where $N_c(w) = I_c(w) - \mu_c(x)$, $N_s(w_{\Delta}) = I_s(w_{\Delta}) - \mu_s(x_{\Delta})$, and $\mu_c(x)$ and $\mu_s(x_{\Delta})$ are mean of the window centered in $x$ of $I_c$ and mean of window centered at $x_{\Delta}$ in $I_s$, respectively.

4.2.3 Census Transform
The census transform encodes the relative ordering of the pixel intensity values rather than the intensity values. It is realized with a comparison function $b$ which outputs a bit:

$$b(w) = \begin{cases} 1 & \text{if } I(w) > I(x) \\ 0 & \text{otherwise} \end{cases} \quad \text{s.t. } w \in W_x,$$

(10)

After vectorizing the census-transformed pixels in Eq. (10) into $b(I, x)$, the affinity of the two vectors are measured with hamming distance $H(\cdot, \cdot)$. $H$ conducts XOR of two bit vectors and numerates non-zero bits. The cost volume is constructed as follows:

$$C_C(x, l) = H(b(I_c, x), b(I_s, x_{\Delta})).$$

(11)

According to the study in [41], the census transform tolerates radiometric distortions that do not change the local ordering of intensities, and provides better performance than ZNCC. However, its performance is degraded in the presence of strong image noise which often occurs at the boundary views of sub-aperture images.

4.2.4 Sum of Gradient Differences (GRAD)
GRAD is not used on its own, but is utilized in synergy with other matching costs by imposing higher weights at the edges of images. The cost volume from GRAD $C_G$ is defined as follows:

$$C_G(x, l) = \sum_{s \in V} \sum_{w \in W_x} \beta(s) \min \{\text{Diff}_y(c, s, w, l), \tau_2\}$$

$$+ (1 - \beta(s)) \min \{\text{Diff}_x(c, s, w, l), \tau_2\}$$

(12)

where $\text{Diff}_x(c, s, w, l) = |\partial I_c(x)/\partial x - \partial I_s(w_{\Delta})/\partial x|$ denotes the absolute difference between the $x$-directional gradient of the sub-aperture images. $\text{Diff}_y$ is defined similarly on the $y$-directional gradients. $\tau_2$ is a truncation constant that suppresses outliers. $\beta(s)$ in Eq. (12) controls the relative importance of the two directional gradient differences based on the relative $st$ coordinates. $\beta(s)$ is defined as follows:

$$\beta(s) = \frac{|s - s_c|}{|s - s_c| + |t - t_c|}.$$  

(13)

According to Eq. (13), $\beta$ increases if the target view $s$ is located at the horizontal extent of the center view $s_c$. In this case, only the gradient costs in the $x$ direction are aggregated to $C_G$. Note that $\beta$ is independent of the scene because it is determined purely using the relative position between $s$ and $s_c$.

4.2.5 Combinations of Matching Costs
Many works [42], [43], [44] have demonstrated the effectiveness of combining matching costs, and reported better accuracy. With the same observation, we design a framework that automatically determines the best combination of matching costs that is resilient to the challenging input image. Based on the study [41], we define the most effective matching combination candidates (1) SAD-GRAD, (2) Census-GRAD, (3) SAD-Census as follows:

$$C_{A-G}(x, l) = \alpha C_A(x, l) + (1-\alpha)C_G(x, l),$$

(14)

$$C_{C-G}(x, l) = \alpha C_C(x, l) + (1-\alpha)C_G(x, l),$$

(15)

$$C_{A-C}(x, l) = \alpha C_A(x, l) + (1-\alpha)C_C(x, l),$$

(16)

where $\alpha$ adjusts the relative importance between two complementary costs. We discretize $\alpha$ to systematically determine the best combination. In our implementation, $\alpha \in [0.1, 0.9]$ with an interval of 0.1 which corresponds to 9 costs per pairwise combination. In addition to 4 costs from $C_A, C_Z, C_C$ and $C_G$, 27 combinatorial costs brings us to 31 matching costs in total.

4.3 Depth Label Prediction
We designed a learning framework that automatically selects the most reliable depth label from 31 matching costs. We assigned this task to an ensemble of random forests, which have previously shown promising performance in pixel-wise classification [45], [46], [47], [48]. This success is due to the nature of random forests: the robustness of outliers, nonparametric properties, and ranking the importance of input variables [49]. We build two random forests for classification and regression purposes: the classification forest selects influential and powerful costs, and the regression forest predicts depth values from the selected matching cost. The overall procedure is described in Fig. 8.

4.3.1 Feature Vector
We define the high-dimensional feature vector $q$ for a pixel $x$, given the 31 cost volumes. A feature vector consists of depth labels that are obtained from a winner-take-all strategy:

$$q_9(x) = [D_{g_1}(x), D_{g_2}(x), \ldots, D_{g_{31}}(x)].$$

(17)

$D_1$ to $D_4$ are depth labels obtained from matching costs $C_A, C_Z, C_C$ and $C_G$, respectively. Depth labels $D_5$ to $D_{13}$ are obtained from $C_{A-G}$ in Eq. (14), $D_{14}$ to $D_{22}$ are obtained from $C_{C-G}$ in Eq. (15). $D_{23}$ to $D_{31}$ are obtained from $C_{A-C}$ in Eq. (16). The subscript $g$ in Eq. (17) represents the matching group index. The group divides the cost volumes according to the distances from the central position, which is equivalent to the angular deviation. For example, if $|s - c| < 2$, the corresponding cost volumes belong to matching group 1. In this paper, since we used $9 \times 9$ sub-aperture images, the number of matching groups is 4. Therefore, a 124 dimensional feature vector consists of:

$$q = [q_1, q_2, q_3, q_4].$$

(18)

The matching group was devised to explicitly adapt the learning framework for both the radial fall-off in brightness and the noise level, which are proportional to the distance between the center view and target views. The matching group is illustrated in Fig. 8 (a).
4.3.2 Depth Label Selection

From the 124 candidate depth labels, the best depth is chosen by cascade random forest: one is for supervised classification and another is for regression.

Classification forest. This forest outputs an importance measure, and 20 reliable depth values are selected using the measure. The training set \( Q \) contains a set of \((q, l_{gt})\) where \( q \) is feature vector of \( x \), and \( l_{gt} \) is its ground truth depth label. Starting at the root node, a set of binary split function parameters \( \varphi \) are proposed at random. During training, the cross-entropy \( \mathcal{E}(Q, \varphi) \) of the discrete distribution \( p(l|Q, \varphi) \) is minimized:

\[
\arg\min_{\varphi} \mathcal{E}(Q, \varphi), \quad \mathcal{E}(Q, \varphi) = -\sum_{l \in L} p(l|Q, \varphi) \log p(l|Q, \varphi),
\]

where \( L \) is the number of depth labels. We define a soft penalty for a maximum 2 levels of deviation from the ground truth. This is equivalent to a smooth target distribution [50].

\[
p(l|Q, \varphi) \rightarrow \begin{cases} 
\lambda_1 & \text{if } |l - l_{gt}| = 0 \\
\lambda_2 & \text{if } |l - l_{gt}| = 1 \\
\lambda_3 & \text{if } |l - l_{gt}| = 2 \\
\epsilon, & \text{otherwise}
\end{cases}
\]

We empirically set \( \{\lambda_1, \lambda_2, \lambda_3\} = \{0.6, 0.3, 0.1\} \). \( \epsilon = 10^{-6} \) is defined for the numerical stability of Eq. (20).

A trained forest can output the permutation importance measure [49] which indicates the amount of importance of each feature element. One example of the measure is illustrated at the bottom of Fig. 8 (b). This helps to identify less important matching costs.

The permutation importance measure for the \( k \)-th element of \( q \) is computed as follows. The out-of-bag error of \( q \), or \( e(q) \) is computed. \( k \)-th element of \( q \) is replaced with the training data to make \( q^k \). The out-of-bag error \( e(q^k) \) measured. Compute the difference between \( e(q) \) and \( e(q^k) \). Average the difference over all the trees. The sum of the difference is normalized using the standard deviation of the differences. The normalized difference is the importance measure.

We choose the top \( N \) depth labels based on the permutation importance measure, and make a lower dimensional feature vector \( \tilde{q} \) as follows:

\[
\tilde{q} = [q_{R1}, q_{R2}, \ldots, q_{RN}],
\]

where \( q_{R1} \) indicates element of \( q \) that receives the largest importance measure. In this way, the classification forest picks high fidelity depth labels. This procedure is equivalent to picking most reliable matching cost from all the candidate costs. This scheme also helps to give reliable information to the regression forest.

Regression forest. The same paradigm as the random forest is applied to the regression task. Given \( \tilde{q} \), the importance measure for each element is estimated, and is than used to generate the weighted sum of \( \tilde{q} \)'s elements for the final depth prediction. In this manner, the regression forest produces continuous depth. Although the depth value used for the training is not continuous but discrete in the sub-pixel domain, the forest can be trained to produce a depth value that minimizes prediction error.

As shown in Fig. 9, we choose \( N = 20 \) for the regression forest. A higher \( N \) produces little improvement in the bad pixel ratio and the mean square error (by only 0.001 and \( 2 \times 10^{-6} \) pixels) on the average of 8 synthetic images of light field benchmark [15].

Forest configuration. In practice, we used 7 trees for the classification forest and 5 trees for the regression forest with 1.3 million examples of training set \( Q \). We observed
that random forests with more than 20 trees were a better choice for validation, but it was necessary to sample only a few percent of the training sets due to limited memory space. Our choice is the most efficient configuration that can fully cover the entire training set used without loss of performance.

**Post processing.** As a sequential step, the predicted depth is refined using a weighted median filter \[51\] to alleviate the sparse unreliable matches, as shown in Fig. 8 (c). Here, the central sub-aperture image is used to determine the color weights used for the filter.

### 4.3.3 Analysis of Effective Matching Costs

We discuss effective costs from the perspective of 1) the angular viewpoint groups (matching group) described in Sec. 4.3.1 and 2) spatial distribution of preferred cost according to the scene configuration.

#### By matching group.

In matching group 1, the neighboring views have almost the same level of pixel intensity and noise compared to the reference view. In this group, the Census transform in Eq. (11) performed very well. This is because the ordering of intensities is almost unchanged in this group. The combination of SAD and Census transform in Eq. (16) is also a powerful matching cost. As the distance from the reference view increases, the noise level increases and a local radiometric consistency does not hold due to the vignetting effect. In the cases of matching group 2 and 3, a combination of SAD and GRAD in Eq. (14) is an effective matching cost. SAD and GRAD sum over a window like averaging, which reduces the effect of additive noise for all costs. GRAD makes up for the weakness of SAD, which typically occurs near depth discontinuities and leads to blurred object borders. ZNCC in Eq. (9) also shows good performances by compensating for local gain and offset changes. In the case of matching group 4, the sub-aperture images suffer from severe noise and gain changes. For these images, GRAD in Eq. (12) performs better than other matching costs. This implies that the consideration of image edges is effective in the presence of strong image noise and large radiometric changes.

Through this observation and an example shown in Fig. 11, we observed that the selective utilization of various matching costs is more beneficial than single and global matching costs.

#### By scene configuration.

As another aspect of the analysis, we visualize the effective cost according to scene configurations. For this analysis, we compute the mean square error between a predicted depth from the random forests and the depth values from candidate 31 costs. The matching cost that has the smallest error is chosen by the random forest. The error is independently measured for each pixel. As shown in Fig. 12, we categorize the matching costs into 7 classes for visualization: SAD, SAD+Census, Census, Census+GRAD, GRAD, GRAD+SAD and ZNCC.

We observed a correlation between the most effective
Fig. 13: Overview of our data augmentation pipeline. Based on the lenslet-based light field analysis discussed in Sec. 3, we apply vignetting and source-dependent noise. The dataset is used for the training. Detailed description is in Sec. 4.4.

matching cost and various scene configurations, such as textured, textureless, and depth discontinuity regions. SAD and the SAD+Census are selected as influential matching costs in the weakly textured regions. The depth value in the strong texture or depth discontinuity regions is predominantly provided by Census+GRAD or ZNCC. Those matching costs impose a higher weight on a patch with a larger image gradient.

4.4 Training Data Generation

To generate a training dataset, we augmented the existing synthetic light field benchmark [15]. The augmentation added realistic image degradation to the synthetic images. This augmented dataset bears a close resemblance to real light field camera images.

The following is the procedure for generating realistic light field images. First, the estimated vignetting was applied to the sub-aperture images, as shown in Fig. 13 (a) and (b). Next, we sampled pixels from a location identical to the sub-aperture images, and collected the pixels on a rectangle grid with 9×9 pixels. For this step, we followed the lenslet-based light field camera calibration process [17], [52]. The lenslet image was then converted to the RAW format which is a gray-scale with a BGGR Bayer pattern, to store the values of the different RGB channels (Fig. 13 (c)). The micro-lens array has hexagonal shaped packing to reduce gaps between the micro-lenses, as shown in Fig. 13 (c). Because we focused on emulating the errors associated with demosaicing, the packing shape was not considered for data augmentation. After adding the estimated noise model to this RAW image (Fig. 13 (d)), the RAW image was demosaiced using a conventional linear method (Fig. 13 (e)). We then rearranged the color micro image into sub-aperture images (Fig. 13 (f)).

To validate the effectiveness of our generated training data, we compared the results from three different random forests. FOREST CLEAN: this was trained with clean images. FOREST GAUSSIAN: this was trained with conventional Gaussian noise. FOREST AUGMENTATION: this forest was trained with our augmented data. As shown in Fig. 14, FOREST GAUSSIAN shows better results than FOREST CLEAN because of the data augmentation effect. However, FOREST GAUSSIAN failed to predict precise depth values in low intensity regions. This might be because both the vignetting effect and the signal dependent noise model were not learned. In contrast, FOREST AUGMENTATION predicted better depth than the other forests.

5 Experimental Results

5.1 Configuration

Parameters. In our experiments, the size of window $W$ was set to 3×3 for SAD and GRAD, and to 5×5 for ZNCC.
Table 1: Comparison on light field benchmark [15]. (red = best, green = second best)

<table>
<thead>
<tr>
<th>Method</th>
<th>Bad Pixel Ratio (≥ 0.07 pixel)</th>
<th>Training Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wanner and Goldluecke [21]</td>
<td>23.71</td>
<td>AVG</td>
</tr>
<tr>
<td>Wang et al. [13]</td>
<td>15.21</td>
<td>Dots</td>
</tr>
<tr>
<td>Johannsen et al. [25]</td>
<td>24.22</td>
<td>Pyramids</td>
</tr>
<tr>
<td>Zhang et al. [53]</td>
<td>11.13</td>
<td>Stripes</td>
</tr>
<tr>
<td>Proposed</td>
<td>9.04</td>
<td>Boxes</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Cotton</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Dino</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Sideboard</td>
</tr>
</tbody>
</table>

Mean Square Error (Multiplied with 100)

<table>
<thead>
<tr>
<th>Method</th>
<th>Mean Square Error</th>
<th>Training Method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Wanner and Goldluecke [21]</td>
<td>7.07</td>
<td>AVG</td>
</tr>
<tr>
<td>Jeon et al. [11]</td>
<td>8.84</td>
<td>Backgammon</td>
</tr>
<tr>
<td>Wang et al. [13]</td>
<td>6.20</td>
<td>Dots</td>
</tr>
<tr>
<td>Johannsen et al. [25]</td>
<td>4.30</td>
<td>Pyramids</td>
</tr>
<tr>
<td>Zhang et al. [53]</td>
<td>4.57</td>
<td>Stripes</td>
</tr>
<tr>
<td>Proposed</td>
<td>4.22</td>
<td>Boxes</td>
</tr>
<tr>
<td></td>
<td>7.14</td>
<td>Cotton</td>
</tr>
<tr>
<td></td>
<td>7.96</td>
<td>Dino</td>
</tr>
<tr>
<td></td>
<td>0.11</td>
<td>Sideboard</td>
</tr>
</tbody>
</table>

Fig. 16: Evaluation on light field benchmark [15]. For Pyramids and Dino dataset, the left most column shows ground truth, and the rest of the column shows color coded disparity maps and bad pixels. Bad pixels are colored in red.

and the Census transform. Both of the truncation values \( \tau_1 \) in Eq. (6) and \( \tau_2 \) in Eq. (12) were set to 4 for \( I \in [0, 255] \). As shown in Fig. 15, we performed cross-validations for these choices. The bad pixel ratio was computed using disparity maps extracted by a winner-take-all strategy without any post processing. As shown in Fig. 15 (a), \( 5 \times 5 \) for SAD and GRAD was slightly better than \( 3 \times 3 \) with the amount of 1%. We used \( 3 \times 3 \), as it significantly reduced the time to make cost volumes for the training step.

For \( \tau_1 \) and \( \tau_2 \), we also computed the bad pixel ratio by sweeping the parameters. In Fig. 15 (b), the smallest value of \( \tau \) showed the best performance. Unlike conventional stereo matching, we observed that the variances in SAD and GRAD cost were relatively small because the baseline between the sub-aperture images is very narrow. Thus, our multi-view stereo matching-based method needs small truncation values to find correspondence between similar intensity values. Note that \( k \) in Eq. (8) varied according to the baseline of the dataset. We set \( k \) to 0.03 and 0.02 for the synthetic and real images, respectively.

Computational time. The process of distortion estimation and distortion correction for a \( 9 \times 9 \times 383 \times 552 \) sub-aperture image takes 40 and 10 minutes, respectively. This is a calibration step performed once. To generate the training set using the benchmark images [15], it took about 20 minutes to make one high-dimensional feature vector with 151 labels. We used 10 synthetic images. The random forests took about 25 minutes to train using an Intel i7 3.40 GHz CPU and 16 GB RAM with parallel computation. The inferencing process was relatively faster: it took less than 0.5 seconds for a Lytro Illum image. Our implementation was written in MATLAB. We will release source codes and datasets upon acceptance of the paper.
Fig. 17: Qualitative evaluations on light field images captured by a Lytro Illum camera. (a) Center view images. (b) Wanner and Goldluecke [21]. (c) Yu et al. [9]. (d) Jeon et al. [11]. (e) Williem et al. [16]. (f) Wang et al. [13]. (g) Tao et al. [12]. (h) Proposed method.

5.2 Quantitative Evaluations

A quantitative evaluation was performed using the synthetic light field benchmark provided by Honauer et al. [15], to compare the proposed method to five state-of-the-art methods: Wanner and Goldluecke [21], Jeon et al. [11], Wang et al. [13], Johannsen et al. [25], and Zhang et al. [53]. The synthetic light field images were classified into two types: Stratified scenes representing a variety of characteristics such as foreground fattening, texture sensitivity and robustness to noise, and Training scenes to simulate complex real-world scenes. The dataset has $9 \times 9$ angular resolution and $512 \times 512$ spatial resolution. The disparity between two adjacent sub-aperture images in the $st$ domain was smaller than $\pm 2$ pixels. For the state-of-the-art methods, we directly report the bad pixel ratio, which denotes the percentage of pixels whose disparity error is larger than 0.07 pixels, as well as the mean square error (MSE).

Experimental results are shown in Table 1. The proposed method achieved the best performance on average in terms of bad pixel ratio. In particular, the proposed method

3. We follow the standard provided with the benchmark [15]. The Benchmark webpage is: http://hci-lightfield.iwr.uni-heidelberg.de/
estimated accurate disparities on a stratified dataset with various shapes and textures, as shown in Pyramids of Fig. 16. This is because our random forests learned the various characteristics as well as the light field image noise, and were able to infer disparity with sub-pixel precision. For the training set, the proposed method showed the second best performance on average. Most of the errors in Dino of Fig. 16 were in regions with depth discontinuities, because the proposed method was not designed to handle occlusion.

Although the proposed method also showed the best results in terms of MSE, the performance improvement was marginal compared to Johannsen et al. [25]. We suppose that the sub-pixel disparities have errors due to the quantized disparity values of the regression forest. Using a smaller \( k \) can alleviate the problem, but this increases computational complexity. An efficient implementation using a GPU is needed in this regard.

### 5.3 Qualitative Evaluations

Fig. 17 presents a qualitative comparison of four real-world Lytro Illum images using the proposed method and the state-of-the-art methods: Wanner and Goldluecke [21], Yu et al. [9], Jeon et al. [11], Willemin et al. [16], Wang et al. [13], and Tao et al. [12]. Sub-aperture images of the images were obtained using the calibration toolbox provided by Bok et al. [18]. Since the method of Tao et al. [12] does not need sub-aperture images, lenslet images were used for it. We ran the authors’ source codes or executable files, and parameters were carefully swept for fair comparisons.

As shown in Fig. 17 (c) and (e), the methods of [9] and [16] failed to estimate accurate depth maps, since their algorithm is not intended for small baselines. The work in [12] achieved relatively good results, possibly because of defocus cues. However, the results tended to be over-smooth due to severe matching cost noise. We also observed that the depth results were radially distorted, because these methods use lenslet images without distortion correction. While the well-designed occlusion model of Wang et al. [13] showed good layer separation results, the matching costs from the variance and mean of the image patches, which were similar to [10], did not lead to high depth resolution. Wanner and Goldluecke [21] computed the elevation angles of lines in the epipolar plane images using structured tensors. On challenging Lytro images, this scheme may result in noisy depths, even if the weight of the smoothness term is increased for multi-label optimization. In addition, severe noise in the lenslet light field images hinders the construction of accurate epipolar plane images.

As shown in Fig. 17 (d) and (h), calculating the exact sub-pixel shift using the phase shift theorem in [11] and the proposed method improves the matching quality. Work in [11] utilized multi-label optimization and refinement to handle outliers of the initial disparity. However, this was not successful at preserving fine details. On the other hand, the proposed method was able to tolerate significant outliers because it selected powerful matching costs, using the classification forest with a smooth target distribution. Thanks to the selected matching costs, the regression forest predicted accurate depth values, even though the objects were composed of various materials, such as plastic, plaster and ceramics.

Additionally, we apply our algorithm to a high resolution light field image dataset shown in Fig. 18 to demonstrate the versatility of the proposed method. The dataset consists of 151 images with \( 1718 \times 2622 \) image resolution. It is captured by DSLR camera mounted on the linear stage. Since the baseline of the light field image is broader than that of the Lytro Illum, we set \( k \) as 0.5. We only used 9 images among the original 151 images. As shown in Fig. 18, the proposed method shows a depth estimation result of similar quality.

### 5.4 Application to 3D Printing

As shown in Fig. 19, we captured an object of 1.2 cm thickness and estimated its disparities. Although the spacing of each layer in Fig. 19 (b) was very narrow, the proposed method successfully handled it. The disparity map was converted into a 3D model in the metric scale using the calibration parameters estimated by the toolbox of Bok et al. [18]. The 3D mesh and the printed result produced by a commercial 3D printer are displayed in Fig. 19 (c) and (d), respectively.

### 6 Conclusion

In this paper, we have presented a novel method for estimating high quality depth data from a single light field image. Our work makes three main contributions. First, we analyzed the characteristic of light field images in terms of
noise, aberrations and the narrow baseline between sub-aperture images. Second, we have proposed a method for correcting aberrations to alleviate depth distortion. Last, a novel method of sub-pixel-wise disparity estimation was proposed. To accomplish this, we utilized the sub-pixel shift in the frequency domain and proposed a learning-based matching cost selection. The effectiveness of the proposed method was verified on various synthetic and real-world datasets.

Nevertheless, there are several issues that can be addressed in future work. Although our printed example in Fig. 19 demonstrated the potential of light field cameras, it is inevitable that our cost volume construction might have a quantization error. Our approach also suffers with low-texture and refractive media as shown in Fig. 20. We believe that additional photometric cues such as shading [12] are the key of future researches. We also expect to take advantage of a convolutional neural network for depth prediction [54] and cost computation [55] to improve the performance.

References


