Jitter noise modeling and its removal using recursive least squares in shape from focus systems

Husna Mutahira1, Vladimir Shin2, Unsang Park1* & Mannan Saeed Muhammad3*

Three-dimensional shape recovery from the set of 2D images has many applications in computer vision and related fields. Passive techniques of 3D shape recovery utilize a single view point and one of these techniques is Shape from Focus or SFF. In SFF systems, a stack of images is taken with a single camera by manipulating its focus settings. During the image acquisition, the inter-frame distance or the sampling step size is predetermined and assumed constant. However, in a practical situation, this step size cannot remain constant due to mechanical vibrations of the translational stage, causing jitter. This jitter produces Jitter noise in the resulting focus curves. Jitter noise is invisible in every image, because all images in the stack are exposed to the same error in focus; thus, limiting the use of traditional noise removal techniques. This manuscript formulates a model of Jitter noise based on Quadratic function and the Taylor series. The proposed method, then, solves the jittering problem for SFF systems through recursive least squares (RLS) filtering. Different noise levels were considered during experiments performed on both real as well as simulated objects. A new metric measure is also proposed, referred to as depth distortion (DD), which calculates the number of pixels contributing to the RMSE in percentage. The proposed measure is used along with the RMSE and correlation, to compute and test the reconstructed shape quality. The results confirm the effectiveness of the proposed scheme.

In recent years, extensive research has been conducted towards the recovery of 3D information from its corresponding 2D images. Given a set of images, important depth information can be effectively obtained and further used for 3D reconstruction. This information is used in various applications, such as robotic manipulation, automatic inspection, medical imaging, microscopy, consumer cameras, bioinformatics etc.1-6

Shape from Focus (SFF) is the process of reconstructing the depth of the scene by actively changing the camera optics until the point of interest is in focus. It is one of the passive techniques that uses one camera for 3D shape reconstruction. The SFF system must record a large sequence of image frames of the object/scene that correspond to different levels of the object/scene in focus7. The camera optics can be changed by changing the lens position or the object position relative to the camera. The depth of the focused point can be obtained through the thin lens Gaussian law:

\[ \frac{1}{f} = \frac{1}{u} + \frac{1}{v}, \]

wherein \( f \) denotes the focal length of the imaging device. The distance of the object point from the imaging device is given by \( u \), and the position of the object point where it is best focused by the lens is represented by \( v \), and is given in Fig. 1.

Images can be obtained by manipulating any of the above-mentioned factors, but typically \( u \) of the system is varied to acquire images. The optical microscope is an example of such type of a system. Nonetheless, for the SFF systems the magnification of the imaging system should be constant and the depth of field should be as shallow as possible4. After image acquisition, the resulting image stack \( I \) is represented by \( l \times m \times n \) dimensions. Each

1Department of Computer Science and Engineering, Sogang University, Seoul 04107, South Korea. 2Department of Information and Statistics, Research Institute of Natural Science, Gyeongsang National University, Jinju 52828, South Korea. 3Department of Electrical and Computer Engineering, Sungkyunkwan University, Suwon 16419, South Korea. *email: unsangpark@sogang.ac.kr; mannan@skku.edu
pixel in the stack is represented by $P(i,j)(k)$, where $1 \leq i \leq l$, $1 \leq j \leq m$ and $1 \leq k \leq n$ are the indices in the $l$, $m$, and $n$ directions. $P(i,j)(k)$ also represents the pixel curve along the optical axis. This is shown as in Fig. 2. The number of images ($n$) can be calculated by:

$$n = \frac{U_{disp}}{\Delta},$$

where $\Delta$ in the above equation represents the sampling step size and $U_{disp}$ represents the total displacement of the translational stage during image acquisition. For changes in $u$, the step size expression is provided by Muhammad and Choi.

Since SFF algorithms use image focus analysis to recover 3D shape of an object, a focus measure (FM) is calculated over small image regions of the image frame in the image sequence. The value of the FM increases with the increase in image sharpness and achieves a maximum value when the image is best focused. Since the advent of SFF methods, various FMs have been proposed to measure the degree of focus present in the image. Their performance can be influenced under various conditions, such as image noise level, contrast, saturation and window size, and the effects of these conditions on FMs have been extensively studied in recent years.

A brief introduction of popular FMs used for comparison in this manuscript is given here: Sum of Modified Laplacian ($SML$), and Tenengrad or Tenenbaum ($TENG$) use squares of the second derivative of the image and first derivative of the image, respectively. Gray Level Variance ($GLVA$) is a statistical method to compute variance as focus measurement whereas Image Curvature ($CURV$) calculates the image surface curvature. Some other FMs used for experiments in this study include Image Contrast ($CONT$), 3D Laplacian ($GRA-3$), Discrete Cosine Transform Energy ($DCTE$), Spatial Frequency ($SFRQ$), Sum of Wavelet Coefficients ($WAVE$), and Steerable Filters ($SFIL$).

After FM application an initial depth map estimate is obtained which can be further improved by refinement procedures. Traditional SFF techniques utilize SML and apply Gaussian interpolation to compute intra-frame values for better focus. Piecwise curved surface approximation was used for the focused image surface and curved focused image surface, respectively. Traditional and Deep Neural Networks were introduced by various authors to enhance the results of SFF techniques. Another method to improve the efficiency of neural networks, by introducing the weight passing method, was also employed. Pulsed coupled neural network to aggregate shape using focus was given by Yan et al. Another method to improve the accuracy of 3D shapes. The wavelet transform method was also used to improve shape reconstruction. Guided image filtering is also employed for depth enhancement in SFF.

Recovery of several 3D shapes by applying different FMs, and combining their results into one final shape is also one of the popular techniques. A combination of 3D steerable filters on treating texture-less regions were also utilized to enhance the texture-less surface reconstruction. Another SFF method based on the analysis of 3D structure tensor of the image sequence was proposed by Mahmood et al. A method for depth reconstruction that used non-local matting Laplacian along with Markov random field was developed by Ma et al. Shape optimization through non-parametric regression is presented by Jang et al. Adaptive sum of weighted modified Laplacian is also proposed.

Figure 1. Focusing via Gaussian lens law.
Focus measurement and focus curve modeling

Focus measurement is the main concept in the SFF systems, and the sharpness criterion used to evaluate the level of focus in an image is called the focus measure operator (FM)\textsuperscript{15}. The principle behind FM is to respond to the high-frequency content in the image, and ideally it should produce maximum response when the image is focused\textsuperscript{32}. The main objective of FM in SFF systems is to provide a sharp focus curve (in parallel to the optical axis) for every object point in the image stack\textsuperscript{33}. The FM is applied to each pixel of the image stack, as given in the following equation:

$$\Theta_{(i,j)}(k) = F\{P_{(i,j)}(k)\},$$

where $F$ is the FM transformation of pixel $P_{(i,j)}(k)$ that results in the focus value $\Theta_{(i,j)}$ in the $k$th image. It represents the focus behavior or the focus curve of the pixel\textsuperscript{33}.

In conventional methods, the initial depth map $D_{(i,j)}$ is obtained by maximizing the focus curve along the optical axis\textsuperscript{34-37}. The value of $k$ is obtained using (4), where $\Theta_{(i,j)}(k)$ is at the maximum according to:

$$D_{(i,j)} = \underset{k}{\arg \max}(\Theta_{(i,j)}(k)).$$

However, the obtain depth map can be further improved by some depth adjustment using the correction suggested by Shim\textsuperscript{38}:

$$D_{\text{corrected}}^{(i,j)} = f \times \frac{D_{(i,j)}}{D_{(i,j)} - f},$$

where $f$ denotes the focal length of the imaging device.

Figure 2. Image acquisition in SFF systems.
The focus behavior of every individual pixel, or in other words the focus curves, depends on various factors such as the type of FM used, the camera parameters, and most importantly, the image texture around that object point\(^9,13,39\). If the images are properly acquired, then all the resulting focus curves are bell-shaped\(^{16}\). The Gaussian model\(^{15,29}\), Lorentz-Cauchy model\(^9\), and Quadratic model\(^40\) are used to approximate these bell-shaped focus curves\(^{41}\).

The Gaussian model\(^{42}\) is given by:

\[
\Theta_G(k) = A_G \exp \left[ -\frac{(k-B_G)^2}{2C_G^2} \right],
\]
(6)

the Lorentz-Cauchy model is given by:

\[
\Theta_L(k) = A_L \frac{B_L^2}{B_L^2 + (k-C_L)^2},
\]
(7)

and the Quadratic model is given by:

\[
\Theta_Q(k) = A_Q k^2 + B_Q k + C_Q,
\]
(8)

where the \(A_s\), \(B_s\) and \(C_s\) are the parameters of the respective model. The unification of these models in the Quadratic model is provided by\(^{41}\). If (6) is simplified after logarithmic transformation, it transforms to (8) as follows:

\[
\Theta_G \rightarrow \Theta_Q,
\]

\[
\log(\Theta_G(k)) = \log(A_G) - \frac{1}{2C_G^2} (k^2 + B_G^2 - 2B_G k).
\]
(9)

Equation (7) can be transformed to (8) by applying a reciprocal transformation and some simplification, as follows:

\[
\Theta_L \rightarrow \Theta_Q,
\]

\[
\frac{1}{\Theta_L(k)} = \frac{1}{A_L B_L^2} (B_L^2 + k^2 + C_L^2 - 2C_L k).
\]
(10)

The transformation of the Gaussian and Lorentz-Cauchy models to the Quadratic model is shown in Fig. 3. In the next section, the Quadratic model as given by (8) is used to model the jitter noise in SFF systems.
Motivation
Shape from focus has been under research for many years, but there exist some unresolved problems that impact the system performance. One of them is the instability in the sampling step size. The images are meant to be obtained at different focus levels, at a predetermined constant step size. However, in practice, this step size cannot remain constant because of the mechanical aspects of the imaging device and lens-focusing methods. The instability in sampling step size is referred to as Jittering or Jitter noise. This noise alters the focus values of the image, by oscillating along the optical axis and propagates through the entire stack. Since the same level of error is present in all the images, it is impossible to eliminate it through traditional de-noising techniques.

Jang et al. used the Kalman filter to remove this noise. They also proposed other methods involving Kalman filter. However, all these methods used Kalman filter with scalar models. The system matrix was considered as 1, hence ignoring the dynamic nature of the focus cues. All of their methods obtained multiple images for eliminating jitter at each step. It means if there were images in the stack, samples for each step were obtained, thus requiring samples for each focus curve. However, similar results can be obtained by taking the mean of the measurement values on every step position. Also, the large number of images increases the complexity of the system, resulting in high computational costs, impacting the practical use of their methods. Jitter noise is assumed to have Normal (symmetric) and Levy distributions (non-symmetric) with fixed parameters. Later, a dynamic Kalman filter was used to address these issues proving that the resultant Jitter noise is dynamic in nature and is position-dependent. Its probability density function changes on every step, and depends on many factors.

In this manuscript, the modeling of Jitter noise is presented in section "Focus measurement and focus curve modeling", and the recursive least squares (RLS) filter is designed to eliminate this noise from the SFF systems. In this method, a single new data point is analyzed in each algorithm iteration in order to improve the estimation of the model parameters. Since RLS converges faster, an RLS filter is designed to remove the Jitter noise from the focus curves. The Quadratic function is used to design the measurement model of focus curves, and an expression for shape recovery is followed. In the proposed scheme, a single measurement is taken at each step. Thus, as opposed to previously proposed methods, only samples are required for each focus curve.

The manuscript organization is as follows. Section “Proposed method” discusses Jitter noise modeling and the proposed methodology of this noise removal using RLS. Section “Results and discussions” presents the results and discussions. Finally, Section “Conclusion” presents the conclusion of this study.

Proposed method
Jittering or Jitter noise is an error in signal amplitude, caused by the variation in sampling frequency while sampling a signal. In SFF systems, Jitter occurs when there is uncertainty or unevenness in the sampling step size of either or depending on the variable factor to acquire the stack of images. This section discusses the step size in both situations of image acquisition, followed by modeling of Jitter noise which will be later utilized by the proposed model.

Conditions and assumptions for SFF. There are four main conditions for image acquisition in Shape from Focus systems. First, the object is moved towards (or away) from the lens, ensuring that the whole object is first defocused then it is gradually focused (on every point) and then completely defocused again. This condition ensures that the object only moves along the optical axis and there is no movement perpendicular to the optical axis. Second, as the object is moved, the magnification of the imaging system is assumed to be constant for the image areas that are perfectly focused. Third, the body (object) is piece-wise constant, and there is no occlusion in the scene. Last, the Depth of Field (DoF) of the imaging device in SFF systems must be finite and shallow (as much as possible). If any of these three conditions is violated then SFF algorithms cannot be applied unless pre-processing of the image-stack is performed, e.g. image registration, image resize, etc.

The first condition ensures that complete bell-shaped focus curves are obtained. If this condition is violated, then obtaining complete bell-shaped focus curves becomes difficult or impossible. If the DoF is infinite or large, then the acquired images will have the same degree of focus, violating the second condition, thus making it difficult to measure depth/shape using image focus. The shallow DoF also guarantees that the magnification for focused points in the image-stack, is not changed or the change is minimal.

Step size in SFF image acquisition. Observing all the necessary conditions of SFF mentioned above, i.e. the object is moved towards or away from the lens in constant small steps of , the depth of field is as shallow as possible, and the focal length as well as magnification are kept constant; the step size expression is provided by Muhammad and Choi. The simplified expression for (step size) can be given as:

\[
\Delta u = \frac{\text{DoF}}{12},
\]

which is the maximum limit for .
As mentioned before, the images can also be obtained by changing the focal length of the system in small, constant increments of . In this case the focal length of the device is varied while the object is held static in front of the imaging device. This type of technique is mostly used by auto-focusing algorithms, for searching the best focal position for a single point. It can also be used for depth and shape estimation of the object under consideration. Regardless of the image acquisition technique, an image is stored at every step to obtain a stack of images.
Modeling Jitter in SFF. This paper utilizes Quadratic function to model the jitter:

\[ f(k) = ak^2 + bk + c, \] (12)

where \( a, b \) and \( c \) are the function parameters, and \( f(k) \) is the Quadratic function. The sample points of this function are represented by \( 1 \leq k \leq n \). The step size is \( \Delta k \), and is 1. For modeling jitter noise, the uncertainty in step size is considered as, \( \zeta \sim N(0, \sigma_\zeta^2) \) and thus (12) can be written as follows:

\[ f(k + \zeta) = a(k + \zeta)^2 + b(k + \zeta) + c. \] (13)

By expanding the squared terms and simplifying using the Taylor series, (13) can be written as:

\[ f(k + \zeta) = ak^2 + bk + c + \zeta(2ak + b) + \zeta^2a. \] (14)

the following equation is obtained by using (12) and (14):

\[ f(k + \zeta) = f(k) + \zeta f'(k) + \zeta^2 f''(k)/2. \] (15)

It can be observed in (15) that the noise on the RHS of the equation is multiplied by the first and second derivatives of the function. Therefore, it can be concluded that the Jitter noise, in SFF systems, is dependent on the slope and concavity of the focus curves. If \( \zeta \) is Normal (\( N(0, \sigma_\zeta^2) \)), then \( (k + \zeta) \) follows \( N(k, \sigma_\zeta^2) \), which given by:

\[ p(\zeta) = \frac{1}{\sigma_\zeta \sqrt{2\pi}} \exp \left[ -\frac{1}{2} \left( \frac{\zeta - k}{\sigma_\zeta} \right)^2 \right]. \] (16)

Rewriting (15):

\[ f(k + \zeta) = f(k) + \xi_1 + \xi_2 \]
\[ = f(k) + \xi, \] (17)

where \( \xi_1 \) and \( \xi_2 \) are given by:

\[ \xi_1 = \zeta f'(k) = \zeta(2ak + b), \]

and,

\[ \xi_2 = \zeta^2 f''(k)/2 = \zeta^2a, \]

where \( \xi_1 \) is normally distributed, mean \( \mu_1 = 0 \) and variance \( \sigma_1^2 = (2ak + b)^2\sigma_\zeta^2 \).

Meanwhile, \( \xi_2 \) follows a Gamma (\( \Gamma \)) distribution, with mean \( \mu_2 = a\sigma_\zeta^2 \), and variance \( \sigma_2^2 = 2a^2\sigma_\zeta^4 \). The value of the variance \( \sigma^2 \) is different at every kth step, and becomes zero at:

\[ k = -\frac{b}{2a} \]

The direction of \( \xi_2 \) depends on the sign of \( a \), and will always tend to the concavity of the function. The range (\( R(\xi) \)) of jitter noise is provided by (18):

\[ R(\xi) = \begin{cases} 
\xi \geq \frac{-2ak+b}{4a} & a > 0, \\
\xi \leq \frac{-2ak+b}{4a} & a < 0, \\
-\infty < \xi < \infty & a = 0.
\end{cases} \] (18)

In order to obtain the pdf (\( p(\xi) \)) of the Jitter noise, \( \xi_1 \) and \( \xi_2 \) can be combined as \( \xi = \xi_1 + \xi_2 \), and \( p(\xi) \) is given by (19), where the symbols \( r_1 \) and \( r_2 \) in (19) are computed using the following equation:

\[ r_{1,2} = \frac{-(2ak + b) \pm \sqrt{(2ak + b)^2 + 4a\xi}}{8a^2\sigma_\zeta^2}. \]

Expression (19) holds only if \( \xi \) follows normal distribution. The jitter noise \( \xi \) is the resulting noise in focus values due to uncertainty \( \zeta \) in sampling step \( \Delta u \).

\[ p(\xi) = \begin{cases} 
\frac{1}{\sigma_\zeta \sqrt{2\pi}} \sqrt{\frac{1}{(2ak+b)^2 + 4a\xi}} \left( \exp[r_1] + \exp[r_2] \right) & \forall \ \xi \in (-\infty, -\frac{(2ak+b)^2}{4a}) \cup (-\frac{(2ak+b)^2}{4a}, \infty), a \neq 0, \\
\frac{1}{\sigma_\zeta \sqrt{2\pi}} \exp \left( \frac{\xi^2}{2b^2\sigma_\zeta^4} \right) & \forall \ \xi \in (-\infty, \infty), a = 0.
\end{cases} \] (19)

Proposed Jitter removal methodology. The proposed scheme can be applied after the FM transformation is performed using (3). Pertaining to its faster convergence property, a recursive least squares (RLS) filter
is designed to fully eliminate the jitter noise from the focus curves. RLS is one of the most well-known adaptive filters that recursively calculate the coefficients of a given function, minimizing the cost function related to input measurements.\

Recursive least squares filter design. The measurements of focus curves (of every individual pixel) depend on various factors. These focus curves are bell-shaped and hence different models such as Gaussian model, Lorentz-Cauchy model, and Quadratic model are used to approximate them. The advantage of the Quadratic model is that it makes the focus profiles collected by different FMs to be modeled in a common form that is easy to handle and has computational advantages. Thus, the measurement model of the proposed RLS scheme uses the Quadratic function given in (12), and can also be represented using (20):

\[
f(k) = \begin{bmatrix} k^2 & k & 1 \end{bmatrix} \begin{bmatrix} a \\ b \\ c \end{bmatrix},
\]

where \(a\), \(b\), and \(c\) are the unknown constant coefficients, and by measuring this function, a measurement model is obtained as:

**Data:** Image Stack \(I\) with dimensions \(l \times m \times n\) with Jitter noise \(N(0, \sigma^2_\xi)\).

**Result:** Recovered 3D shape with dimensions \(l \times m\)

**Apply focus measure:**

for \(k \leftarrow 1\) to \(n\) do

| Apply focus measure on every \(k\)th image \(I(k)\) to get modified image: \(\Theta(k) = \mathcal{F} \{I(k)\}\); |

end

**Compute Shape:**

Set \(T\);

for \(i \leftarrow 1\) to \(l\) do

| for \(j \leftarrow 1\) to \(m\) do |

\(\kappa \leftarrow 1;\)

\(\kappa_r \leftarrow 0;\)

Initialize: \(\sigma^2_\xi, \Psi(0), X(0) = [a, b, c]^T;\)

for \(k \leftarrow 1\) to \(n\) do

if \(\Theta(i,j)(k) \geq T\) then

Measurement: \(y(\kappa) = \Theta(i,j)(k);\)

Compute:

Measurement Matrix: \(H(\kappa);\)
Parametric Estimate: \(\hat{X}(\kappa);\)
Filter Gain: \(L(\kappa);\)
Covariance: \(\Psi(\kappa);\)
if \(\kappa = 1\) then

\(\kappa_r \leftarrow \kappa;\)

end

\(\kappa \leftarrow \kappa + 1;\)

else

Discard: \(\Theta(i,j)(k);\)

end

end

Compute Depth of Point \((i, j);\)

\[LD(i,j) = \left( -\frac{b}{2a} \right)_{(i, j)} + \kappa_r - 1;\]

end

**Algorithm 1:** Proposed Algorithm for shape reconstruction.

\[
y(k) = H(k)X + \xi(k),
\]

where \(y(k)\) denotes the measurements of focus curve after FM application at the \(k\)th image, \(H(k)\) is the measurement matrix at \(k\), and \(\xi(k)\) is the measurement error representing jitter noise with pdf given in (19). The unknown parametric vector is \(X = [a, b, c]^T\). The RLS gain \(L(k)\) is computed on every step \(k\) as follows:
\[ L(k) = \Psi(k-1)H(k)^T \left[ H(k)\Psi(k-1)H(k)^T + \sigma_b^2 \right]^{-1}, \]

where \( \sigma_b^2 = \text{Var}(\xi) \) is the variance of Jitter noise. The optimal least squares estimate of the parametric vector \( X \) and covariance \( \Psi(k) \) are computed by the following recursive equations:

\[
\begin{align*}
\hat{X}(k) &= \hat{X}(k-1) + L(k)\left[ y(k) - H(k)\hat{X}(k-1) \right], \\
\Psi(k) &= \Psi^{-1}(k-1) + \frac{1}{\sigma_b^2}H^T(k)H(k)\Psi(k-1). 
\end{align*}
\]

In (23) the covariance \( \Psi(k) \) can be redesigned to remove the matrix inversions using the Sherman-Morrison formula\(^\text{53-56}\), and can be rewritten as follows:

\[ \Psi(k) = \Psi(k-1) - \frac{1}{\sigma_b^2 + H\Psi(k-1)H^T}H(k)\Psi(k-1)H^T(k)^{-1}. \]

The summary of equations for the proposed filter design is provided in Table 1.

### Table 1. Summary of RLS filter equations for proposed scheme.

| Equation name | Equation |
|---------------|----------|
| \( \dot{X}(k) \) | \( aX^2 + bX + c \) |
| Parameters vector | \( X = [a, b, c]^T \) |
| Measure-ment vector | \( y(k) = H(k)X + \xi(k) \) |
| Measure-ment matrix | \( H(k) = [k_i, k, 1] \) |
| Recursive least squares filter equations | |
| Filter gain: \( L(k) \) | \( \Psi(k-1)H(k)^T \left[ H(k)\Psi(k-1)H(k)^T + \sigma_b^2 \right]^{-1} \) |
| Estimate: \( \hat{X}(k) \) | \( \hat{X}(k) = \hat{X}(k-1) + L(k)\left[ y(k) - H(k)\hat{X}(k-1) \right] \) |
| Covariance: \( \Psi(k) \) | \( \Psi^{-1}(k-1) + \frac{1}{\sigma_b^2}H^T(k)H(k)\Psi(k-1) \) |

**Proposed algorithm.** The proposed algorithm is listed in Algorithm 1. First, the FM is applied on every image of the image stack \( I \) to obtain the modified image stack \( \Theta \) as follows:

\[ \Theta(k) = \mathcal{F}[I(k)]. \]

After the modified stack is obtained, the values of the focus curve \( \Theta_{(i,j)}(k) \) for each object point are taken and compared to a threshold value \( T \) (which is set heuristically to ignore tails of the focus curves). The values of \( \Theta_{(i,j)}(k) < T \) are ignored. Additionally, for values of \( \Theta_{(i,j)}(k) \geq T \), the value of index \( k \) is stored in \( k_i \) followed by initialization of a new index parameter \( \kappa \). The measurement value of focus for point \( (i, j) \) in consideration is taken as \( y(\kappa) = \Theta_{(i,j)}(k) \). The next step calculates state measurement matrix \( (H(\kappa)) \), parametric vector \( (\hat{X}(\kappa)) \), covariance matrix \( (\Psi(\kappa)) \), and filter gain \( (L(\kappa)) \) using equations summarized in Table 1. After completion of all the iterations, the depth of the object point \( (i, j) \) is computed by modifying (25) as follows:

\[ LD_{(i,j)} = \left( -\frac{b}{2a} \right)_{(i,j)} + \kappa - 1. \]

### Results and discussions
The experimental results and discussions are presented in this section. The section is divided into three subsections. First, the experimental setup details are given, followed by the reconstructed-shape assessment criteria, and later a detailed analysis of the effects of Jitter noise and its removal using RLS filter for SFF is provided.
Experimental setup. The experiments are performed on seven objects for shape reconstruction analysis. A summary of the objects used in the 3D shape analysis is provided in Table 2. Ten simulated datasets of simulated cone are generated with different focus positions and Jitter noise levels using a camera simulation software (AVS). The details of AVS are provided in13,37,57. The MATLAB code used to generate the simulated object image set is downloaded from13. The AVS software is inputted with the depth map, texture image, and camera parameters. The texture map comprises concentric circles with alternating black and white stripes. The depth maps and the texture images used for image generation via AVS for all sequences, of Simulated Cone are similar. The difference in each dataset is the uncertainty $\zeta \sim N(0, \sigma_\zeta^2)$ in step size $\Delta u$ used to generate the sequences to study the effect of Jitter on shape reconstruction. The values of noise $\zeta$ with variance $\sigma_\zeta^2 = 0$ and $\sigma_\zeta^2 = 2.0$, Real Cone, Real Plane, LCD-TFT, Groove, Coin and Image-I.

Table 2. Summary of experimented objects.

| Name          | Object Type | Image dimensions | Number of images |
|---------------|-------------|------------------|------------------|
| Simulated cone| Simulated   | $360 \times 360$ | 97               |
| Real cone     | Real        | $200 \times 200$ | 97               |
| Real plane    | Real        | $200 \times 200$ | 87               |
| LCD-TFT      | Real        | $200 \times 200$ | 60               |
| Groove       | Real        | $200 \times 200$ | 60               |
| Coin         | Real        | $200 \times 200$ | 68               |
| Image-I      | Real        | $200 \times 200$ | 60               |

Figure 4. Ground truth of Simulated Cone and Real Cone.

Figure 5. 10th frame of image sequence of each experimented object. (left to right) Simulated Cone with $\sigma_\zeta^2 = (2ak + b)^2\sigma_\zeta^2 + 2a^2\sigma_\zeta^4$, where $\sigma_\zeta^2 = 0$ and $\sigma_\zeta^2 = 2.0$, Real Cone, Real Plane, LCD-TFT, Groove, Coin and Image-I.

Images of Real Cone and Real Plane were taken using the CCD camera system17. The LCD-TFT filter images are microscopic images of an LCD color filter. The coin sequence consists of magnified images of Lincoln’s head from the back of the US penny. The sequence of Image-I is the letter I engraved in a metallic surface. These image sequences were originally in grayscale. Figure 4 provides the ground truths of the Simulated and Real cones. Figure 5 shows the 10th frame of each image sequence. These image sequences have been widely used by many researchers including11,24,40,58–60.
is the obtained depth map, and true \( G \) is the ground truth. \( D \) is the depth map obtained from the reconstruction process. The covariance, \( \sigma \) is the standard deviation of the data, \( \mu \) is the mean of the data, and \( \xi \) is the noise level. The Root Mean Square Error (RMSE) is the square root of the variance of the residuals of the data under observation. This indicates how close the perceived shape is to the original shape and is given by:

\[
RMSE = \sqrt{\frac{1}{(l \times m) \sum_{i} \sum_{j} (G_{(i,j)}^{true} - D_{(i,j)}^{obtained})^2}}
\]

(29)

where \( G_{(i,j)}^{true} \) is the ground truth, \( D_{(i,j)}^{obtained} \) is the obtained depth map, and \( l \times m \) are the dimensions of the depth maps. A higher value of RMSE indicates a larger error in the shape reconstruction. For better results, the value of RMSE should be closer to zero.

Correlation is a similarity measure between two shapes, and is given by:

\[
\rho_{GD} = \frac{\text{cov}(G_{(i,j)}^{true}, D_{(i,j)}^{obtained})}{\sigma_G \sigma_D}
\]

(30)

where \( \text{cov} \) is the covariance, and \( \sigma_G \) and \( \sigma_D \) are the variances of \( G_{(i,j)}^{true} \) and \( D_{(i,j)}^{obtained} \), respectively.

Depth Distortion (DD) metric indicates the number of pixels (in percentage), which contribute to RMSE. It is calculated as follows:

\[
DD_{GD} = \left( \frac{|\sum_{i \times m} E_G | - (\sum_{i \times m} E_D |)}{l \times m} \right) \times 100,
\]

(31)

where, \( E_G \) and \( E_D \) are the edge images of the ground truth and the obtained depth map, respectively, and can be calculated by applying edge operators (e.g., Canny or Sobel operator). If the ground truth has any edge, then the number of pixels contributing to that edge should be similar to the obtained depth map; thus, the difference will be zero. But if the obtained depth map has some distortions in depth, then the edge image will have extra pixels from depth distortion; hence, it will contribute to RMSE. The proposed metric determines the quantity of these pixels.

**Experimental results and discussion.** Six well-known FMs–Sum of Modified Laplacian (SML), Tenenbaum (TENG), Grey Level Variance (GLVA), Image Contrast (CONT), 3D Laplacian (GRA3), and Spatial Frequency (SFRQ) are used for the current experiments.

The noise \( \xi \sim N(0, \sigma^2) \) is simulated with different values of variance \( \sigma^2 \) in the noise. The simulation values for variance \( \sigma^2 \) are \( 0, 0.1, 0.25, 0.5, 0.75, 1.0, 1.25, 1.5, 1.75, \) and \( 2.0 \). The experiments are first performed on the Simulated Cone image sequence by simulating this noise for all the six above-mentioned FMs. Tables 4, 5, 6, 7, 8, 9 show the results of the reconstructed shape of Simulated Cone with Jitter noise with variance \( \sigma^2 \) = 0.1. All these methods use 100 iterations per step, whereas, the proposed scheme utilizes only a single measurement for each step. RMSE, correlation and
### Table 4. RMSE, Correlation (Cor.) and Depth Distortion (DD) in % for shape reconstruction using FM:SML with different levels of variance of Jitter noise ($\sigma^2_\xi = (2ak + b)^2 \sigma^2_\xi + 2a^2 \sigma^4_\xi$), with and without proposed RLS filter.

| $\sigma^2_\xi$ | Cor. | RMSE | DD |
|----------------|------|------|----|
|                | With FM only | With RLS & FM | With FM only | With RLS & FM |
| 0              | 0.951 | 0.952 | 4.855 | 4.811 | 2.668 | 0.445 |
| 0.1            | 0.953 | 0.953 | 4.830 | 4.796 | 2.691 | 0.738 |
| 0.25           | 0.952 | 0.952 | 4.880 | 4.833 | 2.894 | 1.140 |
| 0.5            | 0.952 | 0.953 | 4.833 | 4.835 | 4.617 | 0.824 |
| 0.75           | 0.951 | 0.952 | 4.832 | 4.778 | 4.725 | 1.603 |
| 1              | 0.940 | 0.951 | 5.179 | 4.821 | 4.502 | 2.613 |
| 1.25           | 0.949 | 0.949 | 5.261 | 4.848 | 3.859 | 2.244 |
| 1.5            | 0.929 | 0.949 | 5.425 | 4.928 | 5.413 | 2.508 |
| 1.75           | 0.938 | 0.946 | 5.990 | 5.012 | 5.237 | 2.633 |
| 2              | 0.931 | 0.949 | 5.880 | 5.035 | 5.004 | 1.616 |
| Av.            | 0.945 | 0.951 | 5.197 | 4.870 | 4.161 | 1.636 |

### Table 5. RMSE, Correlation (Cor.) and Depth Distortion (DD) in % for shape reconstruction using FM:CONT with different levels of variance of Jitter noise ($\sigma^2_\xi = (2ak + b)^2 \sigma^2_\xi + 2a^2 \sigma^4_\xi$), with and without proposed RLS filter.

| $\sigma^2_\xi$ | Cor. | RMSE | DD |
|----------------|------|------|----|
|                | With FM only | With RLS & FM | With FM only | With RLS & FM |
| 0              | 0.947 | 0.948 | 4.991 | 4.936 | 2.084 | 0.640 |
| 0.1            | 0.946 | 0.948 | 4.992 | 4.933 | 2.192 | 0.616 |
| 0.25           | 0.947 | 0.947 | 5.013 | 4.966 | 2.139 | 0.646 |
| 0.5            | 0.947 | 0.949 | 4.966 | 4.941 | 2.198 | 0.632 |
| 0.75           | 0.947 | 0.948 | 5.006 | 4.987 | 2.167 | 0.777 |
| 1              | 0.937 | 0.949 | 5.218 | 4.887 | 1.998 | 0.664 |
| 1.25           | 0.941 | 0.945 | 5.265 | 4.986 | 2.235 | 1.015 |
| 1.5            | 0.931 | 0.947 | 5.346 | 4.998 | 2.737 | 1.894 |
| 1.75           | 0.939 | 0.949 | 5.183 | 4.939 | 3.215 | 2.306 |
| 2              | 0.934 | 0.949 | 5.151 | 4.924 | 2.532 | 0.985 |
| Av.            | 0.942 | 0.948 | 5.113 | 4.950 | 2.350 | 1.018 |

### Table 6. RMSE, Correlation (Cor.) and Depth Distortion (DD) in % for shape reconstruction using FM:GLVA with different levels of variance of Jitter noise ($\sigma^2_\xi = (2ak + b)^2 \sigma^2_\xi + 2a^2 \sigma^4_\xi$), with and without proposed RLS filter.

| $\sigma^2_\xi$ | Cor. | RMSE | DD |
|----------------|------|------|----|
|                | With FM only | With RLS & FM | With FM only | With RLS & FM |
| 0              | 0.951 | 0.951 | 4.879 | 4.872 | 2.495 | 1.009 |
| 0.1            | 0.951 | 0.952 | 4.872 | 4.861 | 2.504 | 0.964 |
| 0.25           | 0.952 | 0.952 | 4.870 | 4.866 | 3.017 | 1.142 |
| 0.5            | 0.951 | 0.952 | 4.892 | 4.859 | 4.590 | 1.681 |
| 0.75           | 0.951 | 0.951 | 4.860 | 4.856 | 4.663 | 2.135 |
| 1              | 0.941 | 0.953 | 5.170 | 4.785 | 3.026 | 2.243 |
| 1.25           | 0.950 | 0.952 | 4.950 | 4.970 | 3.787 | 1.729 |
| 1.5            | 0.928 | 0.951 | 5.434 | 4.882 | 5.391 | 2.601 |
| 1.75           | 0.939 | 0.948 | 5.259 | 5.052 | 3.367 | 2.552 |
| 2              | 0.931 | 0.945 | 5.370 | 5.090 | 4.270 | 2.983 |
| Av.            | 0.944 | 0.951 | 5.055 | 4.909 | 3.711 | 1.904 |
### Table 7. RMSE, Correlation (Cor.) and Depth Distortion (DD) in % for shape reconstruction using FM:GRA3 with different levels of variance of Jitter noise ($\sigma^2_\xi = (2ak + b)^2 \sigma^2_\zeta + 2a^2 \sigma^4_\xi$), with and without proposed RLS filter.

| $\sigma^2_\xi$ | Cor.     | RMSE With FM only | RMSE With RLS & FM | DD With FM only | DD With RLS & FM |
|---------------|----------|--------------------|--------------------|----------------|-----------------|
| 0             | 0.950    | 4.924              | 4.9221             | 4.017          | 0.459           |
| 0.1           | 0.950    | 4.909              | 4.926              | 3.664          | 0.451           |
| 0.25          | 0.950    | 4.918              | 4.950              | 3.253          | 0.410           |
| 0.5           | 0.948    | 4.963              | 4.868              | 4.163          | 0.573           |
| 0.75          | 0.952    | 4.894              | 4.831              | 4.201          | 1.368           |
| 1             | 0.958    | 5.224              | 4.822              | 3.211          | 1.166           |
| 1.25          | 0.949    | 4.955              | 4.923              | 3.796          | 1.600           |
| 1.5           | 0.930    | 5.388              | 4.941              | 4.713          | 2.405           |
| 1.75          | 0.940    | 5.248              | 4.980              | 3.971          | 1.949           |
| 2             | 0.943    | 5.322              | 5.015              | 3.971          | 1.617           |
| Av.           | 0.945    | 5.075              | 4.918              | 3.896          | 1.200           |

### Table 8. RMSE, Correlation (Cor.) and Depth Distortion (DD) in % for shape reconstruction using FM:SFRQ with different levels of variance of Jitter noise ($\sigma^2_\xi = (2ak + b)^2 \sigma^2_\zeta + 2a^2 \sigma^4_\xi$), with and without proposed RLS filter.

| $\sigma^2_\xi$ | Cor.     | RMSE With FM only | RMSE With RLS & FM | DD With FM only | DD With RLS & FM |
|---------------|----------|--------------------|--------------------|----------------|-----------------|
| 0             | 0.947    | 5.006              | 4.957              | 0.617          | 0.441           |
| 0.1           | 0.946    | 5.009              | 4.946              | 0.582          | 0.432           |
| 0.25          | 0.946    | 5.033              | 4.990              | 0.651          | 0.421           |
| 0.5           | 0.949    | 4.935              | 4.919              | 0.927          | 0.422           |
| 0.75          | 0.949    | 4.994              | 4.886              | 1.093          | 0.465           |
| 1             | 0.937    | 5.269              | 4.917              | 1.814          | 0.457           |
| 1.25          | 0.945    | 5.117              | 5.006              | 1.283          | 0.562           |
| 1.5           | 0.926    | 5.486              | 5.015              | 2.450          | 1.470           |
| 1.75          | 0.934    | 5.355              | 5.009              | 1.593          | 1.002           |
| 2             | 0.939    | 5.241              | 5.098              | 1.868          | 0.687           |
| Av.           | 0.942    | 5.145              | 4.972              | 1.288          | 0.636           |

### Table 9. RMSE, Correlation (Cor.) and Depth Distortion (DD) in % for shape reconstruction using FM:TENG with different levels of variance of Jitter noise ($\sigma^2_\xi = (2ak + b)^2 \sigma^2_\zeta + 2a^2 \sigma^4_\xi$), with and without proposed RLS filter.

| $\sigma^2_\xi$ | Cor.     | RMSE With FM only | RMSE With RLS & FM | DD With FM only | DD With RLS & FM |
|---------------|----------|--------------------|--------------------|----------------|-----------------|
| 0             | 0.951    | 4.894              | 4.917              | 2.470          | 1.051           |
| 0.1           | 0.951    | 4.883              | 4.907              | 2.360          | 1.015           |
| 0.25          | 0.951    | 4.890              | 4.929              | 2.948          | 1.170           |
| 0.5           | 0.949    | 4.922              | 4.864              | 4.361          | 1.840           |
| 0.75          | 0.953    | 4.760              | 4.903              | 4.527          | 2.006           |
| 1             | 0.940    | 5.173              | 4.782              | 2.948          | 1.637           |
| 1.25          | 0.950    | 4.953              | 4.964              | 3.978          | 1.782           |
| 1.5           | 0.928    | 5.429              | 4.895              | 4.943          | 2.800           |
| 1.75          | 0.940    | 5.244              | 5.053              | 3.503          | 2.658           |
| 2             | 0.942    | 5.052              | 5.104              | 4.095          | 2.548           |
| Av.           | 0.945    | 5.020              | 4.932              | 3.613          | 1.851           |
DD results listed in the table utilize SML, GLVA, and TENG FMs. The results of proposed technique in terms all the three measures have outclassed every single of the previous methods. Along with the above mentioned methods, the proposed RLS method is also compared with the dynamic Kalman filter as proposed by Mutahira et al., which also utilizes just one iteration per step. However, since the system model in their method is modeled using Taylor’s series, it first estimates the derivatives of the function \( f'(k), f''(k) & f'''(k) \) using a single measurement of \( f(k) \) and then using these estimates (of the derivatives), the parameters of the cubic equation \((a, b, c, \text{and} \ d)\) are estimated. Later the depth is recovered using the indirect estimated values of the parameters. This indirect estimation introduces lags and errors in the final depth map. The proposed scheme presented in this manuscript is based on estimating the parameters of the quadratic equation \((a, b, \text{and} \ c)\) directly from the measurement. Thus, depth estimation using proposed scheme produces significantly lower error contributing to better RMSE, DD and correlation values, as can be seen in Table 3.

Tables 4, 5, 6, 7, 8, 9 represent the RMSE, Correlation, and Depth Distortion metrics results, for different FMs and the same FMs used with the proposed RLS scheme. The last row of each table represents the average values for RMSE, Correlation, and Depth Distortion metrics for all noise ranges. Figures 6, 7, 8 represent the graphs of the average of shape comparison metrics for all the FMs used. The shaded bars in the graphs represent the

**Figure 6.** Average of Correlation for different FMs for Jitter noise \( \sigma_\xi^2 = (2ak + b)^2 \sigma_\xi^2 + 2a^2 \sigma_\xi^4 \) with \( 0 \leq \sigma_\xi^2 \leq 2 \).

**Figure 7.** Average of RMSE for different FMs for Jitter noise \( \sigma_\xi^2 = (2ak + b)^2 \sigma_\xi^2 + 2a^2 \sigma_\xi^4 \) with \( 0 \leq \sigma_\xi^2 \leq 2 \).
average of metrics when FM is used only for Jitter noise with variance $\sigma_\xi^2$, where $0 \leq \sigma_\xi^2 \leq 2$. The solid bars in the graphs represent the average of metrics when FM is used with the proposed RLS technique. It can be seen that SML has shown the best results when used with the proposed RLS method. The tables and figures clearly show that when the proposed technique is used for all levels of Jitter noise the shape reconstruction results improve as compared to when using FM only.

Figure 9 shows the shape reconstruction of the simulated cone, when only FMs are used, and Fig. 10 represents the reconstructed shape of simulated cone when FMs are used with the proposed RLS scheme, for a noise range of $\sigma_\xi^2 = 0, 0.1, 0.5, 1.0, 1.5, 2.0$, respectively, which results in Jitter noise with $\sigma_\xi^2$. These results clearly demonstrate the effectiveness of the proposed scheme. The shape reconstruction, when only FMs are used, have rough surface when the noise is increased from 0 to 2.0; whereas, for the same noise range, the proposed scheme has shown promising results.

Table 10 shows the results of Real Cone shape reconstruction for different levels of noise. These results show the similar trends as the simulated cone. When only FMs are used, the shape reconstruction deteriorates significantly with an increase in noise, whereas when these FMs are combined with the proposed RLS technique, the noise can be filtered out resulting in better shape reconstruction. Figure 11 represents the reconstructed shape of Real Cone using different FMs only and FMs with the proposed RLS technique. When the noise increases, the surface roughness is clearly visible in the figure when only FMs are used, whereas when RLS is combined.
with these FMs the surface roughness is significantly reduced. The shape reconstruction results of Real Plane, LCD-TFT filter, Image-I, and Coin, using only FMs and FMs with the proposed RLS filter are demonstrated in Figs. 12, 13, 14 and 15. It can be seen from the figures that reconstructed shapes of Real Cone and Real Plane are smoother. When using FM only, the roughness in shape was because of jittering, which is smoothed in the reconstructed shape by the application of the proposed method. The cylindrical shape of the filter is preserved in LCD-TFT filter, and the surrounding surface is smoothed by the filtering process. Jitter in the sequence of the data-set of Image-I is quite low, thus not much difference was observed visually. However, near the vertical axis of 175 value, a depth abnormality can be observed in shape reconstruction when using all the FMs, and also with the proposed scheme in the case of Coin sequence.

The sides and center of groove are over-exposed, in the Groove image sequence. This causes texture degradation, which is critical in SFF systems. The change in focus levels is exhibited only by the slopes in the middle. Figure 16 shows the shape reconstruction results of the Groove using different FMs and FMs with the proposed filter application.
Figure 12. Shape reconstruction of Plane using FM with proposed RLS method. (left to right), SML, CONT, GLVA, GRA3, SFRQ, TENG. (top row) with FM only and (bottom row) using FM with RLS method.

Figure 13. Shape reconstruction of LCD-TFT filter using FM with proposed RLS method. (left to right), SML, CONT, GLVA, GRA3, SFRQ, TENG. (top row) with FM only and (bottom row) using FM with RLS method.

Figure 14. Shape reconstruction of Image-I using FM with proposed RLS method. (left to right), SML, CONT, GLVA, GRA3, SFRQ, TENG. (top row) with FM only and (bottom row) using FM with RLS method.

Figure 15. Shape reconstruction of Coin using FM with proposed RLS method. (left to right), SML, CONT, GLVA, GRA3, SFRQ, TENG. (top row) with FM only and (bottom row) using FM with RLS method.
Conclusion

Shape from focus is one of the passive techniques used to recover 3D shapes of the objects from their respective 2D images, by using the focus information present in the scene. SFF requires a huge number of images obtained at different focus levels, and the sampling step size for this image acquisition is taken as a constant. However, this inter-frame distance is susceptible to errors due to mechanical inaccuracies, such as flaws in gear assembly of the translational stage or lens assembly of the focusing device. These errors are termed as Jitter noise. Jitter

Figure 16. Shape reconstruction of Groove using FM with proposed RLS method. (left to right), SML, CONT, GLVA, GRA3, SFRQ, TENG. (top row) with FM only and (bottom row) using FM with RLS method.

Table 10. RMSE Correlation (Cor.) and Depth Distortion (D.D. in %) Metrics for Real Cone shape reconstruction using FMs with different levels of variance of Jitter noise $\sigma^2 = (2ak + b)^2\sigma^2 + 2a^2\sigma^2$ with ($\sigma^2: 0, 0.1, 0.25, 0.5, 0.75, 1.0$).

| FM used | $\sigma^2$ | Cor. | RMSE | D.D. |
|---------|------------|------|------|------|
|         |            | With FM only | With RLS & FM | With FM only | With RLS & FM | With FM only | With RLS & FM |
| TENG    | 0          | 0.924 | 0.926 | 3.843 | 3.774 | 0.735 | 0.435 |
|         | 0.1        | 0.924 | 0.926 | 3.826 | 3.765 | 0.690 | 0.430 |
|         | 0.25       | 0.924 | 0.926 | 3.836 | 3.775 | 0.662 | 0.438 |
|         | 0.5        | 0.923 | 0.925 | 3.853 | 3.798 | 0.733 | 0.423 |
|         | 0.75       | 0.918 | 0.923 | 3.935 | 3.856 | 1.058 | 0.470 |
| SML     | 0          | 0.927 | 0.927 | 3.711 | 3.693 | 0.975 | 0.120 |
|         | 0.1        | 0.926 | 0.927 | 3.717 | 3.690 | 0.950 | 0.225 |
|         | 0.25       | 0.928 | 0.927 | 3.725 | 3.703 | 1.418 | 0.267 |
|         | 0.5        | 0.927 | 0.928 | 3.735 | 3.693 | 1.375 | 0.238 |
|         | 0.75       | 0.863 | 0.885 | 5.220 | 4.725 | 0.143 | 0.118 |
| GLVA    | 0          | 0.926 | 0.927 | 3.760 | 3.725 | 0.418 | 0.272 |
|         | 0.1        | 0.926 | 0.928 | 3.756 | 3.714 | 0.418 | 0.285 |
|         | 0.25       | 0.927 | 0.927 | 3.771 | 3.746 | 0.435 | 0.275 |
|         | 0.5        | 0.925 | 0.926 | 3.775 | 3.760 | 0.543 | 0.267 |
|         | 0.75       | 0.891 | 0.915 | 4.601 | 4.047 | 0.265 | 0.243 |
| SFRQ    | 0          | 0.904 | 0.903 | 4.483 | 4.527 | 0.660 | 0.562 |
|         | 0.1        | 0.905 | 0.905 | 4.443 | 4.459 | 0.653 | 0.560 |
|         | 0.25       | 0.907 | 0.906 | 4.411 | 4.447 | 0.640 | 0.580 |
|         | 0.5        | 0.901 | 0.901 | 4.530 | 4.560 | 0.600 | 0.528 |
|         | 0.75       | 0.860 | 0.891 | 5.435 | 4.828 | 0.625 | 0.585 |
| CONT    | 0          | 0.923 | 0.927 | 3.866 | 3.799 | 0.980 | 0.692 |
|         | 0.1        | 0.923 | 0.927 | 3.886 | 3.804 | 0.955 | 0.708 |
|         | 0.25       | 0.923 | 0.926 | 3.908 | 3.832 | 0.980 | 0.692 |
|         | 0.5        | 0.922 | 0.926 | 3.879 | 3.808 | 0.935 | 0.685 |
|         | 0.75       | 0.905 | 0.927 | 4.290 | 3.829 | 0.892 | 0.590 |
| GRA3    | 0          | 0.919 | 0.924 | 3.976 | 3.838 | 1.015 | 0.665 |
|         | 0.1        | 0.919 | 0.924 | 3.966 | 3.833 | 1.013 | 0.680 |
|         | 0.25       | 0.920 | 0.924 | 3.988 | 3.843 | 1.135 | 0.730 |
|         | 0.5        | 0.917 | 0.923 | 4.019 | 3.842 | 1.085 | 0.680 |
|         | 0.75       | 0.911 | 0.922 | 4.104 | 3.924 | 1.053 | 0.672 |
noise is not visible in images, because each pixel in an image is subjected to the same error in focus, and thus traditional techniques of image denoising do not work in this case.

In this paper, Jitter noise is modeled using the Quadratic function for focus curves, and a recursive least squares filter is designed to remove this noise from SFF systems. In RLS, a single new data point is analyzed in each algorithm iteration to improve the estimation of the model parameters. Since it converges faster, an RLS filter is designed for shape from systems to remove the jitter noise from the focus curves. To check the robustness of the results the proposed manuscript employs the RMSE and correlation and proposes a new metric referred to as the Depth Distortion metric, which indicates the number of pixels in percentage contributing to the RMSE. Seven objects are used for experiments herein: one simulated and six real. Ten noise levels are tested on the simulated object and four levels on the real objects. The experimental results validate the effectiveness of the proposed scheme.

Data availability
The datasets used and analyzed during the current study is available from the corresponding author on reasonable request.

Received: 23 April 2022; Accepted: 5 August 2022
Published online: 18 August 2022

References
1. Mutahira, H., Muhammad, M. S., Li, M. & Shin, D.-R. A simplified approach using deep neural network for fast and accurate shape from focus. *Micros. Res. Tech.* 84, 656–667 (2021).
2. Zhou, S. & Seita, M. Large-area surface topography analysis of additively manufactured metallic materials using directional reflectance microscopy. *Mater. Sci. Eng. A* 760, 489–497 (2019).
3. Tang, J., Qiu, Z. & Li, T. A novel measurement method and application for grinding wheel surface topography based on shape from focus. *Measurement* 133, 495–507 (2019).
4. Perraud, J.-B. et al. Shape-from-focus for real-time terahertz 3d imaging, *Opt. Lett.* 44, 483–486 (2019).
5. Wendland, M. *Shape from Focus Image Processing Approach Based 3D Model Construction of Machined Part*. Ph.D. thesis, University of Pacific. (2015).
6. Ahmad, M. B. & Choi, T. S. Application of three dimensional shape from image focus in lcd/tft displays manufacturing. *IEEE Trans. Consum. Electron.* 53, 1–4 (2007).
7. Favaro, P., Mennucci, A. & Soatto, S. Observing shape from defocused images. *Int. J. Comput. Vision* 52, 25–43 (2003).
8. Muhammad, M. S. & Choi, T. S. A novel method for shape from focus in microscopy using Bezier surface approximation. *Micros. Res. Tech.* 73, 140–151 (2010).
9. Muhammad, M. & Choi, T.-S. Sampling for shape from focus in optical microscopy. *IEEE Trans. Pattern Anal. Mach. Intell.* 34, 564–573 (2012).
10. Tsai, D.-C. & Chen, H. H. Reciprocal focus profile. *IEEE Trans. Image Process.* 21, 459–468 (2011).
11. Malik, A. S. & Choi, T.-S. A novel algorithm for estimation of depth map using image focus for 3d shape recovery in the presence of noise. *Pattern Recogn.* 41, 2200–2225 (2008).
12. Malik, A. S. & Choi, T.-S. Consideration of illumination effects and optimization of window size for accurate calculation of depth map for 3d shape recovery. *Pattern Recognit.* 40, 154–170 (2007).
13. Fertiz, S., Puig, D. & Garcia, M. A. Analysis of focus measure operators for shape-from-focus. *Pattern Recognit.* 46, 1415–1432 (2013).
14. Huang, W. & Jing, Z. Evaluation of focus measures in multi-focus image fusion. *Pattern Recognit. Lett.* 28, 493–500 (2007).
15. Nayar, S. K. & Nakagawa, Y. Shape from focus. *IEEE Trans. Pattern Anal. Mach. Intell.* 16, 824–831 (1994).
16. Nayar, S. K. *Shape from focus* (CARNEGIE-MELLON UNIV PITTSBURGH PA ROBOTICS INST, Tech. Rep., 1989).
17. Subbarao, M. & Choi, T. Accurate recovery of three-dimensional shape from image focus. *IEEE Trans. Pattern Anal. Mach. Intell.* 17, 266–274 (1995).
18. Choi, T.-S. & Yun, J. Three-dimensional shape recovery from the focused-image surface. *Opt. Eng.* 39, 1321–1326 (2000).
19. Asif, M. & Choi, T.-S. Shape recovery from focus using multilayer feedforward neural networks. *IEICE Trans. Inf. Syst.* 83, 946–949 (2000).
20. Asif, M. & Choi, T.-S. Shape from focus using multilayer feedforward neural networks. *IEEE Trans. Image Process.* 10, 1670–1675 (2001).
21. Kim, H.-J., Mahmood, M. T. & Choi, T.-S. An efficient neural network for shape from focus with weight passing method. *Appl. Sci.* 8, 1648 (2018).
22. Yan, T., Wu, P., Qian, Y., Hu, Z. & Liu, F. Multiscale fusion and aggregation pcm for 3d shape recovery. *Inf. Sci.* 536, 277–297 (2020).
23. Ali, U., Praks, V. & Mahmood, M. T. Image focus volume regularization for shape from focus through 3d weighted least squares. *Inf. Sci.* 489, 155–166 (2019).
24. Ali, U. & Mahmood, M. T. 3d shape recovery by aggregating 3d wavelet transform-based image focus volumes through 3d weighted least squares. *J. Math. Imaging Vis.* 62, 54–72 (2020).
25. Ali, U., Lee, I. H. & Mahmood, M. T. Guided image filtering in shape-from-focus: A comparative analysis. *Pattern Recogn.* 111, 107670 (2021).
26. Ali, U. & Mahmood, M. T. Combining depth maps through 3d weighted least squares in shape from focus. In 2019 *International Conference on Electronics, Information, and Communication (ICEIC)*, 1–4 (IEEE, 2019).
27. Fan, T. & Yu, H. A novel shape from focus method combined with 3d steerable filters for improved performance on treating textureless region. *Opt. Commun.* 410, 254–261 (2018).
28. Mahmood, M. T. & Lee, I. H. Shape from focus based on 3d structure tensor using optical microscopy. *Micros. Res. Tech.* 83, 48–55 (2020).
29. Ma, Z., Kim, D. & Shin, Y.-G. Shape-from-focus reconstruction using nonlocal matting laplacian prior followed by mrf-based refinement. *Pattern Recognit.* 103, 107302 (2020).
30. Iang, H.-S., Muhammad, M. S. & Choi, T.-S. Optimizing image focus for shape from focus through locally weighted non-parametric regression. *IEEE Access* 7, 74393–74400 (2019).
31. Iang, H.-S., Yun, G., Mutahira, H. & Muhammad, M. S. A new focus measure operator for enhancing image focus in 3d shape recovery. *Micros. Res. Tech.* (2021).
32. Nayar, S. K., Watanabe, M. & Noguchi, M. Real-time focus range sensor. *IEEE Trans. Pattern Anal. Mach. Intell.* 18, 1186–1198 (1996).
33. Mutahira, H., Shin, V., Muhammad, M. S. & Shin, D. R. Sampling-noise modeling & removal in shape from focus systems through kalman filter. *IEEE Access* 9, 102520–102541 (2021).
34. Mannan, S., Malik, A. S. & Choi, T.-S. Reducing intricacy of 3d space for 3d camera. In *2008 IEEE International Symposium on Consumer Electronics*, 1–4 (IEEE, 2008).
35. Muhammad, M. S., Mutahira, H. & Choi, T.-S. Using a low pass filter to recover three-dimensional shape from focus in the presence of noise. *Int. J. Innov. Comput. Inf. Control* 8, 1 (2012).
36. Mutahira, H., Muhammad, M. S., Jaffar, A. & Choi, T.-S. Unorthodox approach toward microscopic shape from focus using optical microscopy. *Microsc. Res. Tech.* 76, 1–6 (2013).
37. Mahmood, M. T. & Choi, T.-S. Nonlinear approach for enhancement of image focus volume in shape from focus. *IEEE Trans. Image Process.* 21, 2866–2873 (2012).
38. Shin, S.-O. Multidirectional focus measure for accurate three-dimensional shape recovery of microscopic objects. *Microsc. Res. Tech.* 85, 940–947 (2022).
39. Mutahira, H., Ahmad, B., Muhammad, M. S. & Shin, D. R. Focus measurement in color space for shape from focus systems. *IEEE Access* 9, 103291–103310 (2021).
40. Subbarao, M. & Tyan, J.-K. Selecting the optimal focus measure for autofocusing and depth-from-focus. *IEEE Trans. Pattern Anal. Mach. Intell.* 20, 864–870 (1998).
41. Tsai, D.-C. & Chen, H. H. Focus profile modeling. *IEEE Trans. Image Process.* 25, 818–828 (2015).
42. Li, J. & Ding, F. Fitting nonlinear signal models using the increasing-data criterion. *IEEE Signal Process. Lett.* 29, 1302–1306 (2022).
43. Jang, H.-S., Muhammad, M. S., Yun, G. & Kim, D. H. Sampling based on Kalman filter for shape from focus in the presence of noise. *Appl. Sci.* 9, 3276 (2019).
44. Jang, H.-S., Muhammad, M. S. & Choi, T.-S. Removal of jitter noise in 3d shape recovery from image focus by using kalman filter. *Microsc. Res. Tech.* 81, 207–213 (2018).
45. Lee, S.-A., Jang, H.-S. & Lee, B.-G. Jitter elimination in shape recovery by using adaptive neural network filter. *Sensors* 19, 2566 (2019).
46. Jang, H.-S., Muhammad, M. S. & Choi, T.-S. Rayes filter based jitter noise removal in shape recovery from image focus. *J. Imaging Sci. Technol.* 63, 20501–1 (2019).
47. Jang, H.-S., Muhammad, M. S. & Choi, T.-S. Optimal depth estimation using modified Kalman filter in the presence of non-gaussian jitter noise. *Microsc. Res. Tech.* 82, 224–231 (2019).
48. Jang, H.-S., Muhammad, M. S. & Kang, M.-K. Removal of non-gaussian jitter noise for shape from focus through improved maximum correntropy criterion Kalman filter. *IEEE Access* 8, 36244–36255 (2020).
49. Pollefeys, M., Van Gool, L. & Proesmans, M. Euclidean 3d reconstruction from image sequences with variable focal lengths. In *European Conference on Computer Vision*, 31–42 (Springer, 1996).
50. Li, J., Ding, F. & Hayat, T. A novel nonlinear optimization method for fitting a noisy gaussian activation function. *Int. J. Adapt. Control Signal Process.* 36, 690–707 (2022).
51. Islam, S. A. U. & Bernstein, D. S. Recursive least squares for real-time implementation [lecture notes]. *IEEE Control Syst. Mag.* 39, 82–85 (2019).
52. Ding, F., Liu, P.X. & Liu, G. Multiinnovation least-squares identification for system modeling. *IEEE Trans. Syst. Man Cybern. Part B (Cybern.)* 340, 767–778 (2010).
53. Sherman, J. & Morrison, W. J. Adjustment of an inverse matrix corresponding to a change in one element of a given matrix. *Ann. Math. Stat.* 21, 124–125 (1950).
54. Ding, F. & Chen, T. Performance bounds of forgetting factor least-squares algorithms for time-varying systems with finite measurement data. *IEEE Trans. Circuits Syst. I Regul. Pap.* 52, 555–566 (2005).
55. Ding, F., Shi, Y. & Chen, T. Performance analysis of estimation algorithms of nonstationary arma processes. *IEEE Trans. Signal Process.* 54, 1041–1053 (2006).
56. Ding, F., Chen, T. & Qu, L. Bias compensation based recursive least-squares identification algorithm for miso systems. *IEEE Trans. Circuits Syst. I Express Briefs* 53, 349–353 (2006).
57. Lu, M.-C. Computer Modeling and Simulation Techniques for Computer Vision Problems. Ph.D. thesis, State University of New York at Stony Brook (1993).
58. Minhas, R., Mohammed, A. A. & Wu, Q. J. Shape from focus using fast discrete curvelet transform. *Pattern Recogn.* 44, 839–853 (2011).
59. Shih, S.-O., Aziz, W., Banjar, A., Alamri, A. & Alqarni, M. Improving depth computation from robust focus approximation. *IEEE Access* 7, 20144–20149 (2019).
60. Minhas, R., Mohammed, A. A. & Wu, Q. J. An efficient algorithm for focus measurement computation in constant time. *IEEE Trans. Circuits Syst. Video Technol.* 22, 152–156 (2011).
61. Eskicioglu, A. M. & Fisher, P. S. Image quality measures and their performance. *IEEE Trans. Commun.* 43, 2959–2965 (1995).
62. Ghadiyaram, D. & Bovik, A. C. Massive online crowdsourced study of subjective and objective picture quality. *IEEE Trans. Image Process.* 25, 372–387 (2015).

**Acknowledgements**

This research was supported by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2020R1F1A1072332).

**Author contributions**

H.M. conceived the idea(s). H.M. and M.S.M designed and conducted the experiment(s), V.S. and U.P. analyzed the results, H.M. wrote the manuscript. All authors reviewed the manuscript.

**Competing interests**

The authors declare no competing interests.

**Additional information**

**Correspondence** and requests for materials should be addressed to U.P. or M.S.M.

**Reprints and permissions information** is available at www.nature.com/reprints.

**Publisher's note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.
