



# Improving the localisation of features for the calibration of cameras using EfficientNets

JOE EASTWOOD,<sup>1,†,\*</sup>  GEORGE GAYTON,<sup>1,†</sup> RICHARD K LEACH,<sup>1,2</sup>  
AND SAMANTA PIANO<sup>1</sup>

<sup>1</sup>Manufacturing Metrology Team, Faculty of Engineering, University of Nottingham, UK

<sup>2</sup>Taraz Metrology Ltd., UK

<sup>†</sup>These authors contributed equally to this work

\*[JoeE@stwood.co.uk](mailto:JoeE@stwood.co.uk)

**Abstract:** Camera-based methods for optical coordinate metrology, such as digital fringe projection, rely on accurate calibration of the cameras in the system. Camera calibration is the process of determining the intrinsic and distortion parameters which define the camera model and relies on the localisation of targets (in this case, circular dots) within a set of calibration images. Localising these features with sub-pixel accuracy is key to providing high quality calibration results which in turn allows for high quality measurement results. A popular solution to the localisation of calibration features is provided in the OpenCV library. In this paper, we adopt a hybrid machine learning approach where an initial localisation is given by OpenCV which is then refined through a convolutional neural network based on the EfficientNet architecture. Our proposed localisation method is then compared with the OpenCV locations without refinement, and to an alternative refinement method based on traditional image processing. We show that under ideal imaging conditions, both refinement methods provide a reduction in the mean residual reprojection error of approximately 50%. However, in adverse imaging conditions, with high noise levels and specular reflection, we show that the traditional refinement degrades the results given by pure OpenCV, increasing the mean residual magnitude by 34%, which corresponds to 0.2 pixels. In contrast, the EfficientNet refinement is shown to be robust to the unideal conditions and is still able to reduce the mean residual magnitude by 50% compared to OpenCV. The EfficientNet feature localisation refinement, therefore, enables a greater range of viable imaging positions across the measurement volume. leading to more robust camera parameter estimations.

Published by Optica Publishing Group under the terms of the [Creative Commons Attribution 4.0 License](https://creativecommons.org/licenses/by/4.0/). Further distribution of this work must maintain attribution to the author(s) and the published article's title, journal citation, and DOI.

## 1. Introduction

Optical coordinate measurement systems (CMSs) are growing in popularity for the inspection of manufactured parts [1–5]. This is due to their non-contact probing nature, relatively fast data acquisition times, high surface coverage and high point density. Additionally, optical CMSs can measure the complex freeform surfaces enabled by additive manufacturing (AM), which may otherwise be difficult to measure. Digital fringe projection (DFP) is a particularly popular measuring technique; its fast surface reconstruction times allow it to be capable of providing in-situ measurements of AM surfaces [6,7].

DFP systems make surface measurements by projecting a sinusoidal pattern onto the surface of interest, the surface deforms this pattern and the deformed pattern is detected by a camera. The height surface data can then be decoded using the phase information in the received pattern. Typically, the optical devices in a DFP system require a calibration process, whereby a set of quantities that define the camera and projector models are determined. A popular calibration method for cameras is presented by Zhang et al. [8] (as a projector is optically identical to a camera, projectors can also be calibrated with this method). In this paper, a target with

known features is imaged repeatedly in a range of positions and angles across the measurement volume. Within each camera image, the target features are localised. Because the distribution of the features on the target is known, the camera and projector parameters can be estimated using the pixel coordinates of the detected features within each image. A global minimisation is performed over the estimated camera parameters to minimise the reprojection error of the detected points. Popular calibration target choices include the checkerboard [8] and the dot grid [9], though these are not the only options [10]. Dot grids are particularly popular for DFP system calibration compared to checkerboards as it is difficult to extract reliable phase information at the checkerboard corners due to a lack of contrast, and the calibration procedure benefits from the symmetry of the dot features [11].

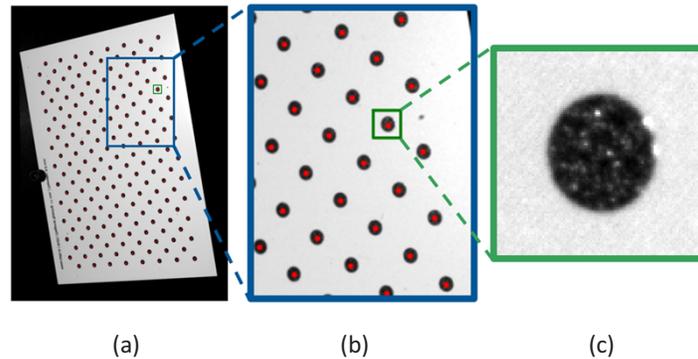
The accuracy of the camera calibration is based on the accuracy of the known artefact points and the accuracy of the localised camera image points. Methods exist to improve the localisation of the camera image points to improve the calibration, such as the line-spread function approach [12] defined in Section 2. However, the refinement of camera image points can be difficult given sub-optimal measurement conditions, which exist to some extent in almost every calibration [13]. Machine learning (ML) provides a potential method to refine the camera image points, despite sub-optimal measurement conditions, due to the ability of ML models to learn to be robust to adverse conditions such as high signal to noise ratios [14].

### 1.1. Previous work

Machine learning methods to augment camera calibration have been the subject of previous studies [15–22]. Early works used a genetic algorithm to globally optimise the camera parameters, but this method was shown to have little benefit over traditional approaches [15]. More recently, some researchers have attempted to replace the entire calibration process with an end-to-end machine learned model [16]. Mohamed et al. [17] explicitly obtained the camera projection matrix through a support vector machine and showed this approach to be more robust to noise and more computationally efficient than traditional techniques. He et al. [18] used a K-singular value decomposition sparse dictionary learning approach to perform a non-linear optimisation of the camera parameters; they claim that, once trained, this approach can enable single image calibration. Other studies [14,20–22], instead, implemented a hybrid pipeline which fuses machine learning techniques with the traditional calibration pipeline proposed by Zhang [19]. The calibration target detection and target feature localisation, specifically, can be improved through machine learning, as traditional methods can be highly influenced by factors such as noise [20,21]. For example, the machine learning for adaptive calibration template detection (MATE) model proposed by Donné et al. [14] is a convolutional neural network trained to be robust to noisy inputs and high levels of lens distortion. Traditional calibration methods require that the entire calibration target is visible in every image in the calibration dataset. A model developed by Chen et al. [22] was designed specifically to be robust to views in which some portion of the calibration target cannot be seen by the camera.

In this paper, a hybrid approach (referred to hereon as the ML method) is adopted where an initial estimate of the feature locations is provided by traditional methods, then refined through a learned model. The calibration target used is comprised of black dot features on a white background and can be seen in Fig. 1(a). This target is chosen as it provides a large degree of phase information when calibrating fringe projection systems. Our proposed approach first takes an initial estimate of the location of each dot feature as given by OpenCV [23]. A set of new images is then created from a  $(101 \times 101)$  pixel bounding box around each feature, such that each sub-image contains a single dot with the OpenCV centre location of that feature at the centre pixel of the new cropped image. Figure 1. shows this process.

Each sub-image is passed to a model based on the EfficientNet architecture [24] which produces a predicted sub-pixel correction to the OpenCV centre location. The EfficientNet is trained



**Fig. 1.** An example image of the calibration target used in this paper. (a) The full image, (b) a zoomed image showing the OpenCV feature detection locations in red, (c) an example of the cropped sub-images formed around each detected feature.

on synthetic data in which the ground truth centre is known implicitly, the generation of this training data is presented in Section 2. Once trained, the EfficientNet model is inserted into the calibration pipeline. We can then evaluate the proposed calibration pipeline against real data. First, the ML method is compared with results using the purely OpenCV (OCV method) and shown to provide significant reductions in the reprojection error. Secondly, the ML method is compared to an alternative refinement approach, using traditional image processing, based on the line-spread function (LSF method), which is described in Section 3. We show that the ML method performs comparably to the LSF method in ideal conditions, but the ML method is more robust than other methods in the case of adverse imaging conditions, such as noise and the presence of speckles caused by specular reflection. This improved robustness allows the calibration image set to contain a wider range of views across the measurement volume when the hybrid pipeline is used and, as such, allows improved calibration results over the LSF method.

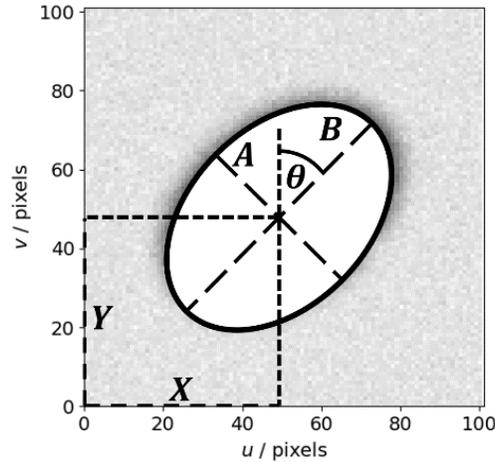
## 2. Dataset creation

As was shown in Fig. 1.(c), when the ML model is deployed, it will operate on sub-images of a single feature, rather than the full calibration image. Therefore, a labelled training dataset of these sub-images is required; this dataset is built by generating a large set of synthetic ellipse images. Each virtual ellipse feature used in the training data is created using a set of parameters, given by:

1. Ellipse position  $X$
2. Ellipse position  $Y$
3. Ellipse semi-major axis  $A$
4. Ellipse semi-minor axis  $B$
5. Ellipse rotation  $\theta$
6. Internal pixel distribution
7. External pixel distribution
8. Blurring kernel width
9. Specular size

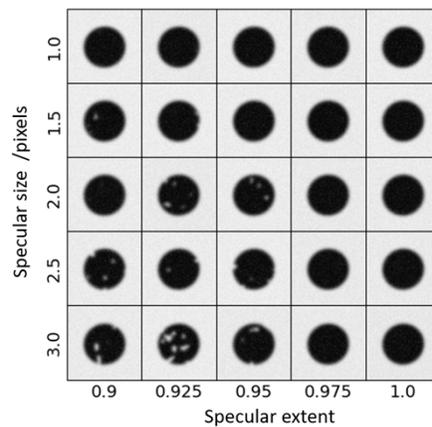
## 10. Specular extent

The ellipse parameters ( $X, Y, A, B, \theta$ ) explicitly define the feature shape itself; these parameters are visualised in Fig. 2.



**Fig. 2.** Ellipse geometry parameters.

The internal distribution defines the distribution of intensity values inside the feature, while the external distribution defines the distribution of intensity values outside the feature. Both the internal and external pixel distribution are taken to be log-normal distributions [25]. The blurring kernel defines the blur of the image of the feature. Finally, the specular size and specular extent determine the internal pixel values that do not typically conform to the internal pixel distribution because of non-Lambertian reflections within the ellipse. Specular size represents the size of each specular feature in pixels, while specular extent represents the percentage of internal pixels which do not constitute specular artefacts. Figure 3. shows the effect of the specular parameters on an example simulated image.



**Fig. 3.** Effect of changing the specular parameters on randomly sampled ellipses with all other parameters set to be constant.

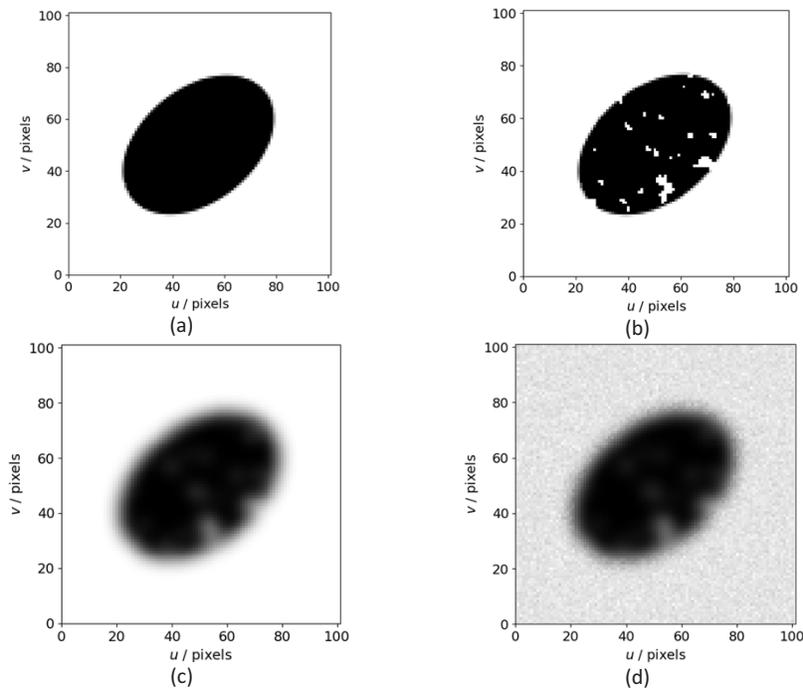
A range of images of real calibration features and measurements of the ellipse parameters were conducted and the distributions of these parameter values estimated via kernel density estimation

(KDE). These probability density functions (PDFs) were then randomly sampled to generate each image in the simulated dataset. The PDFs used to generate some key parameters (ellipse centre and specular parameters) were set manually to exceed the values determined by KDE such that the model was trained to handle outliers. Table 1. summarises how each parameter distribution was set.

**Table 1. Parameter distributions used when creating the simulated dataset.**

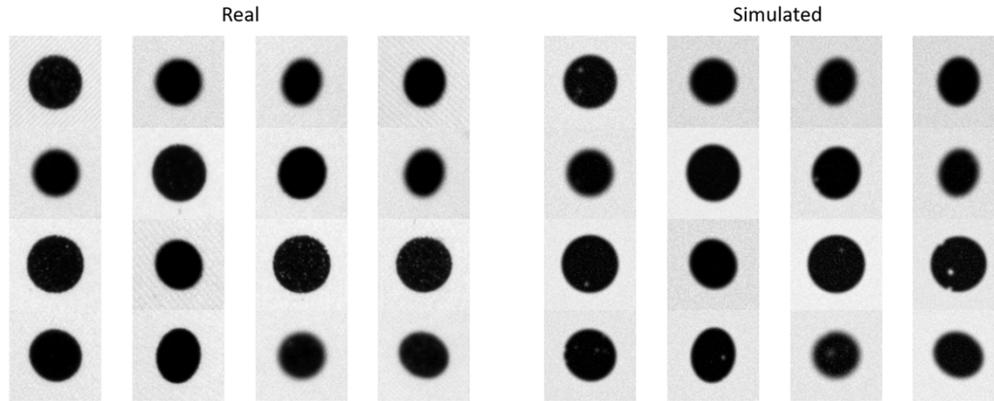
Parameter	Distribution
Ellipse Centre ( $X, Y$ )	Gaussian, $\mu = 50$ , $\sigma = 0.1$
Ellipse axes ( $A, B$ )	Determined via KDE on the real image dataset.
Ellipse rotation ( $\theta$ )	Determined via KDE on the real image dataset
Internal/External pixel distributions	Determined via KDE on the real image dataset
Blurring kernel width	Determined via KDE on the real image dataset
Specular size	A uniform distribution in the range 1–3
Specular extent	A uniform distribution between 0.9–1.0

The creation of an ellipse is shown in Fig. 4. First, in 4.(a), the parameters ( $X, Y, A, B, \theta$ ) are used to generate a rasterised ellipse comprised of pixel values between 0 and 1. The ellipse is then renormalised to the correct contrast and offset. Then, in 4.(b), sub-optimal reflections are added to the ellipse as a series of random white pixel blobs as determined by the specular parameters and in 4.(c), the ellipse is blurred using a Gaussian kernel with width set by sampling the blurring kernel PDF. Finally, in 4.(d), ellipse-specific noise is added to the internal and external portions of the ellipse through random sampling of the internal and external log-normal pixel distributions, with any pixels exceeding the maximum 10-bit value (1023) of the camera being reset to 1023.



**Fig. 4.** Ellipse creation method, (a) rasterised ellipse, (b) addition of sub-optimal reflections, (c) blurring and (d) addition of noise.

Figure 5. shows a comparison between real and synthetic sub-images of features. It can be seen that the synthetic images are qualitatively similar to the real data. We verify that the synthetic data must be a sufficient representation of the real data in Section 5.2 when the EfficientNet, trained on the synthetic data, is tested on real images and shown to produce high quality calibration results.



**Fig. 5.** Comparison of 16 real and 16 simulated calibration dots randomly sampled from each dataset.

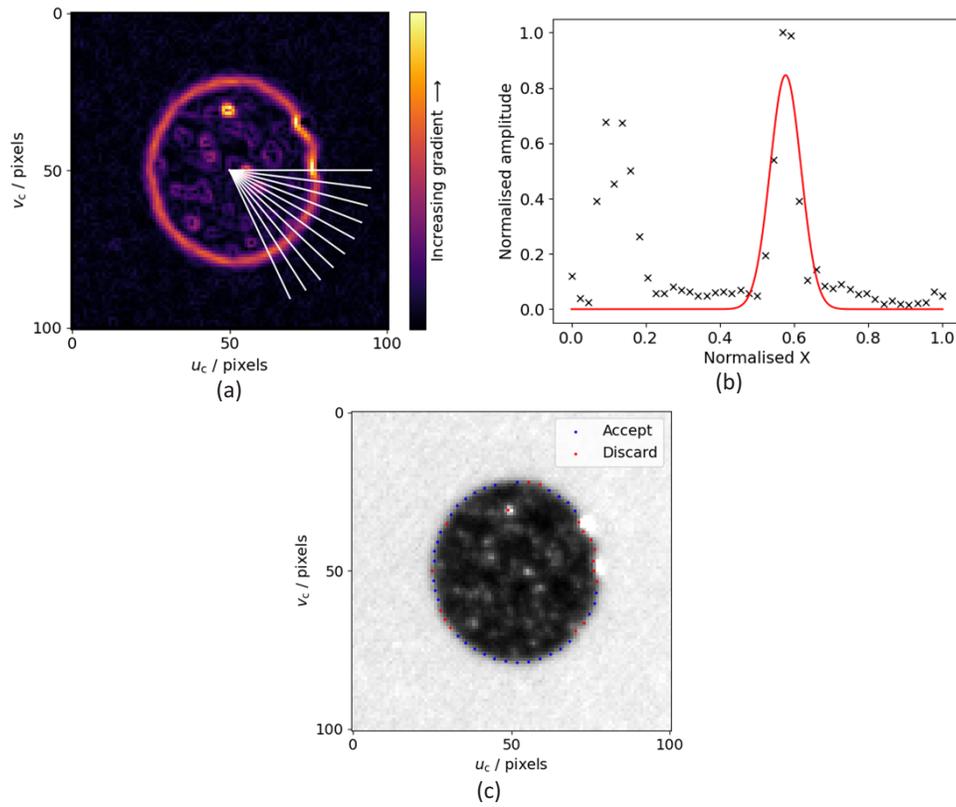
Using the approach described above, a training set of 10 000 synthetic calibration features was created and a further 1000 were saved for testing. During training 10% of the images were randomly sampled from the total dataset and used for validation, the remaining images were used for training. The validation dataset was identical when training all models as presented in Section 5.

### 3. Line-spread function approach

A common method to find ellipse centres is to fit an ellipse to edge points estimated from the largest gradients in the image. For robustness, this can be done along interpolated 1D lines from an estimated centre, where each line is called a line-spread function (LSF). This is, therefore, called the LSF method. First, a gradient image of the region containing the ellipse, shown in Fig. 6.(a), is found by convolving the region with a Sobel kernel [26]. A series of line-spread functions are taken of the gradient image that expand radially from the estimated centre of the ellipse - it is assumed that the initial ellipse centre estimation is within  $\pm 1$  pixel. The line-spread function is interpolated from the gradient image, using a bilinear interpolation, shown in Fig. 6.(b). In Fig. 6.(b), a Gaussian function is fit to all line-spread functions to estimate the centre of the peak that corresponds to the ellipse boundary. Erroneous peak estimations are filtered out using a random sample consensus (RANSAC) algorithm [27] and the result is shown in Fig. 6.(c).

In Fig. 6(c), there are some over-exposed regions of the image – the boundary estimations here do not correspond well with the real ellipse boundary. These erroneous boundary estimation points can have a significant effect on the ellipse fitting result and, in cases when there are many over-exposed regions lying on the ellipse boundary, can cause the ellipse fitting to fail. The EfficientNet based approach given in Section 4 is designed to be more robust to both noise and specular regions.

Once the dot centres have been localised and refined, an extended version of the calibration procedure presented in Ref. [8] is used. The calibration results in a final camera model parameterised by intrinsic parameters  $[f_x, f_y, u_0, v_0, s]$ , where  $(f_x, f_y)$  are the focal lengths,  $(u_0, v_0)$  are the principle point offsets and  $s$  is the skewness; and distortion parameters



**Fig. 6.** Line-spread function approach to ellipse centre localisation. (a) Series of cross-sectional lines over the gradient image. (b) Line-spread function of a line from (a). (c) Estimated ellipse edge points, with erroneous points filtered using the RANSAC algorithm.

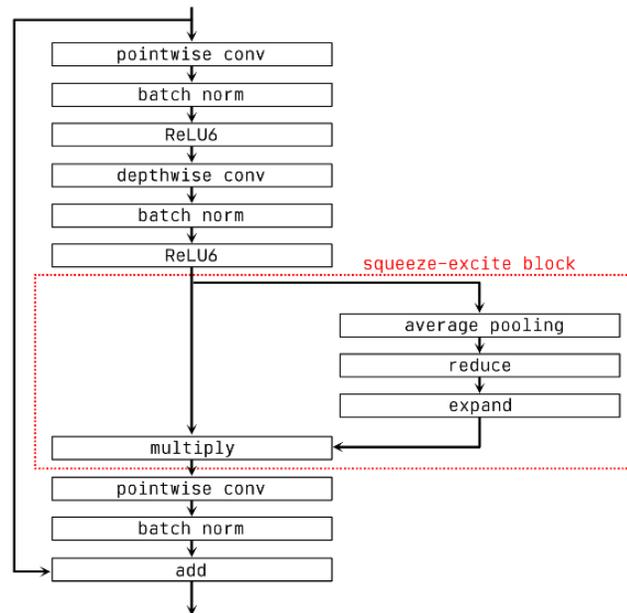
$[k_1, k_2, k_3, p_1, p_2, u_{dc}, v_{dc}]$ , where  $(k_1, k_2, k_3)$  are radial distortion coefficients,  $(p_1, p_2)$  are tangential distortion coefficients, and  $(u_{dc}, v_{dc})$  are the distortion centre coordinates.

#### 4. Machine learning approach

The ML architecture used in this paper is based on the EfficientNet family of models. This architecture was chosen due to EfficientNet based models performing well (ranked top three at the time of writing) on the benchmark ImageNet dataset, while having fewer trainable parameters than other architectures [24].

The building block of an EfficientNet is the MBConv layer which is based on the MobileNet family [28]. MBConv blocks can be summarised as inverted residual linear bottleneck blocks with depthwise separable convolution and squeeze-excite blocks, Fig. 7. shows the layers in an MBConv block.

First, the number of channels in the input is increased through a pointwise convolution. A depthwise-separable convolution is applied which consists of a combination of a depthwise convolution followed by a pointwise convolution. Depthwise separable convolution requires fewer parameters, and thus fewer computations during prediction, than a simple 2D convolutional layer. A squeeze excite block is inserted in the centre of the depthwise separable convolution which essentially learns a weighting to apply to each channel of the input before the pointwise



**Fig. 7.** MBConv block detail.

convolution is applied. The squeeze-excite block was first presented by Hu et al. [29] and shown to lead to improved model predictions. Finally, the output of the convolution is combined with the output of the previous block, which is fed forward through a skip connection, similar to that used in a ResNet [30]. This skip connection creates an alternative path for the error gradient to flow during back propagation which mitigates against the vanishing gradient problem [31].

From the MBConv, a family of networks was presented in two papers by Tan and Le [24,32] called EfficientNets and EfficientNetsV2 respectively. This family of models is created by stacking varying numbers of MBConv blocks together. In the case of EfficientNetV2, the early layers of the model eschew depthwise convolution as it was shown to be more computationally efficient, despite the increase in parameters compared to using depthwise separable convolution in the entire model. From these two model families (EfficientNets and EfficientNetV2s), the nine models summarised in Table 1. were selected for evaluation against the ellipse dataset.

The models summarised in Table 2, which were originally designed for classification, were modified with two linear output nodes used to regress the sub-pixel correction. No transfer learning was employed and each model was initialised with randomised parameter values. Each model was optimised using the Adam optimiser [33] and a LogCosh loss function was used to improve robustness against outliers. The dataset was split into training and validation sets with 10% of the data selected for validation as was described at the end of Section 2. Training was conducted for 100 hours or 1000 Epochs, whichever occurred first. After training, the model weights were restored to the epoch of the lowest mean absolute error as evaluated on the validation dataset. Training was conducted in parallel on the Augusta high performance cluster (HPC) nodes with each process assigned 16 CPU cores and 128 GB of RAM. Training time varied on model complexity with the smallest B0 model taking an average of 632 ms per 64 image batch, and the largest V2L model taking 6 s per batch.

Once trained, the model can be used to refine the centre predictions given by OpenCV and these centre locations are then used in the same calibration procedure that was outlined in Section 3.

**Table 2. EfficientNet models evaluated.**

Model	Size (MB)	Parameters (millions)
B0	29	5.3
B4	75	19.5
B5	118	30.6
B6	166	43.3
B7	256	66.7
V2B0	29	7.2
V2S	88	21.6
V2M	220	54.4
V2L	479	119

## 5. Results

### 5.1. Model training results

Table 3. shows the performance of each model evaluated against the test set once the optimised performing parameter values have been restored. The improvement metric quantifies the change in mean absolute error (MAE) between the model prediction and the assumption that the ellipse centre is directly in the middle of the sub-image.

**Table 3. EfficientNet models evaluated.**

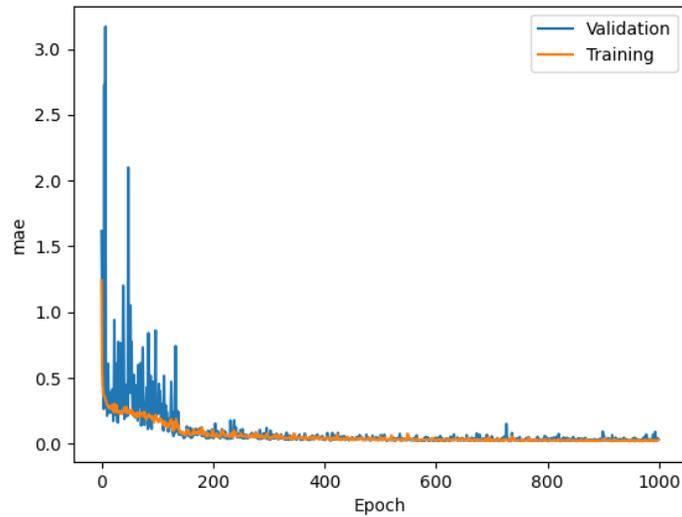
Model	Total training time (hrs)	Metric			
		MAE (px)	MAPE (%)	Improvement (px)	Test Loss
B0	16.5	0.024	6.2	21.84	9.33E-04
B4	58.9	0.021	5.7	21.90	6.51E-04
<b>B5</b>	<b>94.3</b>	<b>0.018</b>	<b>5.2</b>	<b>21.92</b>	<b>6.15E-04</b>
B6	100	0.019	5.3	21.92	7.05E-04
B7	100	0.027	6.2	21.81	1.30E-03
V2B0	22.0	0.026	6.3	21.83	1.10E-03
V2S	68.2	0.021	6.4	21.92	8.11E-04
V2M	100	0.021	6.9	21.95	7.63E-04
V2L	100	0.040	8.4	21.51	2.66E-03

As is clear from Table 3, EfficientNetB5 was the highest performing model in this test, with a mean absolute error of 0.018 pixels, which translates to a mean percentage error of 5.2%. As both the smaller B4 model and the larger B6 model had higher test MAEs, B5 was taken to be the optimal EfficientNet model size for this application. Figure 8. shows how the mean absolute error evolved over the training period.

As can be seen in Fig. 8, the training result converged relatively quickly and stably. The minimum validation mean absolute error was 0.0183 pixels and occurred at epoch 992, therefore, the model weights were restored to this point before the model was deployed into the calibration pipeline.

### 5.2. Results on real data

The performance of both the LSF and ML method were compared by taking their corresponding dot locations used to calibrate a camera – the Prosilica GT 5120 with an attached lens (soligor 35 mm f/2.8). The two dot localisation methods were also compared against the OCV method



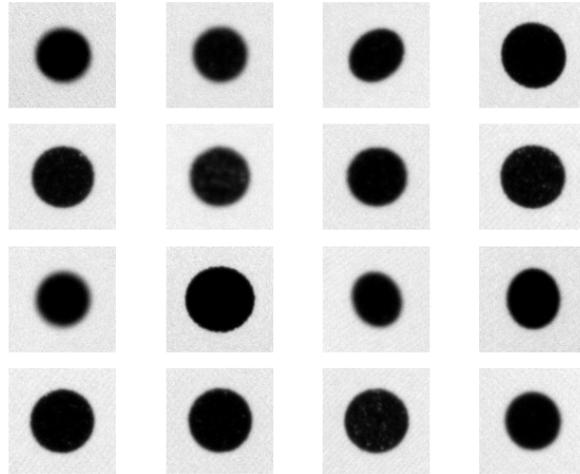
**Fig. 8.** EfficientNetB5 mean absolute error evolution.

which uses only the function `findCirclesGrid` from OpenCV 4.5.5 without further correction. The difference between the feature location as predicted by the given method (LSF, OCV or ML) and the same feature location when reprojected through the camera model back to the imaging plane is called the residual and this residual is minimised during the calibration using the Levenberg-Marquardt algorithm [34]. The final residual value is derived from the combination of errors in the dot grid artefact, errors in the dot locations and possible differences in local against global minima [35] in the calibration. Assuming the errors caused by the manufacture of the dot grid artefact to be stable, and assuming the Levenberg-Marquardt algorithm has converged to the global minimum, the final residual value can be considered a direct evaluation of the accuracy of the feature localisation method. The residual of the  $i^{\text{th}}$  point in the calibration is given by  $(\Delta x_i, \Delta y_i)$ , where the calibration uses  $N$  points, the dot localisation accuracy of each method has been compared using the mean magnitude  $R$ ,

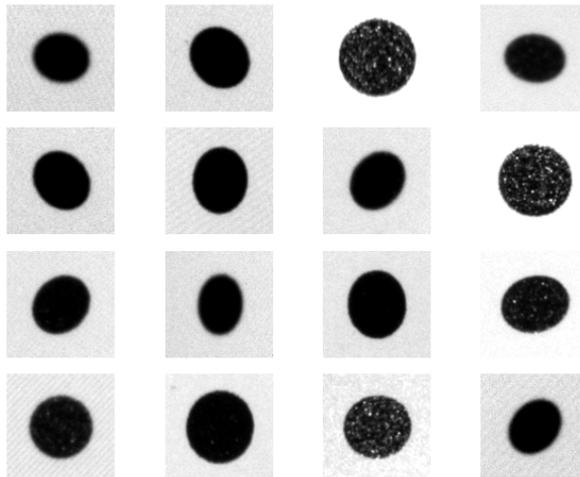
$$R = \frac{1}{N} \sum_{i=1}^N \sqrt{\Delta x_i^2 + \Delta y_i^2}. \quad (1)$$

The LSF, ML and OCV methods were tested using two distinct calibration datasets: a cooperative dataset and an uncooperative dataset. In the cooperative dataset, the calibration data has been taken by minimising specular reflection components – completed by providing feedback to the operator during calibration when there was excessive saturation of pixels. Pixel saturation was identified using an image of the dot grid under a projected image comprised of only pixel value 255. Pixels in the camera image that were at maximum pixel value were classified as saturated. In the uncooperative dataset, there is no limit on the position and orientation of the dot grid, and so some positions will be outside the nominal operating ranges of the LSF method. The cooperative dataset contains images of the calibration target from 18 positions while the uncooperative dataset contains images from 22 positions. The calibration target contains 184 circular features leading to dataset sizes of 3312 ellipse images for the cooperative datasets and 4048 ellipse images in the uncooperative dataset. Figure 9. and Fig. 10. show a number of examples from the cooperative and uncooperative datasets respectively.

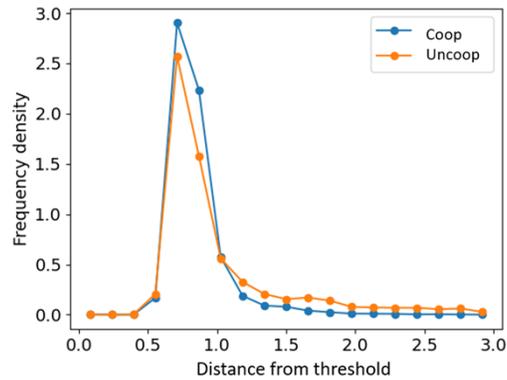
Figure 11. shows the internal pixel distributions in each dataset as quantified by the distance from a “normal” value as determined using Otsu’s method [36].



**Fig. 9.** Example calibration target images included in the cooperative dataset



**Fig. 10.** Example calibration target images included in the uncooperative dataset. As can be seen, the extent of the specular reflections and the range of imaging angles are extended beyond what can be seen in the cooperative examples shown in Fig. 8.



**Fig. 11.** Pixel distributions internal to each ellipse from the cooperative and uncooperative datasets as a distance from a threshold determined by Otsu's method [36].

As can be seen in Fig. 11, the uncooperative dataset contains many more outlying pixels which represents an increased rate of specular artefacts. These specular artefacts are mainly caused when imaging at high angles relative to the imaging plane which were excluded from the cooperative dataset due to the limiting of imaging positions which produce saturated pixels as described previously.

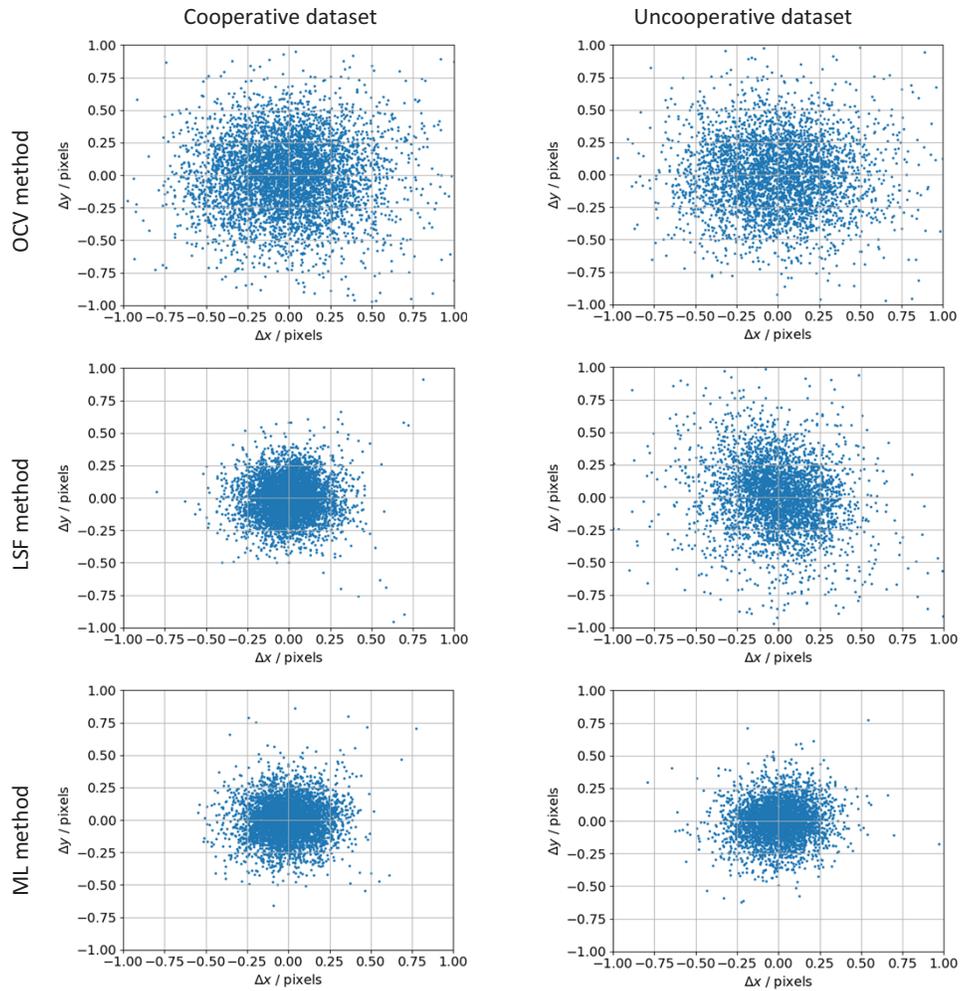
Figure 12 shows the residual values for each method evaluated on both datasets. The mean magnitude residuals are given in Table 4.

**Table 4. Mean residual magnitude.**

Method	Cooperative dataset (pixels)	Uncooperative dataset (pixels)
OCV	0.37	0.38
LSF	0.19	0.51
ML	0.18	0.19

Using the values provided in Table 4, the benefit of using the two refinement methods can be quantified. In the case of the cooperative dataset, it is clear that both the LSF method and ML method provide a considerable reduction in reprojection error. The percentage reduction in the mean residual magnitude is 49% in the case of the LSF method and 51% in the case of the ML method. However, when the dataset is uncooperative, the LSF method in fact degrades the performance of the calibration and the mean magnitude residual increases by 34%. In contrast, the ML method can still provide a reduction in the mean residual magnitude of 50%. Table 5 summarises the effect on parameter estimation of each method on every dataset.

It is hard to draw any direct conclusions from the estimated parameters shown in Table 5, due to the lack of any ground truth camera parameters and as such they are included here only for completeness. However, the greater performance of the ML estimated parameters as shown in Table 4 implies that the ML estimated parameters are likely to be closer to the true values than the LSF and OCV estimations.



**Fig. 12.** Distribution of residual errors in the reprojection of features for each calibration method for each data set

Table 5. Estimated parameters from each dataset.

Parameters	LSF		ML		OCV		
	Coop	Uncoop	Coop	Uncoop	Coop	Uncoop	
Intrinsic parameters / pixels	$f_x$	8521	8547	8520	8533	8516	8545
	$f_y$	8521	8553	8521	8534	8516	8552
	$s$	-0.22	-0.12	-0.23	-0.91	-0.58	-1.17
	$u_0$	2697	2627	2697	2656	2699	2626
	$v_0$	2553	2568	2553	2552	2554	2567
Distortion coefficients / AU	$k_1$	0.1199	0.1135	0.1374	-0.0388	0.1144	0.1182
	$k_2$	0.0864	0.0579	0.0555	0.2145	0.0616	0.1244
	$k_3$	0.0991	0.0910	0.1422	-0.6019	0.1292	0.1302
	$p_1$	0.0838	0.1015	0.0698	-0.0013	0.0667	0.1265
	$p_2$	0.1317	0.1142	0.0990	0.0001	0.1403	0.0790
	$u_{dc}$	0.1351	0.0955	0.0642	-0.0145	0.1453	0.1402
	$v_{dc}$	0.0665	0.0641	0.1130	0.0155	0.1468	0.0824

## 6. Discussion

In this paper, the sub-images were sampled from the captured calibration images at a scale of  $101 \times 101$  pixels. This size was chosen as all imaging positions useful for the calibration task produced features which fit within this size. If a different choice of calibration target or camera were made then this size may need to be adjusted.

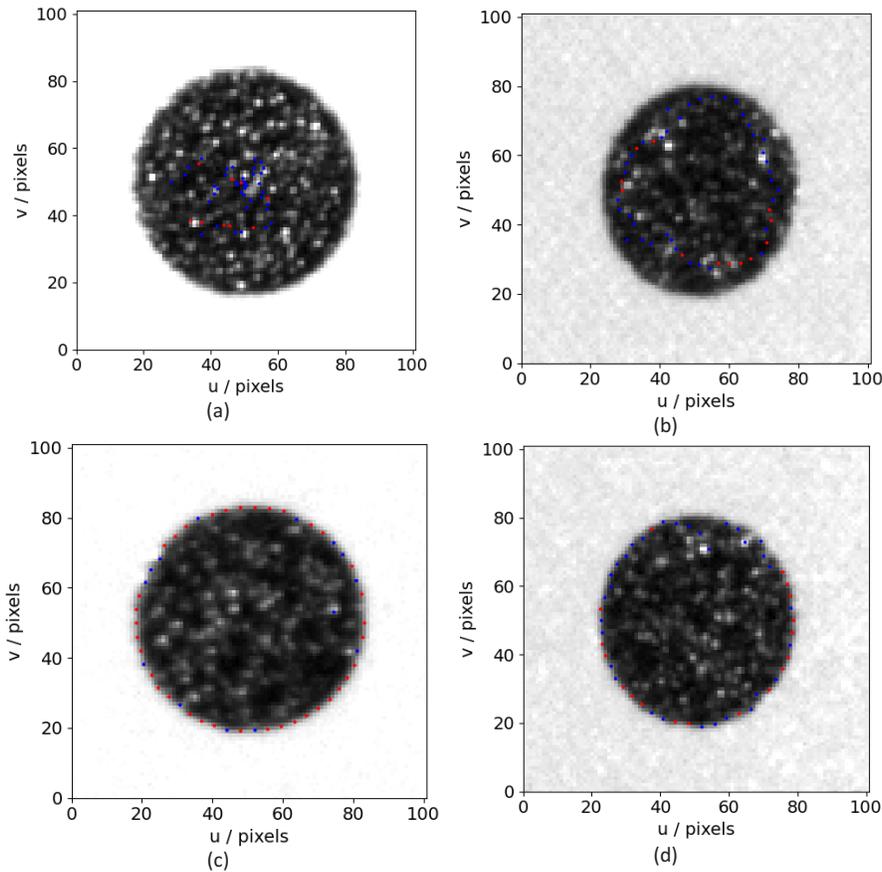
As was summarised in Table 4, the ML method can improve the feature localisation with both cooperative and uncooperative datasets. In comparison, the LSF method can reduce the localisation accuracy as evaluated by the mean residual magnitude by 34%. The decrease in localisation accuracy is, in part, due to the fact that the LSF method was unable to make reasonable estimations of all the features in the uncooperative dataset. Figure 13. shows some example failure cases of the LSF method.

The failure of the LSF method to fit an ellipse to the boundary can be seen in Fig. 13.(a) and Fig. 13.(b); there has been no reasonable estimation of the boundary points. However, the LSF method does not always fail in these conditions – Fig. 13.(c) and Fig. 13.(d) show reasonable approximations under similar conditions. These examples show that the LSF method is unreliable under these measurement conditions.

In comparison, the ML method produced a calibration result of similar quality to that of the cooperative dataset, showing that the desired improvements to robustness have been achieved. This improvement in robustness is visually evident in Fig. 6, where the residual distribution of the ML method is shown to be similar under both calibration conditions. It can also be seen in Fig. 12. that the ML method clearly outperforms the pure OpenCV localisation in both cases, with a reduction in the mean residual of approximately 50%.

The LSF method can be a sound approach, and provides high quality calibration results, but is highly dependent on a cooperative calibration dataset. This is not always possible, particularly in industrial setting where instruments need to be calibrated in-situ by operators who may not be experts. The requirement for a cooperative dataset also limits the number of feasible views which can be acquired as part of the calibration dataset; high-quality calibrations require a high number of views across the measurement volumes at a range of angles relative to the imaging plane. This is evident when considering the impact on parameter estimation shown in Table 5.

Improving robustness allows a greater range of views to be captured and can, therefore, improve the calibration result which will, in turn, improve any measurement results captured by the



**Fig. 13.** Showing some features from the uncooperative dataset, where blue dots have been discarded by the RANSAC algorithm and red dots have been kept as estimated boundary points. (a) and (b) show cases where ellipse fitting failed to produce a good outcome and (c) and (d) show cases where ellipse fitting was succesful despite some outliers.

system. It may be possible to improve the LSF method to be tolerant to a greater range of measurement conditions by fine-tuning hyper-parameters and using alternative filtering methods, but the complexity would outweigh the benefit.

## 7. Results

Two methods for refining the localisation of calibration targets were presented, one based on the line-spread function of the local image gradient (LSF method), and one based on an EfficientNet convolutional neural network (ML method). The two methods were compared to unrefined feature localisation by using two calibration scenarios – a cooperative scenario with minimal overexposures to produce clean ellipses for feature estimate, and an uncooperative scenario with high levels of specular reflection and over-exposure. We show that both refinement approaches lead to a reduction in the mean residual reprojection error magnitude of approximately 50%, with the ML method outperforming the LSF method by 2%. However, in the uncooperative scenario, the use of the LSF method increases the mean residual magnitude by 34%. In contrast, the ML method maintains the 50% reduction in mean residual magnitude. This result shows the EfficientNet has learned to provide localisation refinements which are robust to the adverse

conditions present in the uncooperative calibration image dataset. This improved robustness allows the calibration dataset to include a larger range of imaging positions across the measurement volume, leading to improved parameter estimation and therefore higher quality measurement outcomes.

Future work could investigate the impact of using different characterisation targets and feature shapes on the performance of the EfficientNet position refinement.

**Funding.** Engineering and Physical Sciences Research Council (EP/L016567/1, EP/M008983/1).

**Disclosures.** The authors declare no conflicts of interest.

**Data availability.** Data underlying the results presented in this paper are not publicly available at this time but may be obtained from the authors upon reasonable request.

## References

1. R Kulkarni, E Banoth, and P Pal, "Automated surface feature detection using fringe projection: An autoregressive modeling-based approach," *Opt. Lasers Eng.* **121**, 506–511 (2019).
2. S Ordoñez Nogales, M Servin, M Padilla, I Choque, J L Nuñez, and A Muñoz, "Shape defect measurement by fringe projection profilometry and phase-shifting algorithms," *Opt. Eng.* **59**(01), 1 (2020).
3. R Xia, J Zhao, T Zhang, R Su, Y Chen, and S Fu, "Detection method of manufacturing defects on aircraft surface based on fringe projection," *Optik* **208**, 164332 (2020).
4. C F Cheung, L Kong, and M Ren, "Precision freeform metrology," in R K Leach, eds., *Advances in Optical Form and Coordinate Metrology* (IOP Publishing) (2020).
5. S Catalucci, A Thompson, S Piano, DT Branson, and RK Leach, "Optical metrology for digital manufacturing: a review," *Int. J. Adv. Manuf. Technol.* **120**(7-8), 4271–4290 (2022).
6. B. Zhang, "In Situ Fringe Projection Profilometry for Laser Power Bed Fusion Process," (*PhD thesis*, The University of North Carolina at Charlotte) (2017).
7. N Southon, P Stavroulakis, R Goodridge, and R K Leach, "In-process measurement and monitoring of a polymer laser sintering powder bed with fringe projection," *Mater. Des.* **157**, 227–234 (2018).
8. Z Zhang, "A flexible new technique for camera calibration," *IEEE Trans. Pattern Anal. Mach. Intell.* **22**(11), 1330–1334 (2000).
9. G G Mateos, "A camera calibration technique using targets of circular features," *Proc. SIARP*. (2000).
10. C Schmalz, F Forster, and E Angelopoulou, "Camera calibration: active versus passive targets," *Opt. Eng.* **50**(11), 113601 (2011).
11. S Zhang, "Digital Fringe Projection System Calibration," *High-Speed 3D Imaging with Digital Fringe Projection Techniques* (CRC Press) 103–128 (2018).
12. G Gayton, "Improvements to the characterisation of fringe projection," *PhD Thesis* (University of Nottingham) (2022).
13. S Kopparapu and P Corke, "The effect of noise on camera calibration parameters," *Graph Models* **63**(5), 277–303 (2001).
14. S Donné, J De Vylder, B Goossens, and W Philips, "MATE: Machine learning for adaptive calibration template detection," *Sensors* **16**(11), 1858 (2016).
15. M Roberts and A J Nafel, "A genetic algorithm approach to camera calibration in 3D machine vision," *Proc. IEE CGAIPV* 12 (1994).
16. L Deng, G Lu, Y Shao, M Fei, and H Hu, "A novel camera calibration technique based on differential evolution particle swarm optimization algorithm," *Neurocomputing* **174**, 456–465 (2016).
17. R Mohamed, A Ahmed, A Eid, and A Farag, "Support vector machines for camera calibration problem," *Proc. IEEE ICIP* 1029–1032 (2006).
18. H He, H Li, Y Huang, J Huang, and P Li, "A novel efficient camera calibration approach based on K-SVD sparse dictionary learning," *Measurement* **159**, 107798 (2020).
19. Y Zhang, X Zhao, and D Qian, "Learning-Based Framework for Camera Calibration with Distortion Correction and High Precision Feature Detection," *arXiv*, arXiv:2202.00158 (2022).
20. S N Raza, H R ur Rehman, S G Lee, and G S Choi, "Artificial intelligence based camera calibration," *Proc. IEEE IWCMC* 1564–1569 (2019).
21. B Chen, C Xiong, and Q Zhang, "CCDN: Checkerboard corner detection network for robust camera calibration," *Proc. ICIRA* 324–334 (2018).
22. B Chen, Y Liu, and C Xiong, "Automatic checkerboard detection for robust camera calibration," *Proc. IEEE ICME* 1–6 (2021).
23. G Bradski and A Kaehler, "200 OpenCV Dr. Dobb's J.," **3** 120.
24. M Tan and Q Le, "Efficientnet: Rethinking model scaling for convolutional neural networks," *Proc. PMLR* 6105–6114 (2019).
25. M Konnik and J Welsh, "High-level numerical simulations of noise in CCD and CMOS photosensors: review and tutorial," *arXiv*, arXiv:1412.4031 (2014).

26. N Kanopoulos, N Vasanthavada, and R L Baker, "Design of an image edge detection filter using the Sobel operator," *IEEE J. Solid-State Circuits* **23**(2), 358–367 (1988).
27. M A Fischler and R C Bolles, "Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography," *Commun. ACM* **24**(6), 381–395 (1981).
28. AG Howard, M Zhu, B Chen, D Kalenichenko, W Wang, T Weyand, M Andreetto, and H Adam, "Mobilenets: Efficient convolutional neural networks for mobile vision applications," *arXiv*, arXiv:1704.04861 (2017).
29. J Hu, L Shen, and G Sun, "Squeeze-and-excitation networks," *Proc. CVPR*, 7132–7141 (2018).
30. K He, X Zhang, S Ren, and J Sun, "Deep residual learning for image recognition," *Proc. CVPR* 770–778 (2016).
31. S Hochreiter, Y Bengio, P Frasconi, and J Schmidhuber, "Gradient flow in recurrent nets: the difficulty of learning long-term dependencies," in S C Kremer and J F Kolen, eds., *A Field Guide to Dynamical Recurrent Neural Networks* (IEEE Press) (2001).
32. M Tan and Q Le, "Efficientnetv2: Smaller models and faster training," *Proc. PMLR*, 10096–10106 (2021).
33. D P Kingma and J Ba, "Adam: A method for stochastic optimization," *arXiv*, arXiv:1412.6980 (2014).
34. K Levenberg, "A method for the solution of certain non-linear problems in least squares," *Quart. Appl. Math.* **2**(2), 164–168 (1944).
35. RI Hartley, E Hayman, L de Agapito, and I Reid, "Camera calibration and the search for infinity," *Proc. ICCV* **1**, 510–517 vol.1 (1999).
36. N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst., Man, Cybern.* **9**(1), 62–66 (1979).