# Exploring Positional Characteristics of Dual-Pixel Data for Camera Autofocus

Myungsub Choi Hana Lee Hyong-euk Lee
Samsung Advanced Institute of Technology (SAIT)
{ms00.choi, hana.hn.lee, hyongeuk.lee}@samsung.com

## Abstract

*In digital photography, autofocus is a key feature that aids high-quality image capture, and modern approaches use the phase patterns arising from dual-pixel sensors as important focus cues. However, dual-pixel data is prone to multiple error sources in its image capturing process, including lens shading or distortions due to the inherent optical characteristics of the lens. We observe that, while these degradations are hard to model using prior knowledge, they are correlated with the spatial position of the pixels within the image sensor area, and we propose a learning-based autofocus model with positional encodings (PE) to capture these patterns. Specifically, we introduce RoI-PE, which encodes the spatial position of our focusing region-of-interest (RoI) on the imaging plane. Learning with RoI-PE allows the model to be more robust to spatially-correlated degradations. In addition, we also propose to encode the current focal position of lens as lens-PE, which allows us to significantly reduce the computational complexity of the autofocus model. Experimental results clearly demonstrate the effectiveness of using the proposed position encodings for automatic focusing based on dual-pixel data.*

## 1. Introduction

Automatically focusing to the region of interest is a fundamental problem in digital photography. When an object of interest is not in focus, the captured image will contain defocus blur, which significantly degrades perceptual image quality. Furthermore, an autofocus module usually appears early in the image signal processing (ISP) pipeline of a camera, making this type of blur propagate errors that are difficult to remove.

Existing works on autofocus (AF) mainly fall into two categories: contrast detection autofocus (CDAF) or phase detection autofocus (PDAF). CDAF approach first defines a sharpness metric of an image region of interest (RoI) and then moves the lens back and forth to achieve maximum sharpness. Since CDAF methods require a large number of observations and physical lens movements while searching, they are notoriously slow and power-consuming. Such limitations in contrast-based AF algorithms led to an innovative image sensor design, *i.e.* phase pixels (dual/quad-pixels), which provides an important cue for focus estimation.

PDAF approach determines that a region is in-focus by comparing the disparity between the left and right dual-pixel image. Since the amount of defocus is correlated with the disparity, many methods pre-calibrate the relationship between disparity and focus distance to make the prediction of the lens position in a single shot. However, these pre-calibrated predictions can be error-prone due to many physical / optical constraints, such as lens shading, geometrical distortions due to optical refraction of the camera lens, or extra variations when combined with optical zoom. Note that PDAF can be formulated as an extremely narrow (sub-pixel level) baseline stereo problem, and it is well known that depth estimation is error-prone with a narrow baseline [14, 26]. Moreover, recent smartphone cameras use smaller pixels compared to DSLRs, which makes them even more sensitive to signal noise and distortions. Individually modeling all error sources is impractical, and we propose a learning-based approach to tackle these problems.

In this paper, we present a novel framework for autofocus that better leverages the dual-pixel data. We focus on two aspects: 1) improving AF accuracy by handling spatially-varying distortions and 2) improving efficiency to be practically applicable even on low-end smartphones.

First, we speculate that the phase statistic of the dual-pixel data is different *w.r.t.* its spatial position relative to the full image sensor plane that receives light. Our motivation is illustrated in Fig. 1, where we observe that the same object located at the same depth can have different disparities *w.r.t.* the coordinates projected on the imaging plane. We also calculate the absolute disparity of our full training dataset by moving the lens from index 0∼49 for the patches that have the same ground truth depth (with focal index 10). Ideally, the calculated disparity values should be the same regardless of the relative position in the image sensor plane[1], but

[1]The ideal setting assumes a thin-lens approximation. To be precise, the disparity values of different position in the imaging plane *can* be different due to the field curvature of the thick (real-world) lens, meaning that the

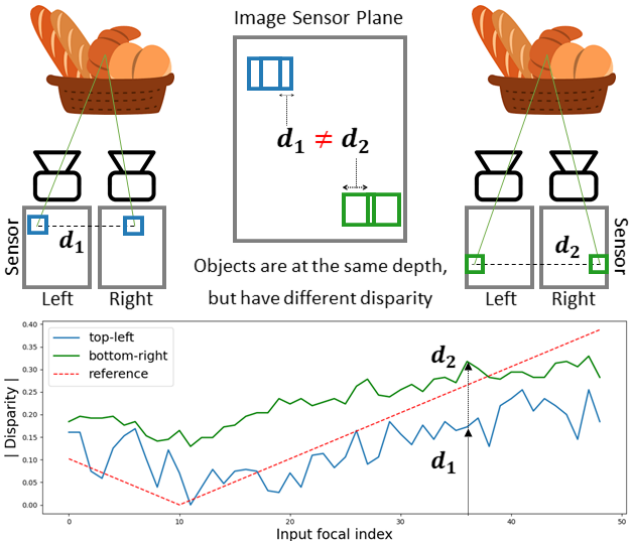


Figure 1: Different characteristics of dual-pixel data depending on the spatial position of each region of interest. We plot the calculated disparities with the same GT depth (focal index 10) *w.r.t.* each input lens position and also provide the reference disparity values in an ideal case. Note that the two regions on the imaging plane with the same depth should have the same disparities, but the values vary, resulting in different depth predictions. This motivates us to develop a new method that is robust to spatially-correlated distortions that hinders consistent disparity calculation.

we can observe clear variations from the plot in our feasibility test of Fig. 1. Therefore, we propose to learn these spatially-varying statistics by encoding the position of our region-of-interest (RoI), which we name *RoI-PE*. Training with our RoI-PE allows the model to effectively learn to be more robust to distortions and significantly boosts the AF accuracy. Note that, while our main target is to improve PDAF performance by exploring dual-pixel data, the proposed RoI-PE can also benefit CDAF algorithms by handling global distortions *e.g.* lens shading (see Sec. 4.3).

Second, we introduce a simple and compact representation to encode the current position of lens, named *lens-PE*. Using lens-PE allows us to substantially decrease the computational complexity of our AF model and also use smaller baseline architectures with minimal performance drop. For instance, we show in Sec. 4.1 that lens-PE can make our MobileNet-v2 [32] based AF model an order of magnitude more efficient (in terms of the # of parameters or FLOPs). Furthermore, our even smaller MCUNet [21] based model can still outperform the state-of-the-art [12] with less than 4MB SRAM (with float32 precision).

Our contributions can be summarized as follows:

- We propose two positional encodings for camera autofocus, RoI-PE and lens-PE, that effectively capture the complex characteristics of dual-pixel data.
- The proposed model significantly improves the focusing accuracy and the computational complexity.
- We thoroughly analyze the effects of spatial position in the dual-pixel data for various problem settings.

## 2. Related works

Conventional method for camera autofocus relies on the sharpness (or contrast) of an image to determine an optimal in-focus lens position. This approach is called contrast detection autofocus (CDAF), and many novel techniques have been proposed for robustly calculating the image sharpness based on image statistics [6, 15, 49] or frequency-domain image representations [13, 18, 19, 44, 47]. For CDAF, multiple images along the focal axis are sequentially captured to construct a focus curve, and the peak position of the curve is considered as the correct in-focus position of the lens with maximum contrast. While showing good AF performance, CDAF methods inevitably suffer from high latency issues, since they require sequential lens movements and image captures at multiple positions.

Recently, the latency issues of CDAF algorithms are being tackled by newly-developed image sensors that provide phase information, and AF algorithms running on these sensors are called phase detection autofocus (PDAF) [34]. The most widely-used sensor type is dual-photodiode (2PD, also called dual-pixel sensors), of which a single photodiode (pixel) is split into two. PDAF algorithms then detect the sign and the amount of disparity between the two (left/right) split pixels, and the direction and the amount of lens movement can be determined with a single image capture [17]. However, most of the existing methods require manual calibration that maps disparities to lens movements on hardware, which are prone to errors from *e.g.* lens fabrication or low-light noise. In this work, we mainly target dual-pixel data, but note that our proposed techniques can be easily extended to other hardware variations for phase detection (*e.g.* SuperPD [46], 2x2 On-Chip Lens [25], etc.).

Existing works on PDAF are fast, but their accuracy suffers from multiple sources of errors, which are difficult to model individually. Thus, numerous works try to improve the performance with learning-based solutions, including task-specific AF approaches for microscopy images [30, 48] and deep learning based methods with custom hardware settings for obtaining phase information [41, 3, 4]. However, existing works only utilize proprietary (dual-pixel) data, which make them difficult to compare.

regions located at the same GT metric *depth* can have different disparity values. However, note that there are also many other sources of spatially-varying distortions in the real world that can be compensated by learning with the proposed RoI-PE, so we generously use the term "same depth" in Fig. 1.

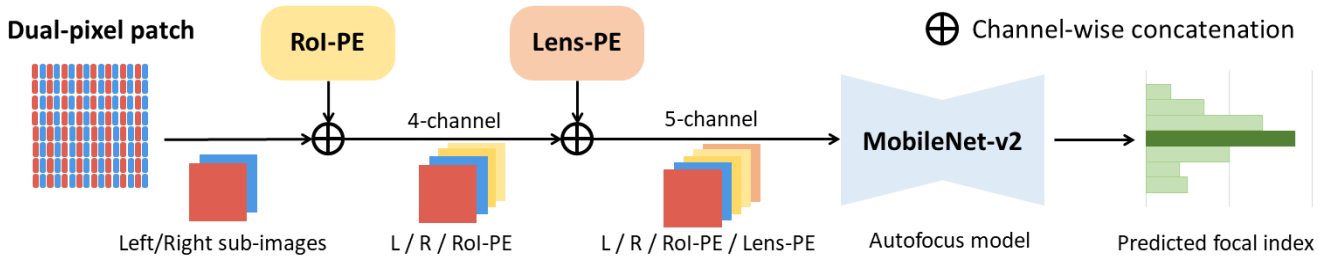


Figure 2: Overview of our AF framework with the proposed position encodings: RoI-PE and Lens-PE.

In the computer vision community, modern dual-pixel sensors are gaining increased attention, being applied to single-image depth estimation [27, 31], defocus deblurring [45], etc. For the task of autofocus, Herrmann *et al*. [12] is the first to propose a large-scale public dataset for training AF models on various settings, including the dual-pixel RAW input images. While demonstrating the state-of-the-art performance, [12] uses relatively heavier baseline model compared with the previous approaches. In this work, we use [12] as our baseline and introduce two position encodings, RoI-PE and lens-PE, which enables us to significantly boost the accuracy and the efficiency of our AF models.

**Depth from focus.** While there has been little attempts for AF in the computer vision community, it is worth mentioning the progress in monocular depth estimation with focus cues. The basic principle of depth from focus (DfF) methodology is to exploit the focus physics that the object within the focus range will appear clear, but the object gets blurry when outside the focus range due to the circle-of-confusion (CoC) of the camera. Traditional DfF pipelines start from building a focal sweep of multiple images by gradually moving the camera lens, and then try to estimate an accurate depth map of a scene [24, 29]. While building the full focal stack typically requires calibrated camera capturing a static scene, Suwajanakorn *et al*. [37] enabled uncalibrated DfF for dynamic scenes by accounting for parallax. In particular, [37] proposed a two-step procedure of 1) focal stack alignment and 2) depth reconstruction, which is also adopted and improved by recent deep learning based DfF methods [11, 43]. On the other hand, many previous works have tried to estimate depth from a single image [2, 35, 38], which is a significantly more difficult problem compared to using the focal stack. Such complication can be mitigated by using dual-pixel cues [8, 40] instead of focus / defocus blurs of a conventional image.

Unlike DfF, the autofocus problem requires predicting the exact lens position to guide the focus module, of which information is difficult to gather from non-metric depth maps [7]. Also, the main target setting of this work is to use a single dual-pixel image of a (given) random lens position as input instead of the focal stack; therefore, multi-image alignment issue due to rapid motion is a problem that is orthogonal to our contributions.

**Spatial position encoding.** Recent self-attention based architectures [39] have gained a lot of interest for its excellent performance in handling sequential data. One of the key innovation behind this is positional encoding, which enabled easy parallel training of sequences by encoding the absolute / relative position information of a certain token as an additional input. Recently, there have been many attempts to extend the positional encoding to 2-dimensional data (*i.e*. images) [20, 22, 42]. Notably, in CoordConv [22] or location-augmentation [42], both concatenate the 2-d coordinates channel-wise to the input tensor, and this way of providing hard-coded spatial information coincides with our proposed RoI-PE. However, while the representations may look the same, the motivation of RoI-PE originates from the need to capture and account for the spatially-correlated (geometric) distortions of the image sensors and the camera lens for the task of autofocus. Also, our RoI-PE is a cropped area that is either automatically detected or chosen by the user, so the physical meanings behind RoI-PE is completely different from CoordConv, which simply adds additional 2-d coordinate channels to learn spatial transformations involving position, orientation, or shape changes.

## 3. Methods

In this work, we generally follow the autofocus problem formulation as defined in Herrmann *et al*. [12]. Given an input image $I$, we extract a patch $I_k^p$, where $p \in \{1, ..., P\}$ denotes the spatial position of the region of interest within the input image and $k \in \{1, ..., n\}$ denotes the lens position discretized into $n$ focus distances. We mainly focus on dual-pixel data, so the input image $I$ becomes a two-channel raw image and $I_k^p \in \mathbb{R}^{2\times h\times w}$ is a cropped patch of $k$-th lens position with resolution $h \times w$. Following [12], we refer to the set of patches obtained by the full lens sweep $\{I_k^p\}$ as a *focal stack*, a single patch $I_k^p$ as a *focal slice*, and $k$ as the *focal index*. One difference with [12] is that we keep the spatial position index $p$, which we use as an additional input to the AF model (see Sec. 3.2).

Depending on the input space, an AF problem can be categorized into 3 settings [12]:

- Full stack: $\{I_k^p \mid k = 1, \ldots, n\} \mapsto k^*$
- Single slice: $I_k^p \mapsto k^*,\ \forall k \in \{1, \ldots, n\}$
- Multi-step: $\left(I_{k_0}^p \mapsto k_1\right),\ \left(I_{k_0}^p, I_{k_1}^p \mapsto k_2\right),\ \cdots \mapsto k^*$

Full-stack setting uses the full focal stack as input to an AF model, and CDAF approaches fall into this category. Single-slice setting uses a single focal slice at a random starting lens position $k$ and predicts the focus distance $k^*$ with one-shot, and many existing PDAF methods follow this formulation. Multi-step setting also receives a single focal slice at the beginning, but the number of observed focal slice can be stacked as we increase the number of prediction steps. In this work, we concentrate on more practical settings of single-slice and multi-step, which can better leverage the power of dual-pixel sensors.

### 3.1. Overall framework

In this framework, we propose two position encodings that can be attached to any baseline AF model as additional input channels to further improve the AF performance in terms of both accuracy and efficiency. First is the current (2-dimensional) spatial coordinates of the region-of-interest (RoI) patch, which captures the subtle spatial varieties in the phase difference of dual-pixel data with respect to its 2-d position. Second is the current position of the lens, which explicitly defines the focus distance of the current input dual-pixel image. We name these two encodings as *RoI position encoding (RoI-PE)* and *lens position encoding (lens-PE)*, respectively.

While the original model in [12] uses 98 input channels to account for 49 focal indices of the left/right dual-pixel images, we greatly reduce the input size by utilizing the proposed PEs to only have 5 input channels: 2 for the dual-pixel input, 2 for the RoI-PE, and the remaining 1 channel for the lens-PE. We describe the details in Secs. 3.2 and 3.3. Our overall framework is illustrated in Fig. 2.

### 3.2. RoI position encoding

The main hypothesis of our work is that the relationship between the disparity and the focus distance is different for each (spatial) position of the RoI. When an image sensor receives light, there exists multiple sources of degradation including lens distortion, intensity shading, etc. The statistical pattern of these degradation will be different for each pixel position in the image sensor plane; the center of an image will be the brightest, and the pixels near the boundary will be darker, because certain portion of light will be blocked by the aperture. Also, even if we were able to normalize the pixel brightness of all position, lens shading correction will also boost the low-light noise of the darker region, which will again make the high-frequency patterns different *w.r.t.* the pixel position. Such a difficult scenario motivates us to model this spatial pattern using neural networks.

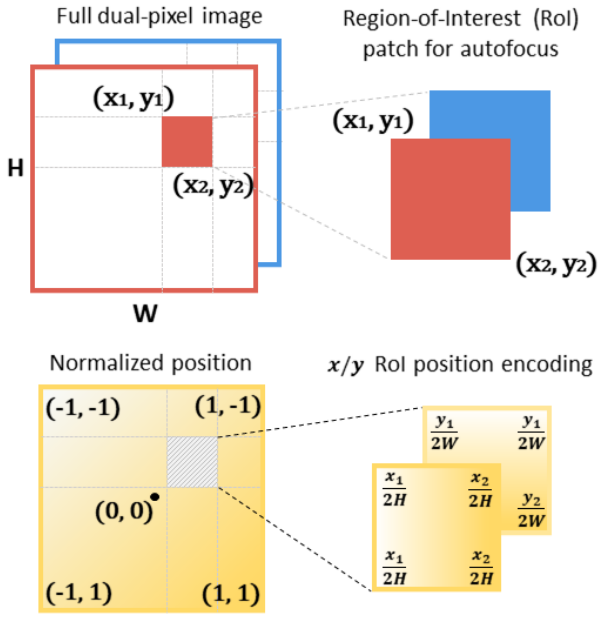


Figure 3: **RoI-PE**. We normalize the coordinates *w.r.t.* the image width/height, and crop the regions corresponding to the RoI patch of the dual-pixel image.

Let us assume that the input data $I_k^p$ has a spatial resolution of $h \times w$, which is cropped at the location $p$ from a much larger image. We can uniquely represent the position $p$ by the top-left coordinate $(x_1, y_1)$ and the bottom-right coordinate $(x_2, y_2)$, where $x_2 = x_1 + w$ and $y_2 = y_1 + h$. Assuming that the resolution of the full image is $H \times W$, we can transform the absolute coordinate values $(x, y)$ to be relative to the image resolution, $\left(\frac{x}{2W}, \frac{y}{2H}\right)$, where we set the center of the image as the origin and normalize the coordinates from -1 to 1. As illustrated in Fig. 3, we can specify the relative $x$ and $y$ coordinates of pixels in our RoI with two separate channels. Though it is possible to use more complex and physics-inspired positional encoding, we found that this simple setting of normalized linear coordinates works well in practice and demonstrates its effects in Secs. 4.1 and 4.3.

### 3.3. Lens position encoding

The single slice model, $I_k^p \mapsto k^*,\ \forall k \in \{1, \ldots, n\}$, takes the form of an $n$-class classification model. Herrmann *et al.* [12] uses $2 \times n$ input channels for the dual-pixel input data, where only 2 channels out of the $2n$ input channels are alive and the rest is masked as zeros ($n = 49$ in practice). We find this quite inefficient, since only 2 input channels out of 98 are used for each sample, and the rest 96 channels are only existent to implicitly represent the current

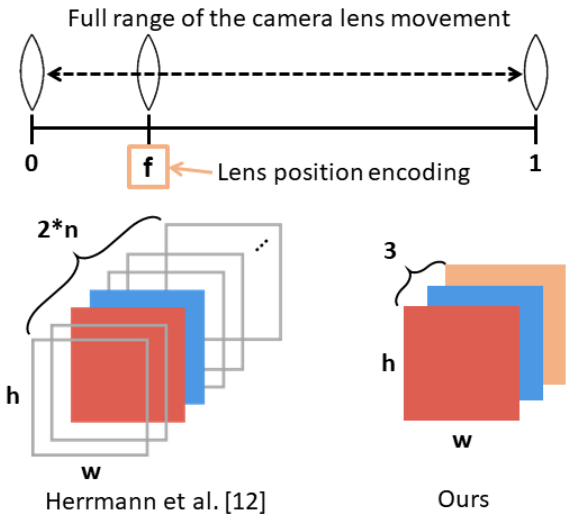


Figure 4: **Lens-PE**. We normalize the lens position from 0 to 1 and concatenate the (broadcasted) scalar value, which allows us to remove the zero-masked channels in [12].

position of the lens.

In this work, we propose to just use the 2-channel input data and encode the current lens position as a single additional channel to build a 3-channel lightweight input (total 5-channel if we include the RoI-PE). The modified input configuration is illustrated in Fig. 4. Note that we normalize the full range of the camera lens movement and define the lens position encoding (lens PE) as $f \in [0, 1]$, which can be represented as a single scalar value. In practice, we simply use $f = \frac{k}{n}$, $k \in \{1, \ldots, n\}$ to account for the discrete focal indices in [12] and for the ease of comparison. The scalar value of $f$ is then broadcasted as the spatial resolution of the input patch and concatenated channel-wise.

As a side effect of greatly reducing the number of input channels ($98 \rightarrow 3$), we could also reduce the width multiplier of MobileNet-v2 from 4.0 (in [12]) to 1.0 without any accuracy drop. The baseline model of [12] required wider channels to "prevent contraction in the number of channels between the input and the first layer" of MobileNet-v2, but our lightweight input with *lens-PE* safely alleviated the need for using a heavier model. To this end, we could even reduce the complexity of our baseline model to MCUNet-v2, which is more than 3 times lighter than MobileNet-v2, without compensating much accuracy (see Sec. 4.3).

### 3.4. Training

For end-to-end training of our model, we use the ordinal regression loss [5]. We use the $\ell_2$ cost metric as in [12], but additionally incorporate a *temperature* $T$ to control the sharpness of the soft ordinal labels. Hence, given the ground truth focal index $f$ of rank $r_f$, our target probability distribution at $k \in \{1, ..., n\}$ can be calculated as:

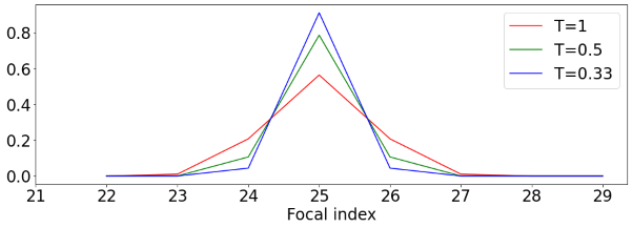


Figure 5: Target distribution *w.r.t.* different temperature values. We show an example for the GT focal index of 25.

$$y_k = \frac{e^{-(r_f - r_k)^2/T}}{\sum_{i=1}^{n} e^{-(r_f - r_i)^2/T}}, \tag{1}$$

where $r_k$ is the rank at focal index $k$, and $T$ is the temperature. Note that when $T \rightarrow 0$, this resorts into a Kronecker delta function as in a normal classification problem, and the loss used in [12] is when $T = 1$.

In practice, we keep $T = 1$ for the single-slice problem setting and decrease the temperature values for training the multi-step models for steps $\geq 2$. The intuition behind this choice is that as temperature gets lower, the target distribution for calculating the cross-entropy loss becomes sharper (see Fig. 5). Since the first step model already moved the lens position to somewhere near the correct focal index, we make the problem more difficult by sharpening the loss, thereby forcing the model to fine-tune its predictions to be close to the ground truth.

## 4. Experiments

For fair comparison, we borrow the experimental settings of [12], but mainly focus on the dual-pixel baselines. D* denotes a full-stack problem setting where all 49 dual-pixel images (hence, 98 channels) are used for input to the AF model. D1 denotes a single-slice setting, where a single dual-pixel image is used as an input. For our model in D1 setting, we include the proposed RoI-PE and lens-PE to build a 5-channel input data, whereas [12] still uses a 98-channel input with just a single dual-pixel image (2 channels) activated.

**Baseline models.** We use two types of network architecture for training our AF model: MobileNet-v2 [32] and MCUNet-v2 [21]. Let us omit the (-v2) suffix for both models to reduce clutter in this section. In particular, we use `mcunet-320kb-1mb_imagenet` model for the MCUNet baseline, which fits in 320KB SRAM and 1MB flash memory for a micro-controller unit when quantized to 8-bit (here, we just use the full float32 precision). We mainly compare with [12], which uses 4.0 MobileNet (channel widths multiplied by 4.0). We also use 4.0 MobileNet for D* setting, but reduce the width to 1.0 for D1 .

**Dataset.** We use the AF dataset proposed by Herrmann *et al*. [12], which includes 51 static scenes, 10 compositions for each scene, and 49 focal depths for all instances captured with Google Pixel 3 smartphones. The 49 focal depths are uniformly sampled from the inverse depth space ranging from 0.102 meters to 3.91 meters to compose a single *focal stack*, and the ground truth focal indices are obtained from a multi-view stereo pipeline [9]. The resolution of the input patch is $128 \times 128$, and there are 387,000 patches (460 focal stacks) in the train set and 56,800 patches (50 focal stacks) in the test set. For additional details on the dataset, we refer the readers to [12].

**Evaluation metric.** We evaluate our model by calculating the proportion of patches that are within 0, 1, 2, or 4 indices away from the ground truth focal index. This metric resembles the scheme used for Middlebury stereo [33], which is adapted by [12]. Qualitatively, we can argue that a patch with no more than 4-index difference lies inside the depth-of-field (DoF) and thus, in-focus. We also report the mean absolute error (MAE) and root-mean-square error (RMSE). For single-slice models, we average the performance of all 49 starting positions of the lens to account for accuracy variations *w.r.t.* the initial position. This also applies when evaluating our multi-step models.

**Implementation details.** We use PyTorch [28] for our implementation. For training, we use the Adam [16] optimizer with batch size 128, initial learning rate 0.001, $(\beta_1, \beta_2) = (0.5, 0.999)$, and cosine learning rate decay [23] for 60k iterations. When training the second-step model for the multi-step setting, we initialize it with the pretrained first-step model, set the temperature $T = 0.5$, and fine-tune the model with the learning rate $10^{-4}$ for 20k iterations.

### 4.1. Quantitative results

**AF accuracy (effects of RoI-PE).** We summarize the results of our AF model and the baselines in Tab. 1. For comparison, we take the methods with high performance in Herrmann *et al*. [12] and a recent survey [29]. We can observe that "Learning to Autofocus [12]", which we use as the baseline in building our model, greatly improved the performance of all other classical methods. Our model further enhanced the AF accuracy by a significant margin, which is mainly due to the proposed RoI-PE. In particular, the D1 setting of our models gave substantial gains of 7.6% classification accuracy for the exact ($= 0$) predictions and 3.7% for $\leq 4$ focal index difference for MobileNet baseline. Also, note that our MCUNet-based model already outperforms [12] consistently, even though its computational complexity is remarkably lower (see Tab. 2).

Though the advantage of using RoI-PE is more notable in the D1 setting, it is also beneficial for the full-stack setting of D*, with an overall 1.7% increase in the patches within the DoF ($\leq$ 4 index). Given that 95.5% of the patches were already in focus using [12], we believe that a further boost to 97.2% is quite meaningful and proves the effectiveness of our proposed RoI-PE.

| | Algorithm | higher is better $= 0$ | $\leq 1$ | $\leq 2$ | $\leq 4$ | lower is better MAE | RMSE |
|---|---|---|---|---|---|---|---|
| D* | Normalized SAD [10] | 0.166 | 0.443 | 0.636 | 0.819 | 4.280 | 8.981 |
| D* | Ternary Census (L1, $\epsilon = 30$) [36] | 0.171 | 0.450 | 0.633 | 0.802 | 4.347 | 8.794 |
| D* | Rank Transform (L1) [50] | 0.172 | 0.451 | 0.633 | 0.811 | 4.138 | 8.558 |
| D* | Census Transform (Hamming) [50] | 0.179 | 0.473 | 0.663 | 0.842 | 3.737 | 8.126 |
| D* | Ternary Census (L1, $\epsilon = 10$) [36] | 0.178 | 0.472 | 0.664 | 0.841 | 3.645 | 7.804 |
| D* | Normalized Envelope (L2) [1] | 0.155 | 0.432 | 0.633 | 0.856 | 2.945 | 5.665 |
| D* | Normalized Envelope (L1) [1] | 0.165 | 0.448 | 0.653 | 0.870 | 2.731 | 5.218 |
| D* | Learning to Autofocus [12] | 0.241 | 0.606 | 0.807 | 0.955 | 1.611 | 2.674 |
| D* | **Ours (MobileNet-v2)** | **0.273** | **0.675** | **0.851** | **0.972** | **1.356** | **2.128** |
| D1 | ZNCC Disparity with Calibration | 0.064 | 0.181 | 0.286 | 0.448 | 8.879 | 12.911 |
| D1 | SSD Disparity [40] | 0.097 | 0.262 | 0.393 | 0.547 | 7.537 | 11.374 |
| D1 | Learned Depth [9] | 0.108 | 0.289 | 0.428 | 0.586 | 7.176 | 11.351 |
| D1 | Learning to Autofocus [12] | 0.164 | 0.455 | 0.653 | 0.885 | 2.235 | 3.112 |
| D1 | **Ours (MCUNet-v2)** | **0.210** | **0.534** | **0.727** | **0.908** | **1.979** | **3.005** |
| D1 | **Ours (MobileNet-v2)** | **0.240** | **0.587** | **0.766** | **0.922** | **1.803** | **2.826** |

Table 1: Results of our AF model and baselines that use the dual-pixel data as inputs. D* indicates the algorithms that use the full focal stack of dual-pixel data. D1 methods receive a single dual-pixel focal slice, where the initial position is randomly chosen out of 49 focal indices. Our models outperform all compared methods for both D* and D1 .

**Computational complexity analysis (effects of Lens-PE).** Table 2 demonstrates the efficiency improvement for our final model, which is mainly achieved by reducing the # of input channels with our lens position encoding and the corresponding channel reduction of the main AF model (4.0 $\rightarrow$ 1.0 MobileNet $\rightarrow$ MCUNet). Compared with the baseline [12], our models require significantly less compute resource, which makes it especially more suitable for low-power mobile phones. Specifically, our MobileNet-based model requires $\times$15 less # of parameters than [12], $\times$18 less FLOPs, and $\times$13 less GPU memory. We could further push the resource constraints with our MCUNet-based model, which uses $\times$59 less parameters, $\times$41 less FLOPs, and $\times$45 less memory when compared with [12]. For the actual running time of the models, however, we could not observe as significant improvements. This is because all models (including 4.0 MobileNet) are relatively lightweight for our current environment using an NVIDIA V100 GPU, and almost all operations could be run in parallel at this scale of channel widths. We believe that the runtime gap will become much more evident on commodity hardwares in our cameras.

In Tab. 3, we analyze the importance of lens-PE in AF accuracy. Starting from our re-implementation of *Learning to AF* [12] for the single-slice setting (D1 ), simply discarding all zero-masked channels and using the input dual-pixel image results in an extremely degraded accuracy (Ours w/o

| Algorithm | # of params | FLOPs | GPU memory | Runtime |
|---|---|---|---|---|
| Learning to AF [12] | 34.75M | 1907M | 170.9MB | 6.42ms |
| **Ours (MobileNet-v2)** | 2.30M | 105M | 12.8MB | 6.22ms |
| **Ours (MCUNet-v2)** | 0.59M | 46M | 3.8MB | 6.07ms |

Table 2: Computational complexity comparison. Our models are significantly more efficient than [12], with more than an order of magnitude less parameters, FLOPs, and peak GPU memory.

| Algorithm | # of input channels | higher is better | | | | lower is better | |
|---|---|---|---|---|---|---|---|
| | | $= 0$ | $\leq 1$ | $\leq 2$ | $\leq 4$ | MAE | RMSE |
| Learning to AF [12] | 98 | 0.164 | 0.455 | 0.653 | **0.885** | **2.235** | **3.112** |
| Learning to AF † | 98 | **0.183** | **0.470** | **0.659** | 0.876 | 2.324 | 3.577 |
| **Ours (w/o lens-PE)** | 2 | 0.068 | 0.178 | 0.244 | 0.405 | 7.429 | 9.788 |
| **Ours (w. lens-PE)** | 3 | **0.182** | **0.460** | **0.652** | **0.871** | **2.367** | **3.549** |

Table 3: Effects of lens PE in autofocus accuracy. Since there is no open-source code available for [12], we reproduced the model and show the results (marked with †). We can observe that the proposed lens PE is crucial in maintaining the original AF accuracy.

| Algorithm | # of steps | higher is better | | | | lower is better | |
|---|---|---|---|---|---|---|---|
| | | $= 0$ | $\leq 1$ | $\leq 2$ | $\leq 4$ | MAE | RMSE |
| ZNCC Disparity with Calibration | 1 | 0.064 | 0.181 | 0.286 | 0.448 | 8.879 | 12.911 |
| | 2 | 0.100 | 0.278 | 0.426 | 0.617 | 6.662 | 10.993 |
| Learned Depth [9] | 1 | 0.108 | 0.289 | 0.428 | 0.586 | 7.176 | 11.351 |
| | 2 | 0.172 | 0.433 | 0.618 | 0.802 | 3.876 | 7.410 |
| Learning to AF [12] | 1 | 0.164 | 0.455 | 0.653 | 0.885 | 2.235 | 3.112 |
| | 2 | 0.201 | 0.519 | 0.723 | 0.916 | 1.931 | 2.772 |
| **Ours (MobileNet-v2)** | 1 | 0.240 | 0.587 | 0.766 | 0.922 | 1.803 | 2.826 |
| | 2 | **0.250** | **0.604** | **0.790** | **0.940** | **1.668** | **2.639** |

Table 4: Results for multi-step D1 models. Our single-step model already outperforms all compared multi-step methods, and our 2-step further enhances the AF accuracy.

lens-PE). However, if we concatenate a single additional input channel of lens-PE, we can safely remove the input channels with almost no accuracy drop. From this result, we can interpret that the current position of the lens is a crucial information for finding the correct focus distance, and that it is difficult for our models (MobileNet) to directly learn the relationship between the phase difference of the dual-pixel sub-images and the depth[2]. Note that the inputs of [12] already includes the lens position implicitly with the activated positions of the mask, while our model explicitly provides the current position as lens-PE.

**Multi-step models.** The quantitative performance of the multi-step models is summarized in Tab. 4. Note that our single-step model already outperforms the 2-step model of [12], and our 2-step model notably increases the performance gap. In terms of computational efficiency, our 2-step model requires 4.60M parameters with 0.209 GFLOPs in total, while the single-step model of [12] already requires 34.75M parameters and 1.907 GFLOPs.

The second step of our 2-step model is trained with the ordinal loss function with temperature $T = 0.5$ in Eq. (1). We further discuss the effects of temperature in Sec. 4.3.

## 4.2. Qualitative results

We qualitatively demonstrate the effects of our proposed RoI-PE in Fig. 6, where we show three different patches with the same GT focal indices but at different spatial positions. Compared with Herrmann *et al.* [12], our model predictions are more consistent *w.r.t.* the different position, being able to precisely estimate the focal index of $\leq 1$ GT index. On the other hand, the predictions by [12] show $\geq 2$ index difference, and the patch nearer to the image center sticks out with $= 4$ index away from the GT. We claim that the more coherent predictions were possible because the spatially-varying characteristics of the input dual-pixel image is successfully captured by learning with our RoI-PE.

In Fig. 7, we show additional qualitative results. From the first and the second row, we can observe that both Herrmann *et al.* [12] and our AF model are able to successfully detect severe defocus blur and move the focus module close to the GT index of 22. However, *Ours* is more precise and show sharper edges (better visual quality). From the third row of Fig. 7, we observed that L2A [12] sometimes breaks drastically and predict completely out-of-focus indices, usually for cases with little texture in the target RoI. While such case is a generally difficult scenario for AF since there is very little disparity / blur cues, our model is able to estimate the (approximately) correct focal index with higher success rate. Given that our baseline network is much more lightweight compared to L2A (see Tab. 2), these results support the effectiveness of our proposed RoI-PE and lens-PE.

In Fig. 8, we also analyze the remaining limitations and the failure cases of our model. In general, the results follow the well-known difficulties of PDAF: (a), it is difficult to learn meaningful phase information in regions with almost no texture, (b) our model successfully predicted the direction of lens movement but sometimes passed the correct index and overshot, (c) there exists focal breathing effects in the dataset (which can be seen as labeling noise), and (d) we calculate the phase difference between the left-right images, so horizontal lines are hard to detect (actually, it is impossible if the line is perfectly parallel to the horizontal direction). While further research is required to resolve all of these issues, we discuss some possible directions below. For (a) and (c), one could use a larger input RoI, so that the AF model can better capture the image context. For (b),

[2]Theoretically, directly learning the depth is possible if we formulate our AF problem as a stereo matching task between the left/right dual-pixel images. However, such setting is impractical with an extremely narrow baseline of 1/2 pixel pitch.

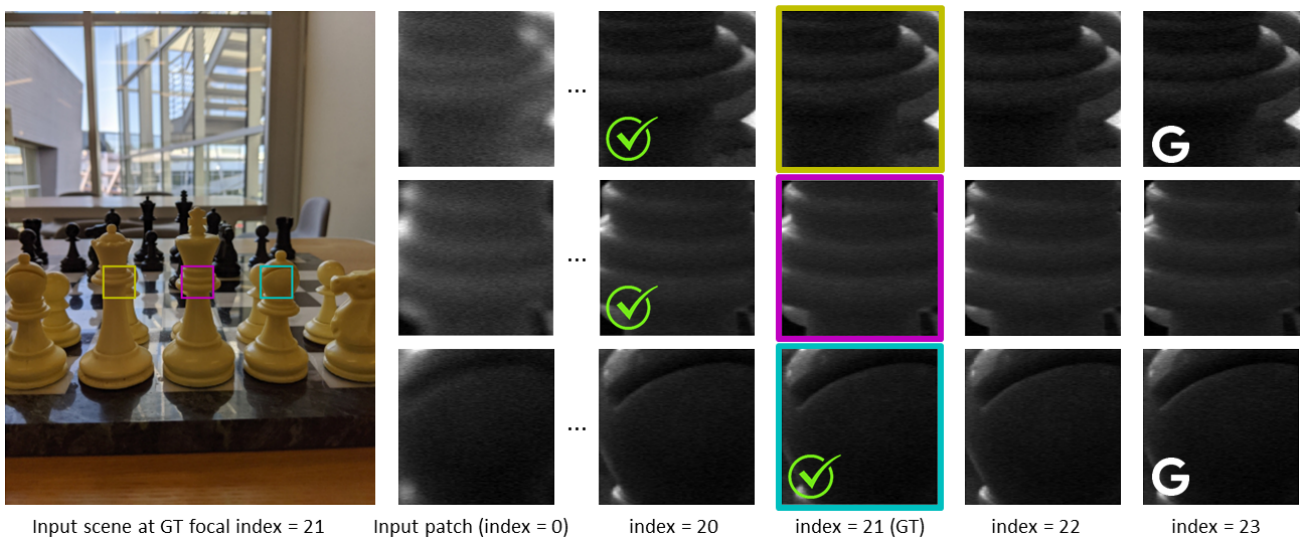


Figure 6: Qualitative result for the patches with the same depth at different positions. Given an input patch at focal index 0, all patches have the same GT index of 21. While our model (green ✓ mark) is able to accurately and consistently predict the focal index to $\leq 1$, the baseline model predictions [12] (white **G** mark) are not as close (for the second row, [12] estimates index = 25). Note that this setting resembles that of Fig. 1, where the estimated disparities become different even though the actual depth values are the same.

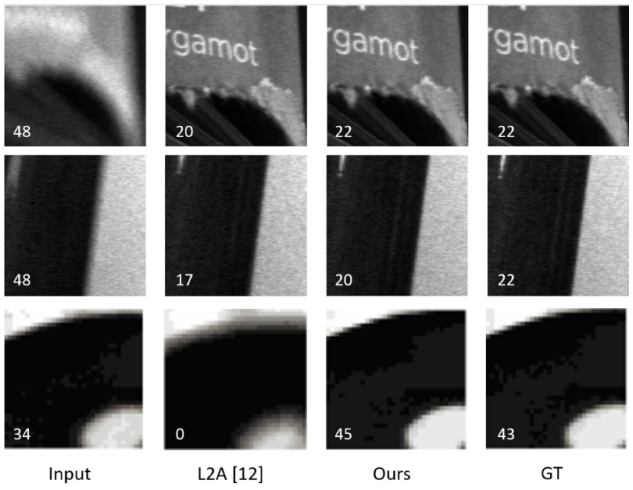


Figure 7: Additional qualitative comparison between L2A [12] and ours. The numbers in the bottom-left corner of each patch indicate the focal index, where 48 is the nearest (10.2cm focus distance) and 0 the furthest. Ours demonstrate more robust results with sharper focus predictions closer to the GT. We visualize the *left* image of the dual-pixel data. Best viewed zoomed in.

we think the problem is due to incomplete learning of the correlation between the dual-pixel disparity and the focal index; thus, better training recipe with larger data would be helpful. For (d), it is extremely difficult to solve using the current dual-pixel data, but improved sensors such as 2x2

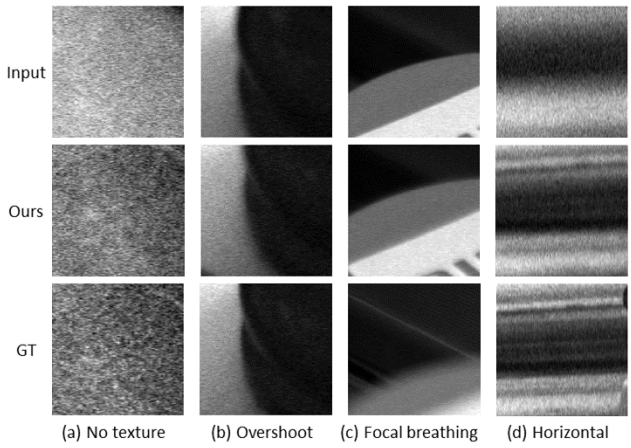


Figure 8: Failure cases of our AF model. We visualize the *left* image of the dual-pixel images.

OCL (on-chip lens) could easily mitigate this issue. For additional qualitative results and analyses, please refer to the supplementary material and our project page[3].

### 4.3. Analysis

**Effects of varying RoI-PE.** To better understand the effects of RoI-PE, we varied the coordinate system that represents the spatial position of our RoIs. In particular, we explore polar coordinates, following our hypothesis that the spatially correlated errors of conventional PDAF ap-

[3] https://myungsub.github.io/autofocus/

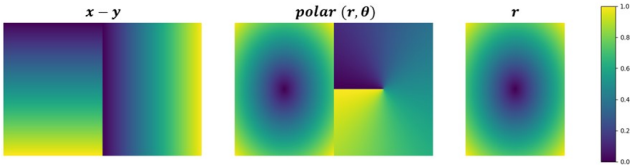


Figure 9: Visualization of the types of RoI-PE.

| RoI-PE type | higher is better | | | | lower is better | |
|---|---|---|---|---|---|---|
| | $= 0$ | $\leq 1$ | $\leq 2$ | $\leq 4$ | MAE | RMSE |
| $x - y$ | **0.240** | **0.587** | **0.766** | **0.922** | **1.803** | **2.826** |
| polar | 0.238 | 0.573 | 0.754 | 0.918 | 1.850 | 2.902 |
| $r$ | 0.211 | 0.522 | 0.714 | 0.900 | 2.079 | 3.237 |

Table 5: Varying the types of RoI-PE. A simple $(x, y)$ coordinate system worked best in our experiments.

| | Algorithm | higher is better | | | | lower is better | |
|---|---|---|---|---|---|---|---|
| | | $= 0$ | $\leq 1$ | $\leq 2$ | $\leq 4$ | MAE | RMSE |
| I* | Learning to AF [12] | 0.233 | 0.600 | 0.798 | 0.957 | 1.600 | 2.446 |
| I* | **Ours (MobileNet-v2)** | **0.267** | **0.663** | **0.848** | **0.970** | **1.395** | **2.220** |
| I1 | Learning to AF [12] | 0.115 | 0.318 | **0.597** | 0.691 | 4.321 | 6.737 |
| I1 | **Ours (MobileNet-v2)** | **0.131** | **0.339** | 0.490 | **0.710** | **4.147** | **6.546** |

Table 6: Results of RoI-PE on conventional raw image data. Even without phase information, RoI-PE is still beneficial to the AF performance, although the improvements are not as dramatic as D* or D1 settings.

proaches originate from lens distortions (*e.g.* pincushion / barrel). Denoting our original RoI-PE that uses a simple $(x, y)$ cartesian coordinate as $x - y$, *polar* is its polar coordinate counterpart, and $r$ stands for the first dimension of $(r, \theta)$ polar coordinate system (thus, RoI-PE for $r$ is single-channel, while the others use 2 channels). We visualize the type of RoI-PEs used in our experiments in Fig. 9, and the experimental results are summarized in Tab. 5.

**Effects of RoI encoding on conventional image data.** In Tab. 6, we demonstrate the results of training with RoI encoding on a conventional single-channel raw image (we take the average of the left and right images to get the single-channel image). Recall our motivation of RoI-PE, which was to capture the spatially-correlated distortions of the dual-pixel data. While there is no phase information available for conventional images, our RoI-PE can still be beneficial for learning to be more robust to spatial distortions such as lens shading. Consequently, we can observe small performance improvements for both the full-stack (I* ) and the single-slice (I1 ) settings. For I1 setting, note that our model is significantly more efficient since we use a 5-channel input including our RoI-PE and lens-PE with 1.0 MobileNet.

From these results, we can conclude that our proposed

| Algorithm | $T$ | higher is better | | | | lower is better | |
|---|---|---|---|---|---|---|---|
| | | $= 0$ | $\leq 1$ | $\leq 2$ | $\leq 4$ | MAE | RMSE |
| Single-step | 1.0 | 0.240 | 0.587 | 0.766 | 0.922 | 1.803 | 2.826 |
| Multi-step | 1.0 | **0.250** | 0.593 | 0.778 | 0.935 | 1.721 | 2.741 |
| | 0.5 | **0.250** | **0.604** | **0.790** | **0.940** | **1.668** | **2.639** |
| | 0.33 | **0.250** | **0.604** | **0.790** | **0.940** | 1.674 | 2.661 |

Table 7: Effect of temperature $T$ in ordinal regression loss.

RoI-PE is advantageous at all input settings, but it is especially more effective for dual-pixel inputs where the subtle phase differences *w.r.t.* the spatial position within the image sensor plane can be accounted for.

**Effects of temperature in ordinal regression loss.** Table 7 summarizes the effects of using different temperature $T$ in the ordinal regression loss in our multi-step AF framework. Regardless of the value of $T$, multi-step models all outperform the single-step baseline, but $T = 0.5$ shows slightly better accuracy compared with the others. Interestingly, we can observe that $T = 0.5$ and $T = 0.33$ settings give identical performance until $\leq 4$, but differ in MAE or RMSE. This is because the initial predictions that are close to the ground truth focal index are both well-tuned, but $T = 0.5$ can better refine the *wrong* ($> 4$ index difference) initial predictions to be closer to the correct index. In other words, $T = 0.33$ could not bring the nearby classes (indices) closer, since an ordinal loss with lower temperature becomes more similar to a standard cross entropy.

## 5. Conclusion

We proposed an improved autofocus model that can capture the subtle patterns of dual-pixel data depending on its spatial position. We introduced two position encodings, RoI-PE and lens-PE, that are channel-wise concatenated to the dual-pixel input image. By encoding the spatial position of the region-of-interest as RoI-PE, we could significantly improve the AF accuracy. In addition, by compactly encoding the current lens position with lens-PE, we could build an order-of-magnitude more efficient AF model compared with our baseline [12]. Experimental results clearly demonstrated the effectiveness of using the proposed position encodings for dual-pixel data, and we extensively analyzed the characteristics of our model on various autofocus problem settings.

## References

[1] Stan Birchfield and Carlo Tomasi. A pixel dissimilarity measure that is insensitive to image sampling. *IEEE TPAMI*, 1998. 6

[2] Marcela Carvalho, Bertrand Le Saux, Pauline Trouvé-Peloux, Andrés Almansa, and Frédéric Champagnat. Deep

depth from defocus: how can defocus blur improve 3d estimation using dense neural networks? In *ECCV Workshops*, 2018. 3

[3] Homer H. Chen Chi-Jui Ho, Chin-Cheng Chan. Af-net: A convolutional neural network approach to phase detection autofocus. *IEEE TIP*, 2020. 2

[4] Homer H. Chen Chin-Cheng Chan. Autofocus by deep reinforcement learning. *Electronic Imaging*, 2019. 2

[5] Raul Diaz and Amit Marathe. Soft labels for ordinal regression. In *CVPR*, 2019. 5

[6] Judith Dijk, Michael van Ginkel, Rutger J van Asselt, Lucas J van Vliet, and Piet W Verbeek. A new sharpness measure based on gaussian lines and edges. In *10th International Conference on Computer Analysis of Images and Patterns (CAIP)*, 2003. 2

[7] David Eigen, Christian Puhrsch, and Rob Fergus. Depth map prediction from a single image using a multi-scale deep network. In *NIPS*, 2014. 3

[8] Rahul Garg, Neal Wadhwa, Sameer Ansari, and Jonathan T Barron. Learning single camera depth estimation using dual-pixels. In *ICCV*, 2019. 3

[9] Rahul Garg, Neal Wadhwa, Sameer Ansari, and Jonathan T. Barron. Learning single camera depth estimation using dual-pixels. In *ICCV*, 2019. 6, 7

[10] Marsha Jo Hannah. *Computer Matching of Areas in Stereo Images*. PhD thesis, Stanford University, 1974. 6

[11] Caner Hazirbas, Sebastian Georg Soyer, Maximilian Christian Staab, Laura Leal-Taixé, and Daniel Cremers. Deep depth from focus. In *ACCV*. Springer, 2018. 3

[12] Charles Herrmann, Richard Strong Bowen, Neal Wadhwa, Rahul Garg, Qiurui He, Jonathan T. Barron, and Ramin Zabih. Learning to autofocus. In *CVPR*, 2020. 2, 3, 4, 5, 6, 7, 8, 9

[13] Jui-Ting Huang, Chun-Hung Shen, See-May Phoong, and Homer Chen. Robust measure of image focus in the wavelet domain. *IEEE International Symposium on Intelligent Signal Processing and Communication Systems*, 2005. 2

[14] Neel Joshi and C Lawrence Zitnick. Micro-baseline stereo. *Microsoft Research Technical Report*, 2014. 1

[15] Nasser Kehtarnavaz and H-J. Oh. Development and real-time implementation of a rule-based auto-focus algorithm. *Real-Time Imaging*, 2003. 2

[16] Diederick P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015. 6

[17] Masahiro Kobayashi, Michiko Johnson, Yoichi Wada, Hiromasa Tsuboi, Hideaki Takada, Kenji Togo, Takafumi Kishi, Hidekazu Takahashi, Takeshi Ichikawa, and Shunsuke Inoue. A low noise and high sensitivity image sensor with imaging and phase-difference detection af in all pixels. *ITE Transactions on Media Technology and Applications*, 4(2):123–128, 2016. 2

[18] Matej Kristan, Janez Pers, Matej Perse, and Stanislav Kovacic. A bayes-spectral-entropy-based measure of camera focus using a discrete cosine transform. *Pattern Recognition Letters*, 2006. 2

[19] Sang Yong Lee, Jae Tack Yoo, and Soo-Won Kim. Reduced energy-ratio measure for robust autofocusing in digital camera. *Signal Processing Letters*, 2009. 2

[20] Yang Li, Si Si, Gang Li, Cho-Jui Hsieh, and Samy Bengio. Learnable fourier features for multi-dimensional spatial positional encoding. In *NeurIPS*, 2021. 3

[21] Ji Lin, Wei-Ming Chen, Han Cai, Chuang Gan, and Song Han. Mcunetv2: Memory-efficient patch-based inference for tiny deep learning. *arXiv preprint arXiv:2110.15352*, 2021. 2, 5

[22] Rosanne Liu, Joel Lehman, Piero Molino, Felipe Petroski Such, Eric Frank, Alex Sergeev, and Jason Yosinski. An intriguing failing of convolutional neural networks and the coordconv solution. In *NeurIPS*, 2018. 3

[23] Ilya Loshchilov and Frank Hutter. Sgdr: Stochastic gradient descent with warm restarts. In *ICLR*, 2017. 6

[24] Michael Moeller, Martin Benning, Carola Schonlieb, and Daniel Cremers. Variational depth from focus reconstruction. *IEEE TIP*, 2015. 3

[25] Tetuya Okawa, S Ooki, H Yamajo, M Kawada, M Tachi, K Goi, T Yamasaki, H Iwashita, M Nakamizo, T Ogasahara, et al. A 1/2inch 48m all pdaf cmos image sensor using 0.8 $\mu$m quad bayer coding 2$\times$ 2ocl with 1.0 lux minimum af illuminance level. In *International Electron Devices Meeting (IEDM)*, 2019. 2

[26] Masatoshi Okutomi and Takeo Kanade. A multiple-baseline stereo. *IEEE TPAMI*, 15(4):353–363, 1993. 1

[27] Liyuan Pan, Shah Chowdhury, Richard Hartley, Miaomiao Liu, Hongguang Zhang, and Hongdong Li. Dual pixel exploration: Simultaneous depth estimation and image restoration. In *CVPR*, 2021. 3

[28] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *NeurIPS*, 2019. 6

[29] Said Pertuz, Domenec Puig, and Miguel Garccia. Analysis of focus measure operators in shape-from-focus. *Pattern Recognition*, 2012. 3, 6

[30] Henry Pinkard, Zachary Phillips, Arman Babakhani, Daniel A. Fletcher, and Laura Waller. Deep learning for single-shot autofocus microscopy. *Optica*, 2019. 2

[31] A. Punnappurath, A. Abuolaim, M. Afifi, and M.S. Brown. Modeling defocus-disparity in dual-pixel sensors. In *International Conference on Computational Photography (ICCP)*, 2020. 3

[32] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. MobileNetV2: Inverted residuals and linear bottlenecks. In *CVPR*, 2018. 2, 5

[33] Daniel Scharstein and Richard Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *IJCV*, 2002. 6

[34] Przemyslaw Sliwinski and Pawel Wachel. A simple model for on-sensor phase-detection autofocusing algorithm. *Journal of Computational Chemistry*, 2013. 2

[35] Pratul P Srinivasan, Rahul Garg, Neal Wadhwa, Ren Ng, and Jonathan T Barron. Aperture supervision for monocular depth estimation. In *CVPR*, 2018. 3

[36] Fridtjof Stein. Efficient computation of optical flow using the census transform. *Pattern Recognition*, 2004. 6

[37] Supasorn Suwajanakorn, Carlos Hernandez, and Steven M. Seitz. Depth from focus with your mobile phone. *CVPR*, 2015. 3

[38] Huixuan Tang, Scott Cohen, Brian Price, Stephen Schiller, and Kiriakos N Kutulakos. Depth from defocus in the wild. In *CVPR*, 2017. 3

[39] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 3

[40] Neal Wadhwa, Rahul Garg, David E. Jacobs, Bryan E. Feldman, Nori Kanazawa, Robert Carroll, Yair Movshovitz-Attias, Jonathan T. Barron, Yael Pritch, and Marc Levoy. Synthetic depth-of-field with a single-camera mobile phone. *SIGGRAPH*, 2018. 3, 6

[41] Chengyu Wang, Qian Huang, Ming Cheng, Zhan Ma, and David J. Brady. Deep learning for camera autofocus. *IEEE Transactions on Computational Imaging*, 2021. 2

[42] Zhenyi Wang and Olga Veksler. Location augmentation for cnn. *arXiv preprint arXiv:1807.07044*, 2018. 3

[43] Changyeon Won and Hae-Gon Jeon. Learning depth from focus in the wild. In *ECCV*. Springer, 2022. 3

[44] Hui Xie, Weibin Rong, and Lining Sun. Wavelet-based focus measure and 3-d surface reconstruction method for microscopy images. In *IROS*, 2006. 2

[45] Shumian Xin, Neal Wadhwa, Tianfan Xue, Jonathan T. Barron, Pratul P. Srinivasan, Jiawen Chen, Ioannis Gkioulekas, and Rahul Gar. Defocus map estimation and deblurring from a single dual-pixel image. In *ICCV*, 2021. 3

[46] Takahiro Yamasaki, Tomohiro Nakamura, Ryohei Funatsu, and Hiroshi Shimamoto. Hybrid autofocus system by using a combination of the sensor-based phase-difference detection and focus-aid signal. In *International Conference on Consumer Electronics (ICCE)*, 2018. 2

[47] Ge Yang and Bradley J Nelson. Wavelet-based autofocusing and unsupervised segmentation of microscopic images. In *IROS*, 2003. 2

[48] Hyun Jong Yang, Moohyun Oh, Jonggyu Jang, Hyeonsu Lyu, and Junhee Lee. Robust deep-learning based autofocus score prediction for scanning electron microscope. *Microscopy and Microanalysis*, 2020. 2

[49] Yi Yao, Besma Abidi, Narjes Doggaz, and Mongi Abidi. Evaluation of sharpness measures and search algorithms for the auto-focusing of high magnification images. *Pattern Recognition Letters*, 2006. 2

[50] Ramin Zabih and John Woodfill. Non-parametric local transforms for computing visual correspondence. In *ECCV*, 1994. 6