DoCam: Depth Sensing with an Optical Image Stabilization Supported RGB Camera

Hao Pan
Shanghai Jiao Tong University
panh09@sjtu.edu.cn

Feitong Tan
Simon Fraser University
feitongt@sfu.ca

Yi-Chao Chen*
Shanghai Jiao Tong University
yichao@sjtu.edu.cn

Gaoang Huang
Shanghai Jiao Tong University
myalos@sjtu.edu.cn

Qingyang Li
Shanghai Jiao Tong University
disward2017@sjtu.edu.cn

Wenhao Li
Shanghai Jiao Tong University
fire1997ice@sjtu.edu.cn

Guangtao Xue*
Shanghai Jiao Tong University
gt_xue@sjtu.edu.cn

Lili Qiu
University of Texas at Austin
lili@cs.utexas.edu

Xiaoyu Ji
Zhejiang University
xji@zju.edu.cn

ABSTRACT

Optical image stabilizers (OIS) are widely used in digital cameras to counteract motion blur caused by camera shakes in capturing videos and photos. In this paper, we sought to expand the applicability of the lens-shift OIS technology for metric depth estimation, *i.e.*, let a RGB camera to achieve the similar function of a time-of-flight (ToF) camera. Instead of having to move the entire camera for depth estimation, we propose DoCam, which controls the lens motion in the OIS module to achieve 3D reconstruction. After controlling the lens motion by altering the MEMS gyroscopes readings through acoustic injection, we improve the traditional bundle adjustment algorithm by establishing additional constraints from the linearity of the lens control model for high-precision camera pose estimation. Then, we elaborate a dense depth reconstruction algorithm to compute depth maps at real-world scale from multiple captures with micro lens motion (*i.e.*, $\leq$ 3 *mm*). Extensive experiments demonstrate that our proposed DoCam can enable a 2D color camera to estimate high-accuracy depth information of the captured scene by means of controlling lens motion in the OIS. DoCam is suitable for a variety of applications that require depth information of the scenes, especially when only a single color camera is available and located at a fixed position.

CCS CONCEPTS

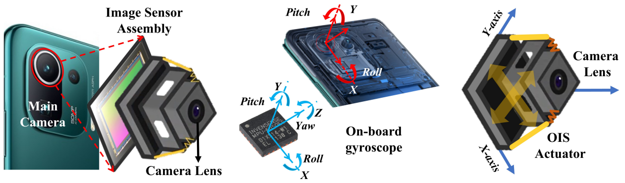
• **Information systems** → **Mobile information processing systems**; • **Computing methodologies** → **Computational photography**; • **Computer systems organization** → *Embedded systems*.

KEYWORDS

optical image stabilization; depth estimation; acoustic injection

ACM Reference Format:
Hao Pan, Feitong Tan, Yi-Chao Chen, Gaoang Huang, Qingyang Li, Wenhao Li, Guangtao Xue, Lili Qiu, and Xiaoyu Ji. 2022. DoCam: Depth Sensing with an Optical Image Stabilization Supported RGB Camera. In *The 28th Annual International Conference on Mobile Computing and Networking (ACM MobiCom '22), October 17–21, 2022, Sydney, NSW, Australia.* ACM, New York, NY, USA, 14 pages. https://doi.org/10.1145/3495243.3560523

1 INTRODUCTION



(a) An OIS-supported camera built in the smartphone

(b) MEMS sensors sense camera movements

(c) OIS actuator controls lens on the *X/Y-plane*

Figure 1: Working principle of lens-shift based OIS models

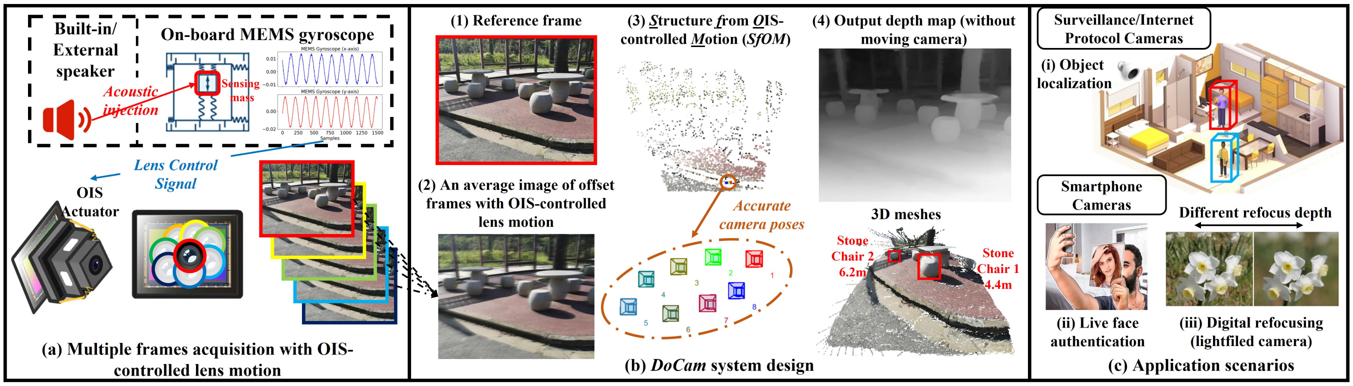
Motivation: Three-dimensional vision is particularly important and has become absolutely critical to enabling exciting applications, such as navigation and indoor localization [20], 3D display [9], scene understanding [54], and augmented reality [48]. Measuring distance relative to RGB cameras remains difficult, and now the best option for retrieving depth is to use active ranging sensors such as structured light sensors and 3D Time of Flight (ToF) sensors [15]. However, these active ranging sensors on the existing mobile devices still have some limitations. The depth of field for acquiring usable data is about one meter for a structured light system [37], which leads to its use only in specific application scenarios, such as face recognition. LiDAR sensors and infrared ToF sensors are sensitive to the ambient light and cannot work well in outdoors scenarios because high intensity sunlight can cause the sensor pixels to quickly saturate and fail to detect the actual light reflected off the object [56].

Stereo or multi-view based depth estimation methods [18, 30] provide the ability to estimate depth using RGB cameras, by means of using multiple cameras or moving a single camera to capture images from different views and combining camera projective geometry to achieve depth estimation. Unfortunately, these methods are limited in the scenarios where multiple cameras are not available and cameras cannot be moved (*e.g.*, fixed surveillance cameras). The deep learning-based approaches [29], which predict depth from a

[*] Guangtao Xue and Yi-Chao Chen are corresponding authors.

ACM MobiCom '22, October 17–21, 2022, Sydney, NSW, Australia

ACM ISBN 978-1-4503-9181-8/22/10. . . $15.00
https://doi.org/10.1145/3495243.3560523



(a) Multiple frames acquisition with OIS-controlled lens motion

(b) *DoCam* system design

(c) Application scenarios

Figure 2: (a) DoCam applies acoustic injection to alter MEMS gyroscope readings in order to adjust lens position in OIS-supported cameras. (b) Structure from OIS-controlled Motion (SfOM) algorithm is proposed to recover the accurate camera poses, and high quality dense depth map can be estimated. (c) DoCam can facilitate many applications

single RGB image, have an obvious drawback – it cannot determine the scale. Moreover, the performance of these supervised learning approaches is unstable due to their strong dependence on the distribution of the training dataset. In this paper, we focus on dense depth estimation from a single OIS-supported RGB camera. Even when the camera position is nearly static during capture, the depth map can also be reconstructed at real-world scale.

Our ideas: Optical image stabilizer (OIS) is a system commonly used in digital RGB cameras to reduce the effects of blur due to hand shake. Take a smartphone camera with OIS as an example, Fig. 1 presents an illustrative example of the lens-shift OIS architecture [26]: the camera movement is detected by the on-board gyroscope (Fig. 1(b)), and OIS actuator then moves the lens position with translational displacements to compensate the unwanted camera shake (Fig. 1(c)). Having the ability to control OIS means that we can program the lens to move precisely, which allows us to develop many applications such as generating super-resolution images [24], equatorial gauges [36], light filed camera [35], etc.

In this study, we develop a Depth sensing system with an OIS-supported Camera system, referred to as DoCam, which achieves the depth estimation in a single RGB camera without additional movement of the the entire camera. However, developing the proposed DoCam system impose a number of challenges.

Challenges: (1) Existing 2D color cameras that support OIS technologies on the market, such as surveillance cameras, smartphone cameras, DSLR cameras, etc., do not provide APIs to control OIS via programming. Therefore, we need to leverage the acoustic injection based method [46, 47] to alter the readings of the on-board MEMS gyroscope to control the OIS. However, existing methods based on phase modulation lead to frequent failures in our scenario due to the strict requirement over phase alignment between the acoustic signals and gyroscope readings. **(2)** The traditional multi-view geometry approaches assume that the lens and the image sensor move simultaneously. In our system, only the movement of the lens is controlled by the OIS, while the position of the image sensor is fixed. We need to adapt the existing multi-view geometry model to fit our system where only the lens moves. **(3)** For depth reconstruction, there is no prior art computing multi-view geometry using lens motion, it remains a challenge to incorporate OIS-controlled micro-scale lens motion (*i.e.*, $\leq$ 3 *mm*) into accurate camera poses for resolving high-quality depth estimation.

Solutions: The challenges above are addressed as follows: **(1)** Instead of using the phase modulation, we sought to realize the smooth and stable control over the gyroscope readings, by using the amplitude and frequency modulations to generate adequate and suitable acoustic injection signals. The built-in or external speaker is then used to drive the sensing mass (see Fig. 2(a)) by playing the modulated acoustic signals for the implementation of moving the lens in the OIS. **(2)** We propose a mathematical model to formulate the relationship between lens motion and camera motion, *i.e.*, camera pose parameters, as the latter is widely used in the multi-view geometry of computer vision. A two-stage camera calibration approach is applied to verify the modeled relationship between camera pose parameters (intrinsic and extrinsic) and OIS control signals. After the conversion, we can take advantage of precise control over OIS and apply the model to implement depth estimation. **(3)** We elaborate an accurate camera pose recovery scheme, called the structure from OIS-controlled motion (*SfOM*) algorithm, which leverages the linearity of the OIS control model to impose constraints to facilitate the estimation of camera pose associated with lens motion. The algorithm is proved to be effective in recovering the camera pose and geometrically reconstructing the extracted local points as depth information (see Fig. 2(b)).

Compared to existing works using visual geometry and algebraic methods which require hand motions (*i.e.*, 10 *cm*) to produce multi-view observations, controlling the OIS provides more precise information about the lens position and additional geometry constraints to yields better performance of dense depth map estimation. Also, DoCam can work together with the existing active depth ranging hardware (*e.g.*, Apple's LiDAR sensor). Although LiDAR can only obtain the depth information of sparse featured points in outdoor scenes due to the strong ambient light interference. Our proposed DoCam can utilize these sparse depth points for depth calibration and obtain almost the same photo-consistent depth map with higher estimation accuracy. Furthermore, our approach using OIS is complementary to deep learning-based methods and can be used to further improve these state-of-the-art solutions [27, 42, 52].

Applications: The DoCam we propose can flourish abundant applications. In scenarios with fixed cameras, such as surveillance and Internal Protocol (IP) cameras, DoCam can be used to reconstruct a depth map of the shooting scene and perform indoor/outdoor object localization (1st part of Fig. 2(c)). In scenarios with mobile cameras, such as smartphone cameras, our system can realize liveness detection during face authentication (2nd part of Fig. 2(c), note that the image is from [34]). With the precise photo-consistent depth map output by our system, digital refocusing can also be achieved, *i.e.*, changing a point or a plane of focus after taking a photo (3rd part of Fig. 2(c)).

The main contributions of this work are as follows:

- We cleverly exploit the potential of the OIS techniques to facilitate depth estimation in RGB cameras without adding additional hardware. To the best of our knowledge, our proposed system is the first to use lens motion in the OIS module to achieve depth estimation.
- We are the first to propose the formulation which mathematically model the conversion between camera poses and lens motions, and further establish the relationship between camera poses and the signals used to control the OIS during the capture of multiple images. These constraints can be leveraged in depth estimation.
- We develop a unified framework by which to estimate accurate camera poses from image sequences with a micro-scale stereo baseline for use in high-quality depth estimation. Specifically, the bundle adjustment is reformulated by applying constraints on the multi-view geometry and the OIS controlling signal.
- We prototype the DoCam on an Android smartphone (Xiaomi 10 Ultra). Evaluation results demonstrate the accuracy of depth from our DoCam system, additional experiments also are conducted to verify the effectiveness of three application scenarios based on DoCam.

2 RELATED WORKS

2.1 Optical Image Stabilization

There are two main methods to implement the OIS system: lens shifting and sensor shifting [26]. In the lens shifting method, the image sensor is fixed to the bottom of the camera case and the lens undergoes translational movement. In the sensor shifting method, the lens is fixed and the CMOS sensor undergoes translational movement. Sensor shifting is a DSLR technology that is also more complex and expensive [8], and a very small number of smartphone cameras (*e.g.*, iPhone 12 Pro Max) use the sensor-shift compensation OIS scheme. Therefore, considering that the vast majority of surveillance cameras, Internet Protocol (IP) cameras [11], and smartphone cameras on the market use the lens shift compensation method, in this work, we focus exclusively on lens-shift OIS modules that move the lens horizontally and vertically.

2.2 Acoustic Injections on MEMS Sensors

Researchers have demonstrated that MEMS gyroscopes and accelerometers can be affected by acoustic signals [40, 46, 47], and several researchers have exploited this phenomenon to implement various types of attack. In [40], researchers demonstrated a denial of service (DoS) attack using resonant acoustic signals to facilitate the intentional crashing of drones. In [46], researchers proposed output biasing and output control attacks to compromise the integrity of MEMS accelerometer readings. In [47], researchers achieved implicit control over a variety of real-world systems via non-invasive attacks targeting embedded inertial sensors. In [32] and [2], researchers demonstrated the feasibility of using inertial sensors in smartphones to eavesdrop on speech signals. Note that in the works described above, a high-power loudspeaker (*e.g.*, $50Watt$) is required to generate acoustic injection signals of sufficient to enable attacks from a distance. In this study, we manipulate the position of the lens in the lens-shift-based OIS module by injecting an acoustic signal in order to alter the readings of the built-in MEMS gyroscope . We also test mainstream OIS-supported cameras and prove the opportunities that the built-in speaker is close enough to the on-board gyroscope to enable its use in driving the MEMS sensors.

2.3 Depth from a RGB Camera

Monocular depth estimation methods are proposed to predict the depth value given a single RGB image. Such techniques pose the problem as end-to-end supervised deep learning [17, 28, 29, 53] models and have seen significant progress. By learning priors about objects and their relative positions, these networks reach remarkably good performance in restricted evaluation scenarios such as indoor and driving scenes. However, the accuracy of monocular depth estimation on learned scenery data drops considerably for shots taken in a different landscape. Another problem of the single image based methods is that they lack absolute depth information at the world scale, and all their output depth information is relative.

Monocular stereo based solution has led to renewed interest that estimates depth information from a video sequence captured when a single camera is moved. Conventional structure from motion (SfM) [38] and multi-view stereo (MVS) approaches [39, 55] assume that a good 3D reconstruction can be obtained with algebraic methods, which in turn depend on adequate baselines (distance between the first and the last frames), but such images with wide baselines (*e.g.*, $20\ cm$) are generally inconvenient for users to capture, especially in the mobile photography scenarios [50]. Several researches [19, 25, 57] have focused on estimating camera trajectories from image sequences captured while the camera was moved slightly (*e.g.*, $5\ cm$) and intentionally by hand. Accuracy in estimating a camera pose depends heavily on the initialization methods used for bundle adjustment [41, 44]. Moreover, in scenes with limited texture, it is difficult to extract a sufficient number of local feature to perform bundle adjustment, such that the system falls into a local minimum or fails to converge [45]. Thus, the methods mentioned above are prone to failure resulting from inaccurate camera pose estimates derived using bundle adjustment algorithms, especially in the case of narrow baselines. Unlike previous studies, we employ the OIS module to enable accurate control over lens motion and propose a novel camera pose recovery scheme called *SfOM* that reformulates bundle adjustments by imposing constraints on the multi-view geometry and OIS-control model. To the best of our knowledge, this is the first study to demonstrate high-accuracy estimates of camera poses based on a stereo baseline caused by lens motion and reconstruct high-quality dense depth map.

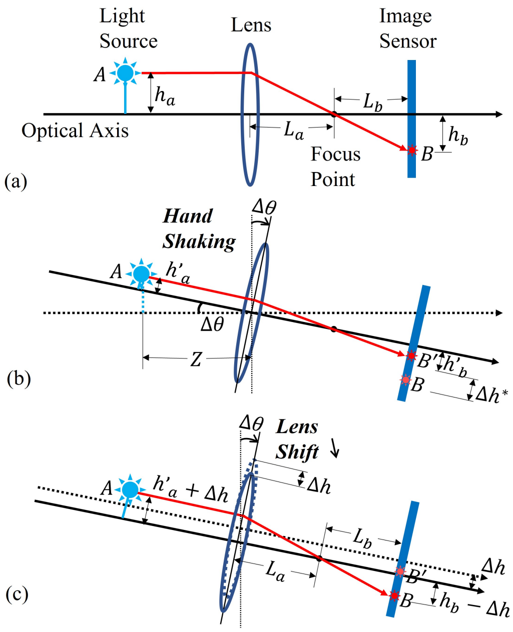


Figure 3: Relationship between offset angle of camera and lens compensatory motion in OIS

3 PRELIMINARY ANALYSIS OF LENS-SHIFT OPTICAL IMAGE STABILIZATION

We begin by formulating a lens-control model for the OIS based on the relationships between offset angles and lens compensation vectors. This initial work helps us to identify many of the challenges involved in the further development the DoCam system. The correctness of the lens-control model is then verified by comparing our camera pose results with those obtained using a two-stage camera calibration algorithm.

3.1 Lens Controlling Algorithm in the OIS

There is no doubt that the OIS module can compensate for camera translational displacement by moving the lens with the appropriate distance in opposite directions, which prompts us to control the accelerometer readings with acoustic injection and control the lens motion in OIS modules. However, the resonant frequencies of MEMS accelerometers (*i.e.*, $\leq 17KHz$) fall in the audible frequency band to the human ears [46], that makes such acoustic injection signals very disruptive to the users. Unlike accelerometers, the resonance frequencies of MEMS gyroscopes (18 ∼ 28 KHz) are in the inaudible frequency band [47], especially for most adults. This is also the main reason why acoustic-based works use signals in the $17 \sim 24\ KHz$ band for object tracking [33, 58] and imaging [59].

Therefore, in this paper, we decide to control the gyroscope readings with acoustic injection (one example of acoustic signals injected into gyroscopes is shown in Fig. 4), then further realize the lens motion control in OIS module. Before that, we must first model how the OIS actuator moves the lens in translation to correct the camera rotational displacement.

We use an example on the $x-$axis to prove that a linear model can be used to link lens shift ($\Delta\theta$) to lens compensation (Δh). As shown in Fig. 3(a), in the absence of camera shake, light source

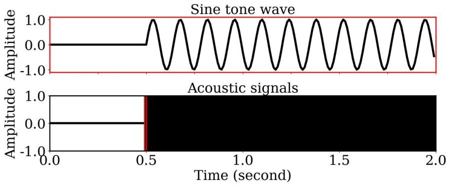


Figure 4: Acoustic injection signals – sin tone waves at $18795Hz$. The subfigure above shows the zoomed in acoustic signals of the red box in the below

A is imaged at pixel point B on the image sensor plane, such that: $\frac{h_a}{h_b} = \frac{L_a}{L_b}$, where L_a refers to the distance between the focal point and lens, and L_b refers to the distance between the focal point and image sensor. In the event of camera shake with rotational displacement $\Delta\theta$ (see Fig. 3(b)), light source A is mapped at point B' with a shift of Δh^*, causing the image to blur. Based on the geometry of optics, moving mapped point B' back upward to its original position B requires moving the lens upward by Δh (see Fig. 3(c)). Thus, Δh should satisfy the following formula:

$$\frac{h'_a + \Delta h}{h_b - \Delta h} = \frac{L_a}{L_b} \tag{1}$$

where h'_a can be obtained from the rotational displacement $\Delta\theta$ and the depth information Z of the light source A:

$$h'_a = (h_a - Z tan\Delta\theta) cos\Delta\theta \tag{2}$$

Thus, we obtain the necessary lens shift as follows:

$$\Delta h = \frac{h_b - \frac{h_a cos\Delta\theta L_b}{L_a} + \frac{Z sin\Delta\theta L_b}{L_a}}{1 + \frac{L_a}{L_b}} \tag{3}$$

Considering that the size of the image plane on a CMOS image sensor is normally far smaller than that of the lens module, and the image plane is very close to the focal point of the lens module. Thus, $L_a \gg L_b$. Also due to the small range of $\Delta\theta (\approx 0^o)$, the analysis above leads to the following OIS control objective:

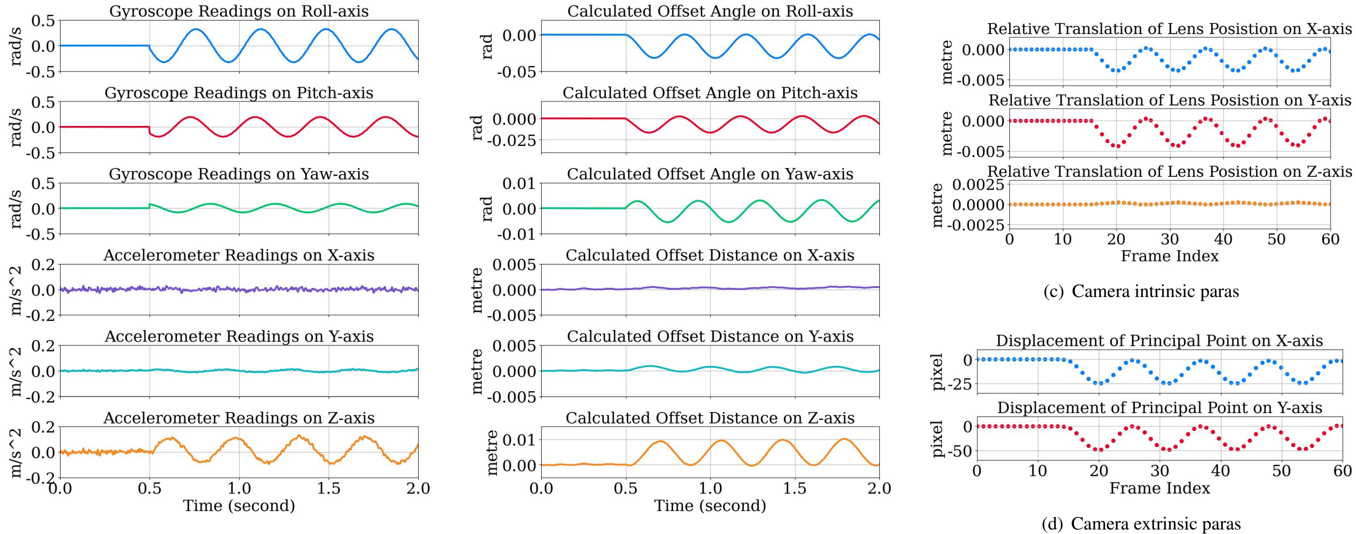
$$\begin{aligned}\Delta h &= \frac{h_b - \frac{h_a L_b}{L_a} + \frac{Z\Delta\theta L_b}{L_a}}{1 + \frac{L_a}{L_b}} \\ &= \frac{Z\Delta\theta}{1 + \frac{L_b}{L_a}} = Z\Delta\theta \rightarrow Z_c\Delta\theta\end{aligned} \tag{4}$$

where Z_c indicates the average constant depth in the weak perspective projection model [7], which assumes that all points on a 3D object are at the same distance when the depth of the object along the line of sight is small compared to the distance from the camera.

Thus, we can conclude that the lens movement required for translational control along the $x-$axis is linearly proportional to both the offset angles ($\Delta\theta$) and distances Δd with the constant terms (we call them OIS parameters). This is the situation encountered in the control systems commonly implemented for OIS modules in the digital cameras [51].

3.2 Altering Gyroscope Readings with Amplitude and Frequency Modulations

We have known that the OIS actuator adjusts the lens position according to the offset angles, which are calculated by the integration of the gyroscope readings. Thus, altering the gyroscope readings with



(a) 6-axis IMU readings

(b) Offset angles and distances

(c) Camera intrinsic paras

(d) Camera extrinsic paras

Figure 5: Corresponding results with a stationary Xiaomi 10 Ultra smartphone (fixed on the tripod) under the effects of acoustic signals with frequencies of $18795Hz$ (start at 0.5 seconds, see Fig. 4) that played by the built-in speaker

acoustic injection is a direct means to control the lens motion. Amplitude and phase modulations have been demonstrated on manipulating the gyroscope readings as the pre-defined values [46, 47]. Amplitude modulation is easy to be implemented by adjusting the speaker output power; however, phase modulation controlling method (also called phase pacing in [47]) has two significant limitations in the practical scenarios. One limitation is that the phase delay information on the gyroscope readings is not a constant, and it needs to be recalculated accurately before each control. The other is that the performance of phase modulation is extremely dependent on the accuracy of phase alignment between the acoustic signals and gyroscope readings, even small errors can lead to sudden changes in the angular value, making the movement of the lens uncontrollable. Therefore, in this paper, we use the frequency and amplitude modulations to achieve smooth control of gyroscope readings, and further enable the OIS actuator to control the lens motion regularly and stably.

A Xiaomi 10 Ultra with OIS camera is used as a test device here. We first identify the resonance frequency (*i.e.*, around $18.79KHz$) of the built-in MEMS gyroscope via frequency sweeping [46]. We then use the built-in speaker to play a *.wav* file of a sinusoidal acoustic signal with frequency modulation of $2Hz$ shift, and the detailed acoustic signals are shown in Fig. 4. Android APIs are used to collect 6-axis IMU readings at a sampling rate of 200 Hz, the results of which are shown in Fig. 5(a). We find that the 3-axis readings of the MEMS gyroscope are altered by the acoustic injection with modulated frequency ($2Hz$). We also observe that the 3-axis readings of the MEMS accelerometer are altered by the acoustic injection due to the similar sensing mass-spring structure [46]. To obtain the rotational displacement, we employ a rectangle approximation scheme based on the backward Euler method, as follows: $\theta(kT_s) = \theta((k-1)T_s) + T_s\omega(kT_s)$, where $\omega(kT_s)$ for $k \in \mathbb{N}^+$ is the measurements from the gyroscope. And translational displacement is calculated in the same way. The corresponding results are shown in Fig. 5(b). In supplementary video [1], we demonstrate that the position of the lens could indeed be controlled by the OIS module through the frequency and amplitude modulations on the acoustic injection signals played by the built-in speaker.

3.3 Conversion from Lens Controlling Signals to Camera Pose Parameters

In this section, we aim at verifying the correctness of the linear lens control model proposed above. Then, we examine changes in camera pose parameters caused by OIS-controlled lens motion. Camera calibration experiments are conducted to assess the impact of lens motion on camera pose parameters (intrinsic and extrinsic matrix). Specifically, we fix the Xiaomi 10 Ultra on the tripod and synchronously capture frames at 30 frames per second of a standard camera calibration checkerboard picture with the same acoustic injection used in Fig. 5(a). Note that the lens move slowly with 2 Hz, so we ignore the effect of rolling shutter in the stage of camera calibration.

We first define some of the operations here. Based on the pinhole camera projection model, 3D coordinates of a world point $\mathbf{X} = [X, Y, Z]^{\mathrm{T}}$ and its corresponding 2D coordinates in image $\mathbf{x} = [u, v]^{\mathrm{T}}$ can be described as follows:

$$s\mathbf{x} = \mathbf{P}\mathbf{X} = \mathbf{K}\mathbf{G}\mathbf{X}, \textit{where } \mathbf{K} = \begin{bmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{bmatrix}, \mathbf{G} = [\mathbf{R}|\mathbf{t}] \qquad (5)$$

where s is a scale factor, and $\mathbf{P}$ is the camera project matrix. For more details in the $\mathbf{P}$, $\mathbf{K}$ is the intrinsic matrix of a camera that contains focal lengths f_x and f_y, principal points c_x and c_y. $\mathbf{G}$ is the extrinsic parameters that represent the location of the camera in the 3-D scene and are consist of rotation matrix $\mathbf{R}$ and the translation vector $\mathbf{t}$.

[1] https://youtu.be/IZ1_tr5mquQ

When the OIS actuator moves the lens using relative translation vector $\Delta\mathbf{t_i} = [\Delta t_x^i, \Delta t_y^i, \Delta t_z^i]$ from the timestamp i to $i+1$, the extrinsic parameters have the following changes:

$$\mathbf{G_{i+1}} = [\mathbf{R_{i+1}}|\mathbf{t_{i+1}}] = \left[\begin{array}{c|c} \mathbf{R_0} & \begin{array}{c} t_{x0} + \Delta t_x^i \\ t_{y0} + \Delta t_y^i \\ t_{z0} + \Delta t_z^i \end{array} \end{array}\right] \tag{6}$$

The image sensor is fixed to the bottom of the camera module; therefore, any movement of the lens also alters the principal point parameters (*i.e.*, c_x and c_y). For example, when the lens moves down/up Δh along the y−axis, the c_y parameter will decrease/increase by $\frac{\Delta h}{p}$, where p is the physical length of each pixel edge (*e.g.*, 10 μm) on the image sensor. Therefore, the intrinsic parameters undergo the following changes between timestamp i to $i+1$:

$$\mathbf{K_{i+1}} = \begin{bmatrix} f & 0 & c_x^i + m\Delta t_x^i \\ 0 & f & c_y^i + m\Delta t_y^i \\ 0 & 0 & 1 \end{bmatrix}, \ where \ \ m = \frac{1}{l_p} \tag{7}$$

We utilize a two-stage camera calibration algorithm to generate the offset angles to lens motion vector ($\Delta\mathbf{t}^i$) actuated by the OIS. For initial calibration, the Camera Calibrator app in Matlab [31] is used to calculate total camera projection matrix P for each frame of a checkerboard of known physical size. We then obtain the following equations using the output projection matrix $\mathbf{P_i}$, $i \in [0, 1, 2, 3, ..., N-1]$, and N indicates the total number of frames,

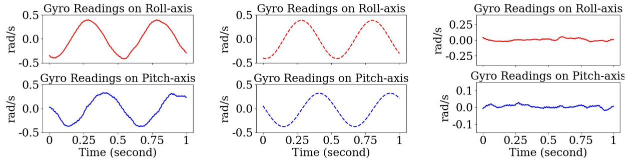
$$\begin{aligned} \mathbf{P_i} &= \mathbf{K_i}\mathbf{G_i} \\ &= \begin{bmatrix} f & c_x^0 + m\Delta t_x^i & 0 \\ 0 & f & c_y^0 + m\Delta t_y^i \\ 0 & 0 & 1 \end{bmatrix} \left[\begin{array}{c|c} \mathbf{R_0} & \begin{array}{c} t_x^0 + \Delta t_x^i \\ t_y^0 + \Delta t_y^i \\ t_z^0 + \Delta t_z^i \end{array} \end{array}\right] \end{aligned} \tag{8}$$

where f is the focal length, and c_x^0, c_y^0, t_x^0, t_y^0, t_z^0, $\mathbf{R_0}$ are the camera principal point parameters, translation vector, and rotation matrix of the first frame. By jointly establishing equations with the output camera project matrix, we can calculate the lens motion information (translation vector) and the principal point parameters. Corresponding results are shown in the Figs. 5(c) and 5(d). We also use the regression method to fit the conversation function from the offset angles ($\Delta\theta$) and distances ($\Delta\mathbf{d}$) to the relative camera pose parameters, and the results proved the effectiveness of the linear models which obtained from the theoretical derivation.

Thus, once we obtain the lens control signals for a given period (*i.e.*, offset angle information derived from gyroscope readings and offset distance information derived from accelerometer readings), it is possible to derive the relative changes in camera pose as follows:

$$\begin{aligned} \Delta t_x &= a_x\Delta\theta_x + b_x\Delta d_x, & a_x > 0, b_x < 0 \\ \Delta t_y &= a_y\Delta\theta_y + b_y\Delta d_y, & a_y > 0, b_y < 0 \\ \Delta t_z &= b_z\Delta d_z, & b_z < 0 \\ \Delta c_x &= m\Delta t_x, & m = 1/l_p \\ \Delta c_y &= m\Delta t_y, & m = 1/l_p \end{aligned} \tag{9}$$

where a_x, a_y, a_z, b_x, b_y are the coefficients of the OIS model, which can be calculated in the two-stage camera calibration described above. And m is known as the inverse value of the pixel length l_p in the image sensor which can be obtained from the datasheet. Note that these OIS coefficients and camera parameters can help us to obtain the lens motion information at the real-world scale, that means we can reconstruct the real depth information of the scene to the camera. Also noteworthy that the depth estimation algorithm we describe in Sec. 4 is also applicable to the uncalibrated OIS camera, since the OIS coefficients can be resolved during the process of depth map reconstruction.



(a) Superimposed total readings (b) Readings caused by acoustic injection (c) Readings caused by hand shake

Figure 6: Built-in gyroscope readings during the user handheld the camera with the acoustic injection

3.4 External Camera Motion Caused by Handhold Shooting

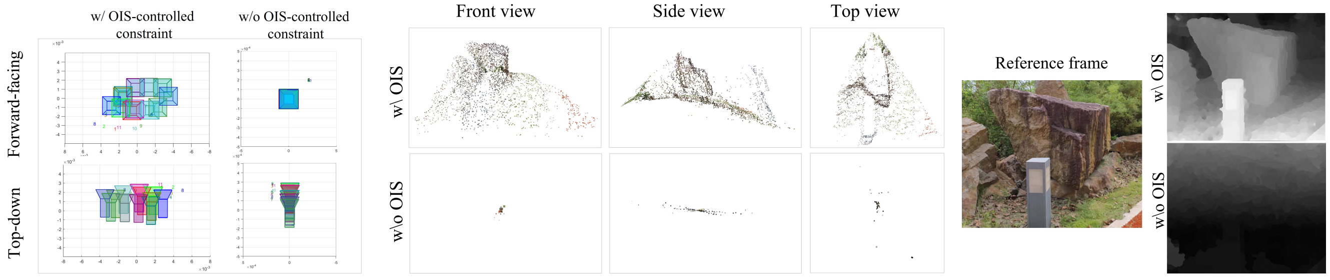
During the above preliminary experiments, we fixed the test camera with a tripod and analyzed the OIS-controlled lens motion model caused by altering the gyroscope readings. However, in addition to the scenarios with fixed cameras, there also exists handheld shooting scenarios, which means that the external camera shake caused by handheld may also interfere with the camera poses. Fortunately, in the depth estimation algorithm deign, we can ignore the external camera motion generated when the camera is handheld due to the fact that the OIS module can naturally compensate for subtle hand shake. Fig. 6(a) shows the superimposed results of the built-in gyroscope readings from the acoustic injection and the handheld camera. In the case of small hand shake, we can directly analyze the fitted sin-wave gyroscope readings in Fig. 6(b) to calculate the actual lens motion, because OIS automatically compensates for camera shake based on the shake information in Fig. 6(c). In Sec. 5.2.2, we will demonstrate the effectiveness of OIS in compensating for additional camera shake in the handheld shooting.

4 HIGH-QUALITY DEPTH RECONSTRUCTION

For reconstructing dense depth maps from small baseline caused by lens motion, we design a novel *S*tructure *f*rom *O*IS-controlled *M*otion algorithm (*SfOM*) to estimate the camera pose parameters. A high-quality dense depth map is then reconstructed via the plane sweeping [10] algorithm with the custom cost function.

4.1 *SfOM*: Structure from OIS-controlled Lens Motion

To achieve the high-quality depth map estimation, it is extremely important to recover the initial skeleton of the 3D structure as accurately as possible. We first capture a reference frame with the lens stationary and the offset frames with the OIS-controlled lens motion caused by the acoustic injection. Then, we elaborate a novel bundle adjustment to recover camera pose parameters and depth value of sparse feature points with high accuracy based on geometric and OIS-controlled model cues.



(a) Optimized camera poses (b) Reconstrctured sparse point clouds (c) Output depth maps

Figure 7: Comparing reconstructed camera poses, sparse 3D point clouds and output depth maps with and without OIS-controlled constraint in the bundle adjustment

4.1.1 Notation and Camera Geometry. We define π as the camera projection operator used to map a 3D point $\mathbf{X} = [X, Y, Z]^{\mathbf{T}}$ to image coordinates $\mathbf{x} = [u, v]^{\mathbf{T}}$. Likewise, π^{-1} is defined to be the back projection operator used to map pixel x and inverse depth w (= $\frac{1}{d}$, and d is the depth value) to a 3D point. Using the pinhole camera model with intrinsic parameters $\mathbf{K} = \begin{bmatrix} f_x & 0 & c_x \\ 0 & f_y & c_y \\ 0 & 0 & 1 \end{bmatrix}$ and radial distortion parameters $\mathbf{d} = (k_1, k_2)$, we have the undistorted coordinates:

$$\hat{\mathbf{x}} = \left[u + uk_1(\frac{u}{f_x})^2 + uk_2(\frac{u}{f_x})^4, v + vk_1(\frac{v}{f_y})^2 + vk_2(\frac{v}{f_y})^4\right]^{\mathbf{T}} \quad (10)$$

and the camera projection and back projection operators:

$$\begin{aligned} \pi(\mathbf{X}) &= \left[f_x \frac{X}{Z} + c_x, f_y \frac{Y}{Z}\right]^{\mathbf{T}}, \\ \pi^{-1}(\hat{\mathbf{x}}, w) &= \left[\frac{\hat{u} - c_x}{w f_x}, \frac{\hat{v} - c_y}{w f_y}, \frac{1}{w}, 1\right]^{\mathbf{T}}, \end{aligned} \quad (11)$$

The extrinsic matrix is represented using rigid body transform $\mathbf{G} = [\mathbf{R}|\mathbf{t}]^{\mathbf{T}}$. To find the image coordinates of point $\mathbf{G}$ in the frame i, we chain the projection and transformation: $(u, v) = \pi(\mathbf{G}_i\mathbf{X})$, where G_i is the ith camera extrinsic matrix.

Now, for the reference frame I_0 and the i–th offset frame I_i, assume that we have obtained the corresponding camera pose parameters $\mathbf{K}_0$, $\mathbf{K}_i$, $\mathbf{G}_0$, and $\mathbf{G}_i$. If we know the inverse depth w_i of a point $\mathbf{x}^0 = (u^0, v^0)$ in the reference frame, we can find it reprojected coordinates in the ith offset frame:

$$\mathbf{x}^i = \begin{bmatrix} u^i \\ v^i \end{bmatrix} = \pi_i(\mathbf{G}_i\mathbf{G}_0^{-1}\pi_0^{-1}(\hat{\mathbf{x}^0}, w_0)) = \pi_i(\mathbf{G}_{0i}\pi_0^{-1}(\hat{\mathbf{x}^0}, w_0)) \quad (12)$$

using the notation $\mathbf{G}_{0i} = \mathbf{G}_i\mathbf{G}_0^{-1}$ for the relative extrinsic matrix between the reference frame and the ith offset frame.

Considering the condition that a user is handholding a camera to take image sequences, the user's hank shake will also affect the built-in IMU sensor readings except the acoustic injection (see Fig. 6(a)). However, the fact is that the OIS actuator can compensate for the actual camera movement caused by the hand shake and make the optical paths stable (see Fig. 3). Therefore, the lens motion caused by the acoustic injection is the main component that affect the camera pose parameters. According to the linear superposition of the force driving on the MEMS sensing mass, we can simply extract the IMU sensor readings caused by the acoustic injection (see Fig. 6(b)), which will be used to model the lens motion. Therefore, we can obtain the G_{0i} as follows:

$$\mathbf{G_{0i}} = [R(\Delta\mathbf{r_i})|\Delta\mathbf{t}] = \left[\begin{array}{ccc|c} 1 & -\Delta r_i^z & \Delta r_i^y & \Delta t_i^x \\ \Delta r_i^z & 1 & -\Delta r_i^x & \Delta t_i^y \\ -\Delta r_i^y & \Delta r_i^x & 1 & \Delta t_i^z \end{array}\right] \quad (13)$$

where $\Delta r_i \longrightarrow \mathbf{0}$ is the relative angle vector (should be zero) caused by the hand shake between the ith offset frame and the reference frame, and $\Delta\mathbf{t}$ is caused by the lens motion controlled by the OIS actuator with the acoustic injection component. We also obtain the $\mathbf{K}_i$ from the $\mathbf{K}_0$ as follows:

$$\mathbf{K}_i = \begin{bmatrix} f & 0 & c_x + \Delta c_{xi} \\ 0 & f & c_y + \Delta c_{yi} \\ 0 & 0 & 1 \end{bmatrix}, \mathbf{K}_0 = \begin{bmatrix} f & 0 & c_x \\ 0 & f & c_y \\ 0 & 0 & 1 \end{bmatrix} \quad (14)$$

In this way, We can obtain the lens controlling signals with the sin-wave-like gyroscope and accelerometer readings which are altered by the acoustic injection, and estimate the camera pose parameter with the linear model from Eqs. 9.

4.1.2 Feature Extraction and Tracking with Rolling Shutter Compensation. To solve the camera pose recovery, the bundle adjustment framework requires a set of feature correspondences across the image sequences. We initially utilize the well-known Harris corner [21] and Kanade-Lucas-Tomasi (KLT) tracker [43] to extract sub-pixel corner features in the reference frame for tracking through the sequence. Considering that rolling-shutter image sensors will generate distortion when a moving lens is used to capture images, and a small error in feature point tracking results can generate significant artifacts on the whole reconstruction especially in the condition of narrow baseline. Therefore, the rolling shutter effect must be corrected to minimize the error of camera pose estimation.

We model the continuous linear changes in lens translation actuated by OIS, where the assignment of features was based on their vertical position within the image. Note that this is achieved by interpolating the translation vectors and principal points between two successive frames. The translation vector and the coordinates of features in two consecutive frames are modeled as follows:

$$\begin{aligned} \mathbf{t}_{ij} &= \mathbf{t}_i + \frac{t_r k_{ij}}{t_f}(\mathbf{t}_{i+1} - \mathbf{t}_i) \\ u_{ij} &= u_{ij} - \frac{t_r k_{ij}}{l_p t_f}(\mathbf{t}_{i+1} - \mathbf{t}_i) \end{aligned} \quad (15)$$

where $\vec{\mathbf{t}}_{ij}$ and u_{ij} are the translation vectors and coordinates for the j-th features on the i-th offset frame respectively. k_{ij} stands for the row number of each feature point, t_f and t_r are the frame time and the shutter time interval, and l_p is the pixel length of the image sensor. Thus, the camera extrinsic G_{ij} and the coordinates u_{ij} are updated to cancel out the rolling shutter effect.

4.1.3 Bundle Adjustment.

We choose bundle adjustment to jointly optimizes camera poses and inverse depth of tracked points, which depends on a nonlinear optimization method by minimizing reprojection error [45]. In our geometric model based on the camera model proposed in Sec. 4.1.1, we first use Eq. 10 to calculate the undistorted coordinates $\hat{x_{ij}}$ for the j-th feature in the i-th offset image relative to the center of the image. We then use the distance between the undistorted coordinates of the extracted feature points (Sec. 4.1.2) and the reprojection coordinates to represent the reprojection error of x_j^i. Finally, we formulate the bundle adjustment with the aim of minimizing the reprojection errors of all features in the non-reference images:

$$\underset{\mathbf{K,R,t,W,O}}{\operatorname{argmin}} \sum_{i=1}^{N-1} \left[\sum_{j}^{M-1} \rho(\hat{x_j^i} - \pi_i(\mathbf{G}_{0i}\pi_0^{-1}(\hat{\mathbf{x}_\mathbf{j}^\mathbf{0}}, w_j)) + \gamma||\mathbf{R}_i||_2 \right] \tag{16}$$

where N is the number of the captured images, M the number of extracted feature points, $\rho(\cdot)$ the element-wise Huber loss function [23], $\mathbf{K}$ is the set of the intrinsic camera parameters for the non-reference images, $\mathbf{R}$ and $\mathbf{t}$ are the sets of the rotation and translation vectors for the non-reference images, $\mathbf{W}$ is the set of inverse depth values of the feature points, and $\mathbf{O}$ is the lens control parameters (a_x, a_y, b_x, b_y) (see Eqs. 9), and γ is the penalty weight which is set as 0.5 here.

To obtain the initial parameters for bundle adjustment, we set the rotation matrix caused by hand shake to zero. The focal length f_x, f_y, the principal points c_x, c_y, and two radial distortion parameters are assigned values obtained from the camera calibration or the Android APIs [3, 12]. For the lens control parameters $\mathbf{O}$, if the OIS camera has been pre-calibrated with the calibration algorithm described in Sec. 3.3, we fix these parameters as the pre-calibrated results; if the OIS camera is not calibrated, we set the lens control parameters at $a_x = a_y = 0.001$, $b_x = b_y = b_z = -0.001$ as the initial values and assigned p a small value of 0.000001. For the inverse depths, we assign a random value between 0.01 and 1.0 for each feature.

The major advantage of the proposed *SfOM* algorithm is the fact that it does not fall into a local optimal solution or fail to converge due to the inclusion of additional OIS-controlled constraints when seeking to obtain camera pose parameters. Thus, even when working with a micro-scale stereo baseline caused by the lens motion, we find that our bundle adjustment can successfully converge with a reasonable approximation for the camera pose parameters (see Fig. 7(a)), and the high accuracy 3D reconstructions of the tracked sparse feature points with minimal back projection error (see Fig. 7(b)). The 3D points obtained in this step are subsequently used for the high-quality dense depth map reconstruction (see Fig. 7(c)).

4.2 Dense Depth Reconstruction

Once the accurate camera pose parameters are obtained from the previous stage, we base our dense depth reconstructions on the plane sweeping algorithm [10]. For the $k-$th depth in n_k sweeping depths, all the images are warped by back-projecting them onto a virtual plane at a given inverse-depth w_k from the reference viewpoint, and then projected onto the reference image domain. The plane-induced homography H_{ik} that describes the transformation from the reference image domain coordinates to the $i-$th offset image domain coordinates when passing through the virtual plane at the $k-$th sweeping depth can be formulated by:

$$\mathbf{H}_{ik} = \mathbf{K}_i \begin{bmatrix} 1 & -\Delta r_i^z & \Delta r_i^y + w_k \Delta t_i^x \\ \Delta r_i^z & 1 & -\Delta r_i^x + w_k \Delta t_i^y \\ -\Delta r_i^y & \Delta r_i^x & 1 + w_k \Delta t_i^z \end{bmatrix} \mathbf{K}_i^{-1} \tag{17}$$

Using this homography, the $i-$th offset image I_i^p can be warped into the reference image domain through the operation described by the following formulation:

$$I_{ik}(\mathbf{p}) = I_i^p(\pi(H_{ik}\mathbf{p})) \tag{18}$$

After warping n images, every pixel $\mathbf{p}$ in the reference image domain has an intensity profile $P(\mathbf{p}, w_k) = [I_{1k}(\mathbf{p}), \cdots, I_{(N)k}(\mathbf{p})]$ for the inverse depth candidate w_k. And our matching cost C_I for pixel $\mathbf{p}$ and depth candidate w_k is defined as follows:

$$\begin{gathered} C_I(\mathbf{p}, w_k) = VAR([\mu_1 I_{1k}(\mathbf{p}), \cdots, \mu_N I_{Nk}(\mathbf{p})]) \\ where,\ \ \mu_i = \frac{||t_i||}{\sum_{i=1}^{N} ||t_i||} \ \ i = 1, 2, ..., N \end{gathered} \tag{19}$$

where $VAR(\cdot)$ is the variance calculation function. Unlike the prior dense depth reconstruction work [19], we give a higher weight to the offset frames that are further away from the reference frame. Given the fact that in multi-view geometry, a narrower baseline leads to lower certainty in the estimated depth since it is more sensitive to perturbations in the projected point positions.

In order to enforce the matching fidelity on the edge regions of the image, we introduce two additional costs $C_{\delta u}$ and $C_{\delta v}$ defined as the horizontal and vertical gradients of the images, respectively. And the comprehensive matching cost C is defined as:

$$C = C_I + \lambda(C_{\delta u} + C_{\delta v}) \tag{20}$$

In the last, we apply the winner-takes-all strategy on the cost volume C, and follow the depth refinement mechanism proposed in [19] to get the final dense depth map D_{out}.

5 EVALUATION

5.1 Experimental Setup

In this study, we implement a prototype of our proposed DoCam on the mainstream smartphones whose built-in camera equipped with a lens-shift OIS module, and evaluate the performance of the output depth map. The whole videos for the depth map estimation are captured in the resolution of 1440×1080 with 30 frames per second. We also record the 6-axis IMU readings with the maximum sampling rate synchronously. A laptop equipped with an Intel i7-10875H 2.80GHz CPU with 32.0GB RAM, and an NVIDIA GeForce RTX 2060 with a 6.0GB VRAM was used for the whole postprocessing computation.

To verify the depth estimation performance of the DoCam, we applied the infrared ToF sensor, Azure Kinect [4], to collect the ground-truth depth information. In details, we calibrated the relative camera pose of the test camera and the IR camera built in the

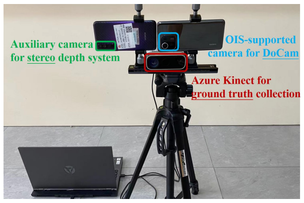
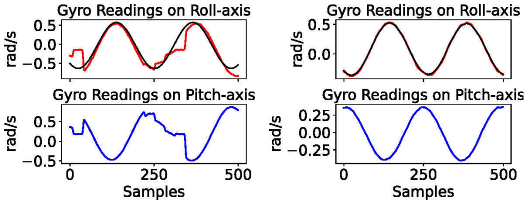

Auxiliary camera for stereo depth system
OIS-supported camera for DoCam
Azure Kinect for ground truth collection

Figure 8: Experiment setup: DoCam, stereo camera system, and Azure Kinect



(a) Speaker power with 1 W (b) Speaker volume with 5 W

Figure 9: Gyroscope readings altered when the speaker played acoustic injection signals with different power

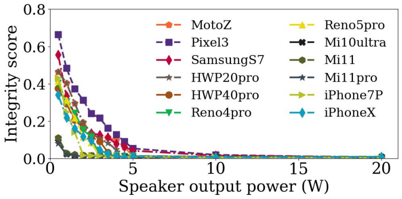


Figure 10: Integrity of the built-in gyroscope readings vs. speaker output power. The external speaker is $10cm$ away from the testing cameras

Kinect, and remapped the depth map to the test camera's view. Meanwhile, we built a stereo RGB depth system with another stationary smartphone for performance comparison, and the relative camera pose of the two smartphones were also calibrated beforehand for the stereo depth estimation algorithm [49]. The experiment set up is shown in Fig. 8. Note that if not specifically instructed, the following experiments were conducted with a tripod.

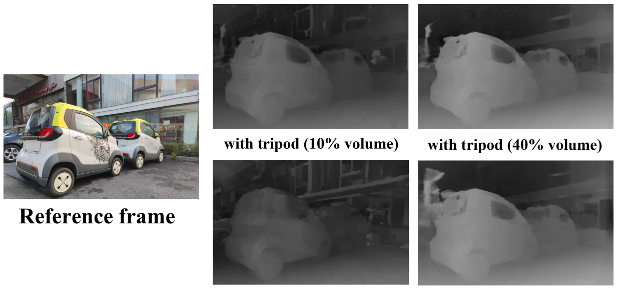
5.2 Micro Benchmark

5.2.1 Parameter Selection in Speaker Volume. Applying acoustic injection to MEMS gyroscopes requires an appropriate acoustic signal frequency as well as signal energy sufficient to create a pressure wave capable of affecting the mass driver. Fig. 9 presents two example axes of the gyroscope readings of the Moto Z that is fixed on the tripod, when using an external speaker (YAMAHA HS5) to perform acoustic injection with $1Watt$ and $5Watt$ volumes respectively. The results in Fig. 9(a) show that when the acoustic signal is not energetic enough, the sensing mass in the gyroscope cannot resonate sufficiently which causes the final output reading (red line) to be incomplete sin wave pattern (black line).

We measure the signal integrity score by calculating the Euclidean distance between the normalized gyroscope readings and the expected sin wave, and the smaller score means the better signal integrity. We adjust the different volumes of testing device, and record the relationship between the integrity score and the speaker output power in the Fig. 10. The results show that the output power $5W$ can be satisfied among the mainstream mobile phones released in the past three years [1]. We also test some mainstream phones with the outstanding built-in speakers, *i.e.*Xiaomi 10 Ultra, Xiaomi 11, and Xiaomi 11Pro, to play corresponding *.wav* files with different volumes. The results in the Table 1 show that when modulating the speaker volume as $\geq 20\%$, the built-in gyroscope readings can be controlled well on the experimental smartphones. However, too much energy of acoustic injection signals makes the gyroscope readings vary more, which causes the OIS module to exceed the maximum compensation range. We selected a representative camera (Xiaomi 10Ultra) to evaluate the depth estimation performance under the different volume settings, the depth maps generated with different volumes were shown in Fig. 11. It can be noticed that too little volume (*e.g.*, 10%) leads to insufficient movement of the lens, making the estimated depth map perform poorly; too much volume (*e.g.*, 100%) causes our *SfOM* algorithm to make errors in estimating camera poses based on gyroscope readings and further results in an unsatisfied depth map. In the following experiments, we modulated the speaker volume of the test smartphone as 40% to achieve the regular control of the lens motion in the OIS-camera.

Volume	5 %	10 %	20 %	40 %	80 %	100 %
Score	0.107	0.076	**0.015**	**0.009**	**0.007**	**0.004**

Table 1: The average integrity scores of the built-in gyroscope readings under different speaker volumes. The smaller score, the better controlling performance



Reference frame

with tripod (10% volume) with tripod (40% volume)

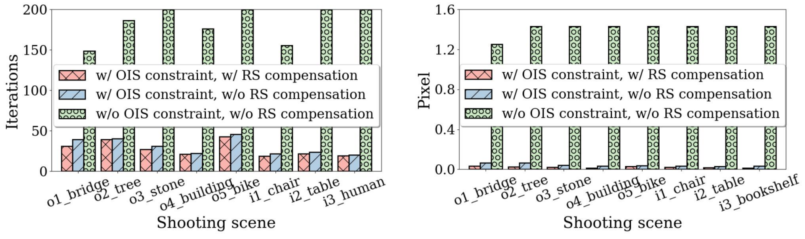
with tripod (100% volume) handhold (40% volume)

Figure 11: Dense depth map results generated with different speaker volumes and shooting modes

5.2.2 Duration Determine of Handhold Shooting. To verity the effectiveness of the OIS module on compensating the hand shake, we hire eight volunteers to hold the Xiaomi 10 Ultra in turns and record the video of the calibration checkerboard at a distance of one meter for 6 seconds. Then we analyze the max average pixel shift of feature tracking points (corners of each boxes in the checkerboard) between the offset frames and the reference frames. The results are shown in Table 2, we find that the small ranges of the hand shake within 2 seconds could be well compensated by OIS, with ≤ 0.5 pixels tracking error. In the following experiments, our proposed system only records the frames within 2 seconds after the user presses the shutter in the handhold shooting scenarios, and these frames are then for the depth estimation. We give an end-to-end comparison of depth estimation in the cases of handheld and with tripods. The corresponding depth maps in Fig. 11 prove that our system is naturally suitable for handheld shooting scenarios.

Duration (s)	0 ~ 1	1 ~ 2	2 ~ 3	3 ~ 4	4 ~ 5	5 ~ 6
Error (Pixel)	**0.0123**	**0.136**	0.577	1.15	4.1354	5.363

Table 2: Performance of the OIS module to compensate for camera shake when handheld shooting



(a) Number of iterations.

(b) Average reproject error.

Figure 12: Quantitative analysis of camera pose estimation under eight scenes with and without the OIS-controlled signals and rolling shutter compensation respectively

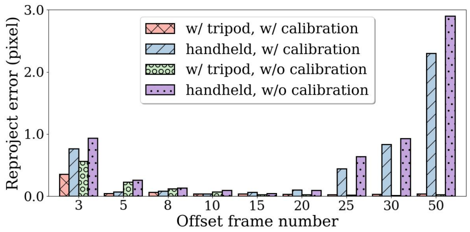


Figure 13: Influence of the number of offset frames on quality of depth reconstruction in terms of the reprojection error

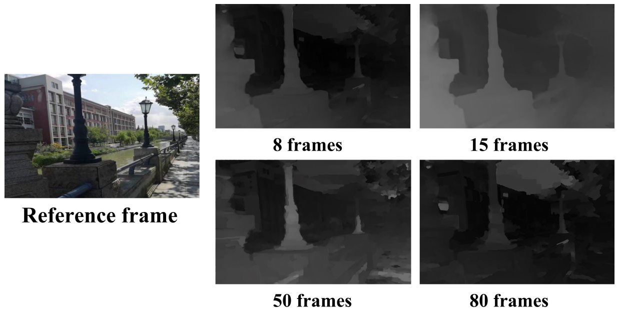


Figure 14: Dense depth map results generated with different number of the offset frames

5.3 Methodology Evaluation

5.3.1 Effectiveness of the Proposed SfOM. To verify the usefulness of our proposed *SfOM*, we perform quantitative analysis under various shooting scenes and presented the results in Fig. 12. Note that, during the dataset collection, we capture one reference frame and 30 offset frames during each scene shooting; during the bundle adjustment, the lens control coefficients ($a_x \sim b_y$) are treated as unknown and initialized as the corresponding values described in Sec. 4.1.3. The experiment results show that our proposed *SfOM* algorithm can indeed work well on the uncalibrated camera and provide two advantages (1) it requires fewer iterations to figure out the optimal solution because of the strong constraint imposed by the OIS-controlled lens motion; (2) the output of camera poses also provides higher precision with the smaller reprojection errors. Meanwhile, we also verify that the rolling shutter compensation indeed help to improve the precision of camera pose estimation.

5.3.2 Depth Quality and the Number of Offset Frames. Our work remains essentially a multi-view stereo (MVS) based depth estimation scheme where we focus on the micro-baseline caused by the lens motion. Thus, we vary the number of offset frames during the depth estimation with two types of modes, one represents a pre-calibrated camera with the known lens control coefficients, and the other is an uncalibrated camera with the unknown lens control coefficients. The experimental results are shown in Fig. 13. For the cameras with pre-calibrated OIS parameters, we observe the reprojection error is small enough with 5 offset frames. For the cameras with unknown OIS parameters, we find that the reprojection error decreases to the minimum value when the number of offset frames is 15. However, when the offset frame number increases from 15 to 50 in the handheld shooting scenario, the additional camera motion brought by handshake cannot be compensated by the OIS itself and causes the lens motion controlled by the acoustic injection to bring wrong constraints, resulting in the no convergence in solving the camera pose with bundle adjustment and bad depth reconstruction (see Fig. 14). In practical application scenarios, for the pre-calibrated cameras, we set the offset frame number as 5 to obtain the high-quality depth map; for the uncalibrated cameras, we set offset frame number as 15.

5.4 Qualitative and Quantitative Evaluation on the Output Depth Maps

For the qualitative and quantitative evaluation, we compare our dense depth map results with those obtained from a deep-learning monocular depth estimation (Adabins [5]), a conventional depth estimation from small motion (DfUSMC [19]), a conventional stereo depth estimation method [6, 49] and a hardware-based solution (Azure Kinect). For each shooting scene, we firstly recorded the reference frame of the shooting scene with the test OIS-support camera while the lens was stationary, and the stereo frame was recorded by the auxiliary camera simultaneously. Then, we used the acoustic injection to drive the lens motion and recorded the entire video to collect the multiple frames with OIS-controlled moving lens. In the last, we moved the smartphone around 5 *cm* deliberately to generate the adequate baselines for DfUSMC.

We present the end-to-end comparative results of depth maps for two outdoor scenes and two indoor scenes in Fig. 15. The qualitative comparison results show that DoCam yields the best performance than the other depth estimation algorithms. For Adabins, a supervised learning model, it lacks generalization and cannot recover the 3D information for the unseen complex scenes. For DfUSMC, the main reason for its unsatisfactory performance is that the conventional SfM algorithm is unable to recover the exact camera poses during the small motions caused by the handheld camera. For the stereo camera system with the known baseline (16 *cm*), the accuracy of depth estimation is mainly rely on the correspondences of featured points. However, stereo observation with one fixed location makes the depth estimation performance unstable. In contrast, our proposed DoCam can improve the accuracy of the correspondences of featured points to some extent, thanks to the multiple observations from offset frames. When combined with the precise information of the lens motion (*i.e.*, camera pose) under the OIS-controlled geometry constraints, DoCam can finally obtain the accurate dense depth maps of the shooting scenes.

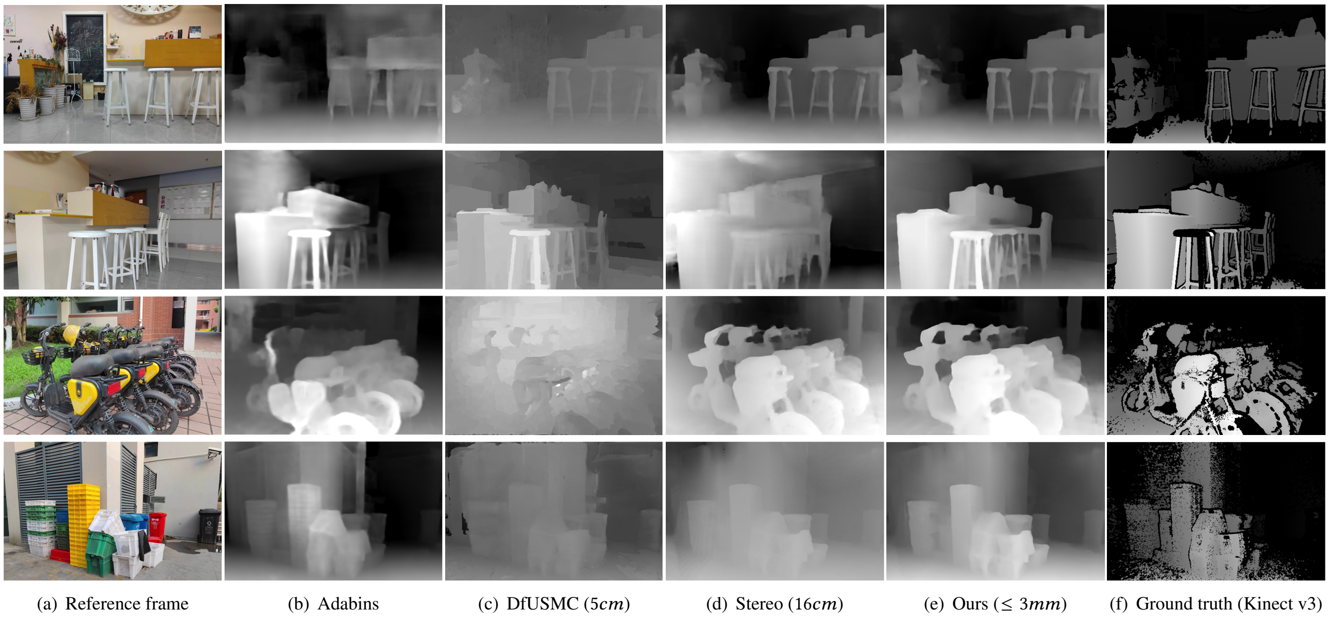

(a) Reference frame (b) Adabins (c) DfUSMC ($5cm$) (d) Stereo ($16cm$) (e) Ours ($\leq 3mm$) (f) Ground truth (Kinect v3)

Figure 15: End-to-end dense depth map comparison of our proposed DoCam and other depth estimation systems

We also quantitatively analyze the performance of depth estimation methods with the ground truth obtained from Azure Kinect. In details, for each valid pixel (*i.e.*, non-black pixels in Fig. 15(f)), we calculate the error between the estimated depth and GT to obtain the average accuracy as one metrics. We also add R10 and R20 as another two metrics for the better understand the performance evaluation among depth estimation methods, where R10 and R20 are the percentage of pixels that have a estimation error of less than 10% and 20% of the maximum depth value (*i.e.*, 5m). The results are listed in Table 3, it shows that our proposed DoCam also performs best in the quantitative analysis.

Metrics	Adabins	DfUSMC ($5cm$)	Stereo ($16cm$)	**Ours**
Accuracy(%)	35.1	14.5	75.4	**87.9**
R10(%)	41.34	17.56	83.56	**93.12**
R20(%)	63.57	47.11	91.03	**99.04**

Table 3: Qualitative evaluation of different depth estimation algorithms using the ground truths from Azure Kinect

6 APPLICATIONS

In this section, we discuss a number of potential applications for DoCam. We mainly list two main application scenarios. The first is the surveillance/IP cameras, which are fixed in position and view. These cameras are usually equipped with OIS to prevent shaking due to road and wall vibrations [11]. By using the ubiquitous speakers around the cameras, such as the broadcast speakers, to play the inaudible acoustic injection signal to drive the OIS module to control the lens motion, our system can be deployed to predict the depth information of the scene and further prompt object localization application. The second is the smartphone cameras that supports OIS. Our system allows the user to obtain the photo-consistent depth map by holding the smartphone without moving any distances, *i.e.*, like pressing a single shutter. The depth information can enable applications, such as live face authentication and digital refocusing .

Object Localization. We pre-calibrate the test camera and utilize our proposed DoCam to output depth map for each reference frame with 5 offset frames, the less offset frame number enables the localization system to track the moving object. In the object localization, we utilize the mask R-CNN [22] to detect the target human and use the average depth of the pixels in the center of the human detection box in our output depth map to calculate the average distance. The same approach is applied to obtain the ground truth distance from the registered depth map output from Azure Kinect. The experimental results show that our system can obtain a localization error of approximately only 0.71 m with the sensing range of 5 m in both indoor and outdoor scenes.

Live Face Authentication. Authentication with the user's face is very common in mobile apps. However, in most cases, 2D RGB cameras without IR sensors or dot projectors can easily be fooled by photos/videos displayed on screen (*e.g.*, grabbed from social media). Therefore, liveness detection is significant to the face authentication. Prior related work [13] utilized the smartphone screen to implement the liveness detection via illuminating a user's face from different directions. With the help of our proposed DoCam system that turns a RGB camera into a depth camera, we can also realize the similar liveness detection. Fig. 16 shows the depth maps of a user's photo shown on the smartphone screen and a live face of the user respectively, the results verify the effectiveness of the DoCam when recognizing the live faces.

Digital Refocusing. The reconstructed depth map can also enhance the mobile photography that are nearly impossible with a single color image. We use the 3D reconstruction information to simulate different aperture effects or synthesize new views. To test our depth map is good enough for such applications, we generate the refocusing results of the reference image from different focus point. As shown in Fig. 17, the generated depth map can clearly show the depth change (*i.e.*, from close to far) of the objects in the scene.

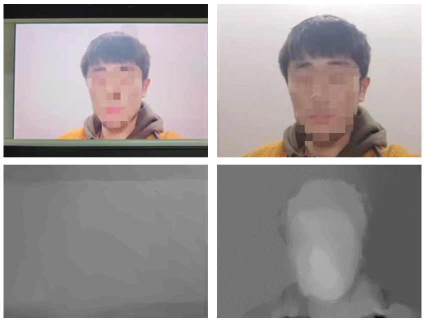


Figure 16: Liveness detection in face authentication

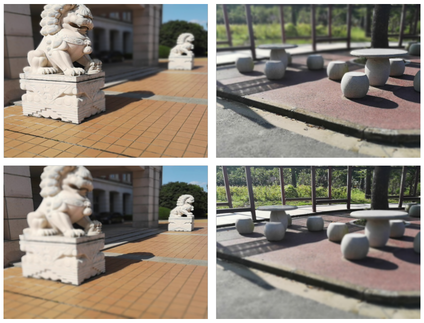


Figure 17: Refocusing results based on the depth maps output from DoCam

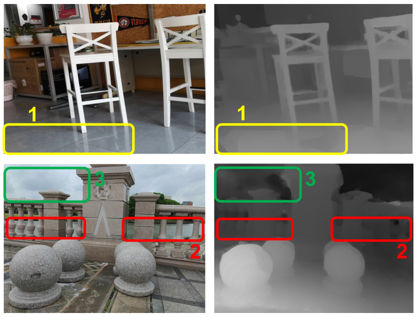


Figure 18: Limitations of DoCam on the dense depth estimation

7 DISCUSSION AND LIMITATIONS

In this section, we discuss the limitation of DoCam and the practical issues that may happen during the usage.

7.1 Computational Time

To generate an accurate dense depth map from one reference frame and 15 offset frames (1920×1080 resolution) with the micro-scale stereo baseline caused by the OIS-controlled lens motion, the implementation of our proposed *SfOM* algorithm takes about $4-6$ *min*. In the current version of our system implementation, we cannot realize the real-time depth map estimation due to the lack of the algorithm computation optimization. However, there are some possible solutions to increase the frame rate of our system. For scenarios that require high frame rates, we can first reduce the output depth map resolution to 360×270; then, we can optimize the algorithms for feature extraction, tracking and bundle adjustment stages respectively, and use CPU/GPU parallelization techniques (*e.g.*, Gipuma [16]) on the final dense stereo matching to decrease computational time and achieve a high frame rate on the depth estimation.

7.2 Depth Sensing Accuracy and Range

Unlike conventional MVS algorithms, DoCam enables good depth map estimation due to the precise camera pose recovery with the OIS-controlled lens motion. In addition to camera pose, correspondence is another significant factor affecting the performance of DoCam in estimating depth. We list three instances of inaccurate depth estimation brought by the wrong correspondences. (1) Reflection and shadow will lead to corresponding inaccuracies. Box 1 of Fig. 18 gives an example of error depth estimation in the mirror region of the chair. (2) Repetitive patterns also bring the incorrect correspondences. Box 2 in Fig. 18 gives two examples of error depth estimation in the repeated patterns – lake surface and bridge rails. (3) Particularly distant objects bring great difficulties to the correspondence solutions. Box 3 in Fig. 18 shows inaccurate depth estimates for distant clouds, which are predicted to have the similar depth values as bridge rails.

Sensing range of the monocular stereo based depth estimation algorithms is proportional to the length of the baseline. Although the baseline of DoCam is micro (≤ 3 *mm*), with the help of a priori knowledge of OIS-controlled lens motion and more precise correspondences brought about by multiple observations from offset frames, our system can achieve a maximum depth range of 5 *m* with 10% accuracy. Although our proposed DoCam will have a worse accuracy in the far range, the 5 *m* of depth sensing range with guaranteed depth resolution enables abundant applications.

7.3 Nearby Smartphones and Ambient Noises

In the scenario where users are close to each other (*e.g.*,50 *cm*) and use DoCam at the same time, the acoustic injection from the user's smartphone speaker will not cause interference to other users around. There are mainly two points to explain this phenomenon: (1) MEMS gyroscope in the different smartphones have different resonant frequencies [47], and its readings are hardly be altered by the acoustic injection signals played by the other smartphones; (2) Even if two users use the same model of smartphones at a distance of 50 *cm*, in this case, it is unlikely for one smartphone's speaker to interfere with the other smartphone's gyroscope readings due to the strong attenuation of the acoustic signals. We conducted an experiment with two smartphones (Xiaomi 11) with the volume set to 100%, and we found that only when the distance between the two phones was too close (*e.g.*, ≤ 10 *cm*), there exists interference between each other. However, when the two smartphones are 50 *cm* apart, the interference is completely ignorable.

For the interference of ambient noises, since our proposed DoCam uses acoustic injection signals at frequencies in the inaudible band of the human ear, only the ambient signals that are particularly close to the resonance frequency of the gyroscope (*e.g.*, 18795 ± 20 *Hz* for Xiaomi 11) will interfere DoCam. However, such high frequency ambient noises do not often exist frequently in daily life [14].

8 CONCLUSION

In this paper, we dig into the potential of the existing OIS techniques in depth sensing and propose DoCam, the first work that utilizes the OIS-controlled lens motion to perceive metric depth of scene. Our system is able to achieve high-quality depth estimation without additional camera movement, making it particularly suitable for scenarios where the camera is fixed and requiring surrounding 3D information. We finally present how DoCam can facilitate new applications when only a single OIS-supported RGB camera is available.

ACKNOWLEDGMENTS

We thank the anonymous reviewers for their constructive comments. This work is supported in part by NSFC (62072306, 61936015) and Program of Shanghai Academic Research Leader (20XD1402100).

REFERENCES

[1] 91mobiles. 2022. Phones with optical image stabilization. https://www.91mobiles.com/list-of-phones/phones-with-ois. (2022).

[2] S Abhishek Anand and Nitesh Saxena. 2018. Speechless: Analyzing the threat to speech privacy from smartphone motion sensors. In *2018 IEEE Symposium on Security and Privacy (SP)*. IEEE, 1000–1017.

[3] Android ARCore. 2022. CameraIntrinsics. https://developers.google.com/ar/reference/java/com/google/ar/core/CameraIntrinsics. (2022).

[4] Microsoft Azure. 2022. Azure Kinect DK Build for mixed reality using AI sensors. https://azure.microsoft.com/en-us/services/kinect-dk/. (2022).

[5] Shariq Farooq Bhat, Ibraheem Alhashim, and Peter Wonka. 2021. Adabins: Depth estimation using adaptive bins. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4009–4018.

[6] Thomas Brox, Andrés Bruhn, Nils Papenberg, and Joachim Weickert. 2004. High accuracy optical flow estimation based on a theory for warping. In *European conference on computer vision*. Springer, 25–36.

[7] Alfred M Bruckstein, Robert J Holt, Thomas S Huang, and Arun N Netravali. 1999. Optimum fiducials under weak perspective projection. *International Journal of Computer Vision* 35, 3 (1999), 223–244.

[8] Gannon Burgett. 2021. Digital Photography Review Report: Apple expected to use sensor-shift image stabilization units in all of its next-generation iPhone models. https://www.dpreview.com/news/7769281511/report-apple-expected-sensor-shift-image-stabilization-units-next-generation-iphone-models. (2021).

[9] Google Clay Bavor, VP. 2021. Project Starline: Feel like you're there, together. https://blog.google/technology/research/project-starline/. (2021).

[10] Robert T Collins. 1996. A space-sweep approach to true multi-image matching. In *Proceedings CVPR IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 358–363.

[11] DETEC. 2022. IP Cameras Tagged optical image stabilization (OIS). https://detec.no/collections/ip-cameras/optical-image-stabilization-ois. (2022).

[12] Android developer. 2022. LENS DISTORTION. https://developer.android.com/reference/android/hardware/camera2/CameraCharacteristics. (2022).

[13] Habiba Farrukh, Reham Mohamed Aburas, Siyuan Cao, and He Wang. 2020. FaceRevelio: a face liveness detection system for smartphones with a single front camera. In *Proceedings of the 26th Annual International Conference on Mobile Computing and Networking*. 1–13.

[14] Gregory A Flamme, Mark R Stephenson, Kristy Deiters, Amanda Tatro, Devon Van Gessel, Kyle Geda, Krista Wyllys, and Kara McGregor. 2012. Typical noise exposure in daily life. *International journal of audiology* 51, sup1 (2012), S3–S11.

[15] Sergi Foix, Guillem Alenya, and Carme Torras. 2011. Lock-in time-of-flight (ToF) cameras: A survey. *IEEE Sensors Journal* 11, 9 (2011), 1917–1926.

[16] Silvano Galliani, Katrin Lasinger, and Konrad Schindler. 2015. Massively parallel multiview stereopsis by surface normal diffusion. In *Proceedings of the IEEE International Conference on Computer Vision*. 873–881.

[17] Rahul Garg, Neal Wadhwa, Sameer Ansari, and Jonathan T Barron. 2019. Learning single camera depth estimation using dual-pixels. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 7628–7637.

[18] Dariu M Gavrila and Larry S Davis. 1996. 3-D model-based tracking of humans in action: a multi-view approach. In *Proceedings cvpr ieee computer society conference on computer vision and pattern recognition*. IEEE, 73–80.

[19] Hyowon Ha, Sunghoon Im, Jaesik Park, Hae-Gon Jeon, and In So Kweon. 2016. High-quality depth from uncalibrated small motion clip. In *Proceedings of the IEEE conference on computer vision and pattern Recognition*. 5413–5421.

[20] Christian Häne, Christopher Zach, Jongwoo Lim, Ananth Ranganathan, and Marc Pollefeys. 2011. Stereo depth map fusion for robot navigation. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 1618–1625.

[21] Chris Harris, Mike Stephens, et al. 1988. A combined corner and edge detector. In *Alvey vision conference*. Citeseer, 10–5244.

[22] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. 2017. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*. 2961–2969.

[23] Peter J Huber. 1992. Robust estimation of a location parameter. In *Breakthroughs in statistics*. Springer, 492–518.

[24] Mark C Hughes. 2022. Understanding Sensor-Shift Technology for High-Resolution Images. https://digital-photography-school.com/understanding-sensor-shift-technology-high-resolution-images/. (2022).

[25] Sunghoon Im, Hyowon Ha, Gyeongmin Choe, Hae-Gon Jeon, Kyungdon Joo, and In So Kweon. 2015. High quality structure from small motion for rolling shutter cameras. In *Proceedings of the IEEE International Conference on Computer Vision*. 837–845.

[26] Fabrizio La Rosa, Maria Celvisia Virzì, Filippo Bonaccorso, and Marco Branciforte. 2015. Optical Image Stabilization (OIS). *STMicroelectronics. Available online: http://www. st. com/resource/en/white_paper/ois_white_paper. pdf* (2015).

[27] Rushi Lan, Long Sun, Zhenbing Liu, Huimin Lu, Cheng Pang, and Xiaonan Luo. 2020. Madnet: A fast and lightweight network for single-image super resolution. *IEEE transactions on cybernetics* 51, 3 (2020), 1443–1453.

[28] Chen Liu, Jimei Yang, Duygu Ceylan, Ersin Yumer, and Yasutaka Furukawa. 2018. Planenet: Piece-wise planar reconstruction from a single rgb image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2579–2588.

[29] Fayao Liu, Chunhua Shen, Guosheng Lin, and Ian Reid. 2015. Learning depth from single monocular images using deep convolutional neural fields. *IEEE transactions on pattern analysis and machine intelligence* 38, 10 (2015), 2024–2039.

[30] Bruce D Lucas, Takeo Kanade, et al. 1981. An iterative image registration technique with an application to stereo vision. In *DARPA Image Understanding Workshop*. Vancouver, British Columbia.

[31] MathWorks. 2022. Single Camera Calibrator App. https://www.mathworks.com/help/vision/ug/single-camera-calibrator-app.html. (2022).

[32] Yan Michalevsky, Dan Boneh, and Gabi Nakibly. 2014. Gyrophone: Recognizing speech from gyroscope signals. In *23rd {USENIX} Security Symposium ({USENIX} Security 14)*. 1053–1067.

[33] Hao Pan, Yi-Chao Chen, Qi Ye, and Guangtao Xue. 2021. Magicinput: Training-free multi-lingual finger input system using data augmentation based on mnists. In *Proceedings of the 20th International Conference on Information Processing in Sensor Networks (co-located with CPS-IoT Week 2021)*. 119–131.

[34] Alex Perekalin. 2018. Why face unlock is a bad idea. https://www.kaspersky.com/blog/face-unlock-insecurity/21618/. (2018).

[35] Raytrix. 2022. 3D light field camera technology. https://raytrix.de/. (2022).

[36] RICOH. 2022. PENTAX Star Photography. https://www.pentax.com.tw/index.php?do=share&act=info&pid=0&id=83. (2022).

[37] Hamed Sarbolandi, Damien Lefloch, and Andreas Kolb. 2015. Kinect range sensing: Structured-light versus Time-of-Flight Kinect. *Computer vision and image understanding* 139 (2015), 1–20.

[38] Johannes L Schonberger and Jan-Michael Frahm. 2016. Structure-from-motion revisited. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4104–4113.

[39] Steven M Seitz, Brian Curless, James Diebel, Daniel Scharstein, and Richard Szeliski. 2006. A comparison and evaluation of multi-view stereo reconstruction algorithms. In *2006 IEEE computer society conference on computer vision and pattern recognition (CVPR'06)*, Vol. 1. IEEE, 519–528.

[40] Yunmok Son, Hocheol Shin, Dongkwan Kim, Youngseok Park, Juhwan Noh, Kibum Choi, Jungwoo Choi, and Yongdae Kim. 2015. Rocking drones with intentional sound noise on gyroscopic sensors. In *24th {USENIX} Security Symposium ({USENIX} Security 15)*. 881–896.

[41] Peter Sturm and Bill Triggs. 1996. A factorization based algorithm for multi-image projective structure and motion. In *European conference on computer vision*. Springer, 709–720.

[42] Zachary Teed and Jia Deng. 2018. Deepv2d: Video to depth with differentiable structure from motion. *arXiv preprint arXiv:1812.04605* (2018).

[43] Carlo Tomasi and Takeo Kanade. 1991. Detection and tracking of point. *Int J Comput Vis* 9 (1991), 137–154.

[44] Bill Triggs. 1996. Factorization methods for projective structure and motion. In *Proceedings CVPR IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 845–851.

[45] Bill Triggs, Philip F McLauchlan, Richard I Hartley, and Andrew W Fitzgibbon. 1999. Bundle adjustment a modern synthesis. In *International workshop on vision algorithms*. Springer, 298–372.

[46] Timothy Trippel, Ofir Weisse, Wenyuan Xu, Peter Honeyman, and Kevin Fu. 2017. WALNUT: Waging doubt on the integrity of MEMS accelerometers with acoustic injection attacks. In *2017 IEEE European symposium on security and privacy (EuroS&P)*. IEEE, 3–18.

[47] Yazhou Tu, Zhiqiang Lin, Insup Lee, and Xiali Hei. 2018. Injected and delivered: Fabricating implicit control over actuation systems by spoofing inertial sensors. In *27th {USENIX} Security Symposium ({USENIX} Security 18)*. 1545–1562.

[48] Lucía Vera, Jesús Gimeno, Inmaculada Coma, and Marcos Fernández. 2011. Augmented mirror: interactive augmented reality system based on kinect. In *IFIP Conference on Human-Computer Interaction*. Springer, 483–486.

[49] Open Source Computer Vision. 2022. Depth Map from Stereo Images. https://docs.opencv.org/3.4/dd/d53/tutorial_py_depthmap.html. (2022).

[50] 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. 2018. Synthetic depth-of-field with a single-camera mobile phone. *ACM Transactions on Graphics (ToG)* 37, 4 (2018), 1–13.

[51] Jeremy H-S Wang, Kang-Fu Qiu, and Paul C-P Chao. 2017. Control design and digital implementation of a fast 2-degree-of-freedom translational optical image stabilizer for image sensors in mobile camera phones. *Sensors* 17, 10 (2017), 2333.

[52] Xingkui Wei, Yinda Zhang, Zhuwen Li, Yanwei Fu, and Xiangyang Xue. 2020. Deepsfm: Structure from motion via deep bundle adjustment. In *European conference on computer vision*. Springer, 230–247.

[53] Diana Wofk, Fangchang Ma, Tien-Ju Yang, Sertac Karaman, and Vivienne Sze. 2019. Fastdepth: Fast monocular depth estimation on embedded systems. In *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 6101–6108.

[54] Jiangjian Xiao, Hui Cheng, Feng Han, and Harpreet Sawhney. 2008. Geo-spatial aerial video processing for scene understanding and object tracking. In *2008 IEEE*

Conference on Computer Vision and Pattern Recognition. IEEE, 1–8.

[55] Yao Yao, Zixin Luo, Shiwei Li, Tian Fang, and Long Quan. 2018. Mvsnet: Depth inference for unstructured multi-view stereo. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 767–783.

[56] SEEED Yida. 2019. What is a Time of Flight Sensor and How does a ToF Sensor work? https://www.seeedstudio.com/blog/2020/01/08/what-is-a-time-of-flight-sensor-and-how-does-a-tof-sensor-work/. (2019).

[57] Fisher Yu and David Gallup. 2014. 3d reconstruction from accidental motion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3986–3993.

[58] Sangki Yun, Yi-Chao Chen, Huihuang Zheng, Lili Qiu, and Wenguang Mao. 2017. Strata: Fine-grained acoustic-based device-free tracking. In *Proceedings of the 15th annual international conference on mobile systems, applications, and services*. 15–28.

[59] Yongzhao Zhang, Wei-Hsiang Huang, Chih-Yun Yang, Wen-Ping Wang, Yi-Chao Chen, Chuang-Wen You, Da-Yuan Huang, Guangtao Xue, and Jiadi Yu. 2020. Endophasia: utilizing acoustic-based imaging for issuing contact-free silent speech commands. *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies* 4, 1 (2020), 1–26.