

Received June 17, 2021, accepted July 12, 2021, date of publication July 16, 2021, date of current version July 26, 2021.

Digital Object Identifier 10.1109/ACCESS.2021.3097814

# Sampling-Noise Modeling & Removal in Shape From Focus Systems Through Kalman Filter

HUSNA MUTAHIRA<sup>1</sup>, (Student Member, IEEE), VLADIMIR SHIN<sup>2</sup>,  
MANNAN SAEED MUHAMMAD<sup>1</sup>, (Senior Member, IEEE), AND DONG RYEOL SHIN<sup>3</sup>

<sup>1</sup>Department of Electrical and Computer Engineering, College of Information and Communication Engineering, Sungkyunkwan University, Natural Sciences Campus, Suwon 16419, South Korea

<sup>2</sup>Department of Information and Statistics, Research Institute of Natural Science, Gyeongsang National University, Jinju 52828, South Korea

<sup>3</sup>Department of Computer Science and Engineering, College of Computing and Informatics, Sungkyunkwan University, Natural Sciences Campus, Suwon 16419, South Korea

Corresponding authors: Dong Ryeol Shin (drshin@skku.edu) and Mannan Saeed Muhammad (mannan@skku.edu)

**ABSTRACT** Shape from Focus (SFF) is one of the passive techniques to recover the shape of an object under consideration. It utilizes the focus cue present in the stack of images, obtained by a single camera. In SFF when the images are acquired, the inter-frame distance, also known as the sampling step size, is assumed to be constant. However, in practice, due to mechanical constraints, sampling step size cannot remain constant. The inconsistency in the sampling step size causes the problem of jitter, and produces Jitter noise in focus curves. This Jitter noise is not visible in images, because each pixel in an image (of the stack) will be subjected to the same error in focus. Thus, traditional image denoising techniques will not work. This paper formulates a model of the Jitter noise, followed by the design of system and measurement models for Kalman filter. Then, the jittering problem for SFF systems is solved using the proposed filtering technique. Experiments are performed on simulated and real objects. Ten noise levels are considered for simulated, and four for real objects. RMSE and Correlation are used to measure the reconstructed shape. The results show the effectiveness of the proposed scheme.

**INDEX TERMS** Shape from focus, shape reconstruction, Jitter noise, Noise Modeling, Kalman filter.

## I. INTRODUCTION

Three-dimensional shape recovery using two-dimensional images is a well-established research problem in computer vision applications, robot and machine vision, bio-informatics, medical imaging, consumer cameras, microscopy, and so forth [1]–[6]. In recent years, many techniques have been proposed to recover depth maps from acquired images, as natural scenes under different conditions produce different cues [7]. These cues are distinguished from each other, and can be measured, depending on various factors. One of these cues is *focusing* that is measured by determining the blur-degree of the image. The method by which object shapes within scenes are estimated, by accommodating focus cues by means of fixed-axis-multiple-images, is referred to as shape from focus (SFF).

This area of research has progressed extensively in recent years. All of SFF methods broadly consist of three main steps; image acquisition, focus measure (FM) application,

The associate editor coordinating the review of this manuscript and approving it for publication was Liang Hu<sup>1</sup>.

and shape improvement techniques, which are discussed briefly below.

Shape from Focus systems are modeled by the simple lens equation, given as:

$$\frac{1}{f} = \frac{1}{u} + \frac{1}{v}, \quad (1)$$

where  $f$  is the focal length of the imaging device,  $u$  is the distance of the object point from the imaging device, and  $v$  is the position of the object point where it is best focused by the lens. Fig. 1 illustrates this.

In SFF, the image stack is acquired by manipulating one of the factors of (1), while keeping the other two factors constant. Conventionally, the images are acquired by changing  $u$  of the system. An example of such type of system is an optical microscope. In other systems, focus settings (focal length  $f$ ) can also be changed to capture images. Examples of these systems can be found in [8]. Either  $f$  or  $u$  is changed in small steps, and an image (of dimensions  $l \times m$ ) is obtained, and stored in the image stack, giving the total number of images as  $n$ . Changing  $v$  for image acquisition

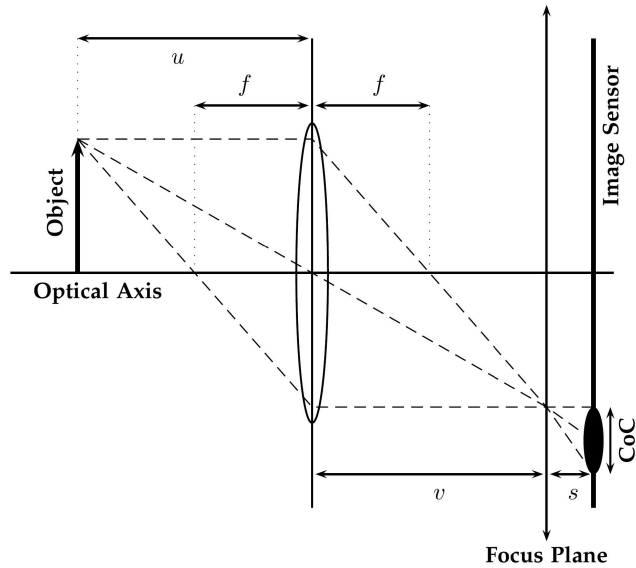


FIGURE 1. Focusing via Gaussian lens law.

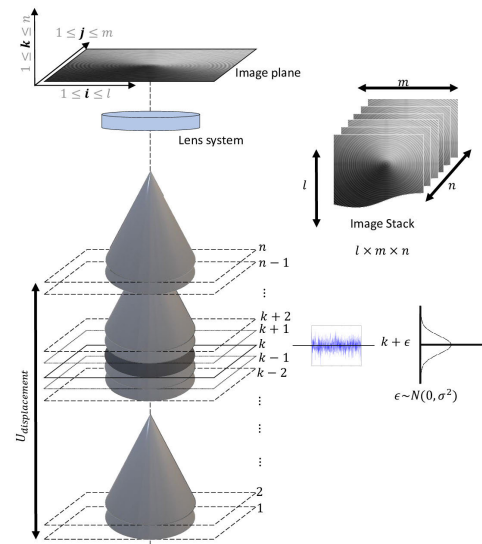


FIGURE 2. Image acquisition in SFF systems.

is quite challenging, or mostly not feasible [9]. However, whatever factor is manipulated, the magnification of the imaging system should remain constant, while the depth of field should be as shallow as possible [10].

When all the images are acquired, the result is an image stack  $I$  of dimensions  $l \times m \times n$ , and each 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 in Fig. 2. The number of images ( $n$ ) is given by:

$$n = \frac{U_{displacement}}{\Delta}, \quad (2)$$

where  $\Delta$  is the sampling step size. The step size expression for change in  $u$  is provided in [9]. The main idea of SFF is to estimate the shape of the object in consideration using the focus cues present in the image stack. The sharpness of focus in an image is measured by a sharpness criterion, called the focus measure (FM) operator. After the image stack is obtained, the FM is applied to each pixel, to measure the amount of focus each pixel possesses, by the following:

$$\Theta_{i,j}(k) = \mathbb{F} \{P_{i,j}(k)\}, \quad (3)$$

where  $\mathbb{F}$  is the FM transformation of pixel  $P_{i,j}(k)$  to obtain the focus value  $\Theta_{i,j}$  in the  $k^{th}$  image, and represents the focus behavior or (in other words) the focus curve of the pixel. There are numerous FMs proposed in the literature, summarized by [8], each designed to suppress the out-of-focus regions, and enhance the in focus points in each image. For example, Sum of Modified Laplacian (SML) utilizes squares of the second derivative of the images, Tenengrad or Tenenbaum (TEN) utilizes the first derivative of the image, and Gray Level Variance (GLVA) uses statistical method to compute variance as focus measurement, whereas, Image Curvature (CURV) calculates image surface curvature. There are other FMs such as Image Contrast (CONT), 3D Laplacian

(GRA-3), Discrete Cosine Transform Energy (DCTE), Spatial Frequency (SFRQ), Sum of Wavelet Coefficients (WAVS), and Steerable Filters (SFIL). The key role 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.

Conventionally, the initial depth map  $D_{i,j}$  can be obtained by maximizing the focus curve along the optical axis, and obtaining the value of  $k$  where  $\Theta_{i,j}(k)$  is the maximum according to:

$$D_{i,j} = \arg \max_k (\Theta_{i,j}(k)). \quad (4)$$

When the images are acquired, the shape of the object is discretized into image frames, causing loss of information between two consecutive frames. To address this issue, many techniques have been proposed in the literature. The traditional techniques used SML as focus measure and apply Gaussian interpolation technique to compute intra-frame values for better focus [11], [12]. The other concept of focused-image-surface and curved-focused-image-surface [13], [14], utilizes piecewise curved surface approximation. Alternatively, Neural Networks and Deep Neural Networks have also been employed [2], [15], [16].

Kim et al. in [17] proposed a method to improve the efficiency of neural networks by introducing a weight passing method. Muhammad & Choi in [10] proposed a method based on Bezier Surface approximation. Ali et al. in [18] increased the accuracy of 3D shapes by applying 3D weighted least squares to enhance image focus volume. Ali et al. in [19] also used the wavelet transform method to improve shape reconstruction. Guided image filtering for depth enhancement in SFF is proposed by [20]. Also, Ali et al. in [21] recovered several 3D shapes by applying different FMs, and combining their results into one final shape. Fan et al. in [22] used the combination of 3D steerable filters on treating texture-less regions. Since the reconstructed 3D shape quality in SFF depends on the applied FM, another FM based on the analysis

of 3D structure tensor of the image sequence is proposed by [23]. Ma *et al.* in [24] proposed a method for depth reconstruction that utilized non-local matting Laplacian along with Markov Random Field. Yan *et al.* [25] utilized pulsed coupled neural network to aggregate shape using focus. Jang *et al.* in [26] proposed shape optimization through non-para-metric regression.

The paper is structured as follows. Focus measurement and focus curve models are discussed in next section followed by motivation of the work. Section IV addresses Jitter noise Modeling. Section V proposed the methodology, while Section VI presents the Results and Discussion. Section VII then concludes the work.

**II. FOCUS MEASUREMENTS & FOCUS CURVE MODELS**

The focus curves or the focus behavior (of every individual pixel) depend on the FM used, the nature of cue the FM utilizes, the camera (imaging device) parameters, and most importantly, the image texture around that object point [8], [9]. If the images are acquired properly, then these focus curves are bell-shaped [12].

To approximate these bell-shaped focus curves, three models are used, namely, Gaussian Model [11], Lorentzian-Cauchy Model [9], and Quadratic Model [27].

The Gaussian Model is given by:

$$\Theta_G(k) = A_G \exp \left[ -\frac{(k - B_G)^2}{2C_G^2} \right], \tag{5}$$

the Lorentzian-Cauchy Model is given by:

$$\Theta_L(k) = A_L \frac{B_L^2}{B_L^2 + (k - C_L)^2}, \tag{6}$$

and the Quadratic Model is given by:

$$\Theta_Q(k) = A_Q k^2 + B_Q k + C_Q, \tag{7}$$

where the *A*s, *B*s and *C*s are the parameters of each model. The unification of these models into Quadratic Model have been provided by [27]. If the logarithmic transformation is applied to (5), and the equations are simplified, it transforms to (7) [27], as follows:

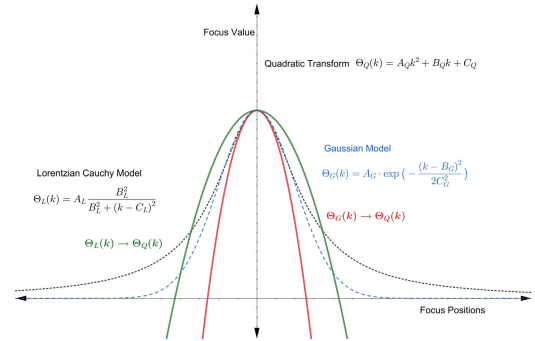
$$\log(\Theta_G(k)) \xrightarrow{\Theta_G \rightarrow \Theta_Q} \log(A_G) - \frac{1}{2C_G^2}(k^2 + B_G^2 - 2B_G k). \tag{8}$$

Similarly, the reciprocal transformation, when applied to (6) can result in (7) after simplification, as follows:

$$\frac{1}{\Theta_L(k)} \xrightarrow{\Theta_L \rightarrow \Theta_Q} \frac{1}{A_L B_L^2} (B_L^2 + k^2 + C_L^2 - 2C_L k). \tag{9}$$

Fig. 3 shows the transformation of the Gaussian and Lorentzian-Cauchy Models to the Quadratic Model.

This paper utilizes the Quadratic model (given in (7)) to model the Jitter noise in SFF systems in the next section. The Quadratic model provides computational advantage over Gaussian and Lorentzian-Cauchy models, due to its simplicity and robustness [27].



**FIGURE 3. Quadratic transformation of Gaussian and Lorentzian-Cauchy models.**

**III. MOTIVATION**

In SFF, when the shape is discretized into image frames by sampling the object in scene, the step size for sampling is presumed constant [9]. Although shape from focus has been thoroughly investigated in recent years, there still exist several insufficiently solved problems that impact the performance of the system. One of these problems is the unstable or non-constant sampling step size. This can be due to the mechanical structure of the imaging device and lens-focusing methods. The resultant variation in the amplitude of the signal due to instability in sampling step size is referred to as Jittering or Jitter noise.

Jang *et al.* in [28] proposed the removal of Jitter noise using Kalman Filter. Since then, many variants of their method have been proposed [29]–[33]. However, all of their methods used scalar-models for Kalman filter (i.e., the system matrix was taken as 1), and ignored the dynamic nature of focus cues. For each step, multiple images were acquired to eliminate the Jitter i.e., if there were *n* images (of dimensions *l* × *m*) in the stack, they required 100 samples for each step, and therefore, *n* × 100 samples were required for each focus curve. This increases the complexity of the system, and huge computational cost has to be paid. It also impacts the practical use of their methods. Also, Jang *et al.* in [28]–[33] considered only symmetric bell-shaped distributions for vibrational noise in translational stage, and their designed measurement model measures only a constant (each step position *k*). However, in such case taking the mean of the measurement values on every step position *k* can provide the similar results. The authors have also considered the Jitter noise to have Normal and Levy distributions [28], [31]; however, in practice, this resultant Jitter noise due to the vibrational noise does not follow (Normal or Levy) symmetric bell-shaped distributions.

In this paper, the nature of Jitter noise has been studied and necessary conditions for approximating this noise are proposed and discussed. Jitter noise in SFF systems is a position dependent noise and varies according to the focus position. The manuscript models Jitter noise and conclude that it follows gamma ( $\Gamma$ ) distribution (a non-symmetric distribution) with a constant mean and position dependent variance (discussed in Section IV). A Kalman filter is then designed for removing this noise from SFF systems.

The system matrix is formulated using the Taylor series, followed by explanation and design of the measurement model. The shape recovery expression is then provided. In the proposed scheme, a single measurement is taken for each step. Thus, for  $n$  images (of dimensions  $l \times m$ ) in the image stack, the proposed method requires only  $n$  samples for each focus curve, and utilizes 100 times less images as compared to previous methods, providing better shape recovery results in terms of correlation and RMSE (provided in Section VI).

#### IV. JITTER MODELING IN SHAPE FROM FOCUS SYSTEMS

In SFF systems, jitter occurs when there is uncertainty or unevenness in the step size of  $u$  or  $f$ . In this section, we discuss the step size in both situations of image acquisition i.e., change in object distance from the lens ( $\Delta u$ ), and change in focal length of the imaging device system ( $\Delta f$ ).

After this, we discuss types of Jitter, followed by the proposed model for Jitter noise. The next section utilizes this proposed Jitter noise model and Kalman filter to remove the effects of jittering on focus curves.

##### A. STEP SIZE IN SFF IMAGE ACQUISITION

The step size expression for  $u$  is provided by [9]. In their system, the object is moved towards (or away from) the imaging device in small constant steps of  $\Delta u$ , by keeping the focal length and magnification constant, and also keeping the depth of field as shallow as possible. Their simplified expression for  $\Delta u$  (step size) is provided as follows:

$$\Delta u = \frac{DoF}{4 \times \rho}, \quad (10)$$

where  $DoF$  is the depth of field of the system, and  $\rho = 2.9957$ , as provided by [9]. Equation (10) provides the maximum limit for  $\Delta u$ . The ideal example of such systems is an optical microscope.

The images can also be acquired by changing the focal length of the system in small, constant increments of  $\Delta f$  [34]. In this type of image acquisition for SFF, the object is held static in front of the imaging device, and the device focal length is changed. Mostly, auto-focusing algorithms utilize this type of technique for searching for the best focal lens position for a single point. This can also be used for depth and shape estimation of the object under consideration [35].

In both the above cases, an image is stored at every step to obtain a stack of images, as discussed in Section I.

##### B. MODELING JITTER IN SFF

To model the jitter, consider again the Quadratic function, as follows:

$$g(k) = a_2 k^2 + a_1 k + a_0, \quad (11)$$

where  $a_2$ ,  $a_1$  and  $a_0$  are the equation parameters,  $g(k)$  is the quadratic function, and  $1 \leq k \leq n$  represents the sample points of this function. The step size is  $\Delta k = k - (k - 1)$ , and is considered as 1.

To model Jitter noise, consider the uncertainty in step size as,  $\epsilon \sim N(0, \sigma_\epsilon^2)$ , then (11) can be written as follows:

$$g(k + \epsilon) = a_2(k + \epsilon)^2 + a_1(k + \epsilon) + a_0. \quad (12)$$

Rewriting (12), by expanding the squared terms and simplifying using the Taylor series, the following equation is derived:

$$g(k + \epsilon) = a_2 k^2 + a_1 k + a_0 + \epsilon(2a_2 k + a_1) + \epsilon^2 a_2. \quad (13)$$

Using (11) and (13), following is obtained:

$$g(k + \epsilon) = g(k) + \epsilon g'(k) + \epsilon^2 \frac{g''(k)}{2}. \quad (14)$$

Equation (14) shows that the noise on the RHS of the equation is multiplied to the first and second derivatives of the function, concluding that the Jitter noise in SFF systems depends on the slope and concavity of the focus curves. If  $\epsilon$  is Normal ( $N(0, \sigma_\epsilon^2)$ ),  $\epsilon^2$  will follow a chi-square distribution. Equation (14) can be rewritten as:

$$g(k + \epsilon) = g(k) + \eta_{k_N} + \eta_{k_\chi}, \quad (15)$$

where  $\eta_{k_N}$  and  $\eta_{k_\chi}$  are given by the following:

$$\eta_{k_N} = \epsilon g'(k) = \epsilon(2a_2 k + a_1), \quad (16)$$

and,

$$\eta_{k_\chi} = \epsilon^2 \frac{g''(k)}{2} = \epsilon^2 a_2, \quad (17)$$

where  $\eta_{k_N}$  is normally distributed with mean  $\mu_N = 0$  and variance  $\sigma_N^2 = (2a_2 k + a_1)^2 \sigma_\epsilon^2$ . Meanwhile,  $\eta_{k_\chi}$  follows gamma ( $\Gamma$ ) distribution, with mean  $\mu_\chi = a_2 \sigma_\epsilon^2$ , and variance  $\sigma_\chi^2 = 2a_2^2 \sigma_\epsilon^4$ . Therefore, the total resultant noise  $\eta$  will have mean  $\mu_\eta = a_2 \sigma_\epsilon^2$  and variance as  $\sigma_\eta^2 = (2a_2 k + a_1)^2 \sigma_\epsilon^2 + 2a_2^2 \sigma_\epsilon^4$ . Derivation of the mean ( $\mu_\eta$ ) and variance ( $\sigma_\eta^2$ ) is based on the standard manipulations with probability theory formulas and properties of expectations.

The value of the variance of  $\eta_{k_N}$  ( $\sigma_N$ ) will be different at every  $k^{\text{th}}$  step, and will become zero at:

$$k = -\frac{a_1}{2a_2}. \quad (18)$$

The direction of  $\eta_{k_\chi}$  will always be towards the concavity of the function.

The range ( $R_\chi$ ) of  $\eta_{k_\chi}$  depends on  $g''(k)$ . If  $g''(k) > 0$ , then  $0 < \epsilon_\chi < \infty$ ; similarly, if  $g''(k) < 0$ , then  $-\infty < \epsilon_\chi < 0$ ; and if  $g''(k) = 0$ , the effect of this noise is zero. However, the physical limitations and restrictions of the imaging device restrict  $\epsilon_\chi < \Delta k$ . The noise factor of  $\eta_{k_\chi}$  will remain throughout the function  $g(k + \epsilon)$ ; however, the sign of  $\eta_\chi$  will depend on the sign of  $g''(k)$ , and will always be towards the concavity of the function.

For the values of  $k$  other than (or near to) the value given in (18),  $\eta_{k_N}$  will be more significant for  $\eta_{k_\chi}$ . But as the function approaches the value given in (18), the effect of  $\eta_{k_N}$

diminishes, while the contribution of  $\eta_{k\chi}$  becomes significant. However, if the variance of  $\epsilon$ , given by:

$$\sigma_\epsilon \leq \frac{\Delta}{6}, \tag{19}$$

is chosen, the effect of  $\eta_\chi$  can be ignored, making  $\eta_N$  the only contribution to the noise, and the resultant noise will be Normal. However, if the condition given in (19) is violated, then  $\eta_{k\chi}$  can play a significant role, and thus cannot be ignored. The combination of both  $\eta_{kN}$  and  $\eta_{k\chi}$  results in a non-Normal noise.

### V. THE PROPOSED METHODOLOGY

In the previous section, Jitter noise is modeled, and explained in detail. In this section, the proposed method is presented. The proposed scheme can be applied in two ways, before FM (as pre-FM application, i.e., on pixel curves  $P_{i,j}(k)$ ), or after FM (as post-FM application, i.e., on focus curves  $\Theta_{i,j}(k)$ ). To fully remove the Jitter noise from the pixel/focus curves, the Kalman Filter is designed as follows:

#### A. KALMAN FILTER DESIGN

To model and design the Kalman filter, the proposed method in this manuscript utilizes the cubic equation to design the system, and measurement equations for the filter to remove the Jitter noise from the pixel/focus curves. Although quadratic equation can also be utilized, cubic equation gives the system model more robustness and flexibility, since the cubic equation is of higher degree than quadratic equation. The system and measurement models are derived in the following sections.

##### 1) SYSTEM MODEL

In order to derive the system model for Kalman filter application, the cubic equation is considered and is given as:

$$h_k = a_3k^3 + a_2k^2 + a_1k + a_0, \tag{20}$$

where  $a_0, a_1, a_2$ , and  $a_3$  are the coefficients of the equation, and  $h_k$  is the cubic function. The first, second, and third derivatives of (20) are:

$$\begin{aligned} h'_k &= 3a_3k^2 + 2a_2k + a_1, \\ h''_k &= 6a_3k + 2a_2, \\ h'''_k &= 6a_3. \end{aligned} \tag{21}$$

Using the Taylor series again, the equation for  $h_{k+1}$  is written as:

$$h_{k+1} = a_3(k+1)^3 + a_2(k+1)^2 + a_1(k+1) + a_0. \tag{22}$$

By expanding the powers in (22) and rearranging the following is obtained:

$$\begin{aligned} h_{k+1} &= a_3k^3 + a_2k^2 + a_1k + a_0 + 3a_3k^2 \\ &\quad + 2a_2k + a_1 + 3a_3k + a_2 + a_3. \end{aligned} \tag{23}$$

Utilizing (20) and the set of equations in (21); (23) can be rewritten as:

$$h_{k+1} = h_k + h'_k + \frac{1}{2}h''_k + \frac{1}{6}h'''_k. \tag{24}$$

Using (24), and repeating the process for  $h'(k+1)$ ,  $h''(k+1)$ , and  $h'''(k+1)$ , a similar set of equations can be obtained for the set of  $h_k$ , and its derivatives:

$$\begin{aligned} h_k &= h_{k-1} + h'_{k-1} + \frac{1}{2}h''_{k-1} + \frac{1}{6}h'''_{k-1}, \\ h'_k &= h'_{k-1} + h''_{k-1} + \frac{1}{2}h'''_{k-1}, \\ h''_k &= h''_{k-1} + h'''_{k-1}, \\ h'''_k &= h'''_{k-1}. \end{aligned} \tag{25}$$

Thus, utilizing this set of equations in (25), the system equations for the Kalman filter can be written as:

$$X_k = AX_{k-1} + \omega_k, \tag{26}$$

where  $A$  is the system state matrix,  $X_k$  is the state vector. The predicted state noise at  $k$  is given by  $\omega_k \sim N(0, Q)$ , and  $h$  represents the focus curve ( $\Theta_{i,j}(k)$ ) or pixel curve ( $P_{i,j}(k)$ ) values. The manuscript assumes that there is no system noise and the only noise present is due to jitter in measurements, therefore,  $Q$  (the process covariance matrix) is assumed to have a very small value but not 0. Then, state vector  $X_k$ , and system matrix  $A$  are given by:

$$X_k = \begin{bmatrix} h_k \\ h'_k \\ h''_k \\ h'''_k \end{bmatrix}, \quad A = \begin{bmatrix} 1 & 1 & \frac{1}{2} & \frac{1}{6} \\ 0 & 1 & 1 & \frac{1}{2} \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

The system covariance equation is provided by:

$$\Psi_k^- = A\Psi_{k-1}A^T + Q, \tag{27}$$

where  $\Psi_k^-$  is the predicted (estimated) covariance matrix at  $k$ ,  $\Psi_{k-1}$  is the predicted covariance matrix at  $k-1$ , and  $Q$  is the process covariance matrix, respectively.

##### 2) MEASUREMENT MODEL

The next step in the proposed methodology is to design the measurement model for the Kalman filter. For this purpose, (20) is rewritten using the Taylor series and the steps explained in Section IV, as follows:

$$h_{k+\epsilon} = a_3(k+\epsilon)^3 + a_2(k+\epsilon)^2 + a_1(k+\epsilon) + a_0, \tag{28}$$

by rearranging, and utilizing (21), the following is obtained:

$$h_{k+\epsilon} = h_k + \epsilon h'_k + \frac{\epsilon^2}{2}h''_k + \frac{\epsilon^3}{6}h'''_k. \tag{29}$$

Utilizing the condition explained in (19), the  $\epsilon^2$  and  $\epsilon^3$  factors can be ignored, resulting in a simplified measurement model, as follows:

$$Y_k = CX_k + \eta_k, \tag{30}$$

**TABLE 1. Summary of Kalman Filter equations for proposed scheme (as pre/post FM application).**

Equation Name	Pre-FM Application	Post-FM Application
$h_k$ represents	$P_{i,j}(k)$	$\Theta_{i,j}(k)$
State Vector	$X_k = [P_{i,j}(k), P'_{i,j}(k), P''_{i,j}(k), P'''_{i,j}(k)]^T$	$X_k = [\Theta_{i,j}(k), \Theta'_{i,j}(k), \Theta''_{i,j}(k), \Theta'''_{i,j}(k)]^T$
System Model	$X_k = AX_{k-1} + \omega_k; \quad \omega_k \sim N(0, Q)$	
Measurement Model	$Y_k = CX_k + \eta_k; \quad \eta_k \sim N(0, R)$	
Kalman Filter Equations		
Predicted Estimate	$\hat{X}_k^- = A\hat{X}_{k-1}$	
Predicted Covariance	$\Psi_k^- = A\Psi_{k-1}A^T + Q$	
Filter Gain	$K_k = \Psi_k^- C^{-1} [C\Psi_k^- C^T + R]^{-1}$	
Updated Estimate	$\hat{X}_k = \hat{X}_k^- + K_k[Y_k - C\hat{X}_k^-]$	
Updated Covariance	$\Psi_k = (I - K_k C)\Psi_k^-$	
Final Estimate represents	$\hat{P}_{i,j}(k)$	$\hat{\Theta}_{i,j}(k)$

where  $Y_k$  represents the measurements of the pixel curve or focus curve of pixel  $P_{i,j}(k)$  after FM application at the  $k^{th}$  image,  $C$  is the state measurement matrix (given as  $C = [1 \ 0 \ 0 \ 0]$ ) and  $\eta_k$  is the noise in measurement due to jitter, as modeled in Section IV.

As the filter can be applied in two ways:

- when applied before FM (as pre-FM application), i.e., on pixel curves, then:  $Y_k = P_{i,j}(k)$ ,
- or when applied after FM (as post-FM application), i.e., on focus curves, then:  $Y_k = \Theta_{i,j}(k)$ .

**3) UPDATED STATES AND KALMAN GAIN**

The Kalman gain is computed on every step using (31) as:

$$K_k = \Psi_k^- C^{-1} [C\Psi_k^- C^T + R]^{-1}, \tag{31}$$

where  $K_k$  is the Kalman gain at  $k$ , and  $R$  is the measurement covariance matrix. The optimal state estimate is computed using the following:

$$\begin{aligned} \hat{X}_k &= \hat{X}_k^- + K_k[Y_k - C\hat{X}_k^-], \\ \Psi_k &= (I - K_k C)\Psi_k^-, \end{aligned} \tag{32}$$

where  $\hat{X}_k^- = A\hat{X}_{k-1}$ .

**B. SHAPE RECOVERY**

After the  $n^{th}$  step iteration for the focus curve is completed, the depth for every pixel is recovered to obtain the shape of the object under consideration. As presented in the previous section, the filter can be applied in two ways, pre- or post-FM application.

If the filter is applied as pre-FM application then before recovering the depth map, FM is applied on  $\hat{P}_{i,j}(k)$  to obtain  $\hat{\Theta}_{i,j}(k)$  using (3). However, if the filter application is post-FM, then the depth map can be recovered directly using  $\hat{\Theta}_{i,j}(k)$ .

For every object point  $i, j$  the coefficients of (20) (around  $k^*$ ) are estimated using the following:

$$\hat{M}_{i,j} = [\Xi_{i,j}(k^*)]^{-1} \Phi_{i,j}(k^*), \tag{33}$$

where  $k^*$  is the position where  $\hat{\Theta}_{i,j}(k)$  is maximum, and is obtained by the following:

$$k^* = \arg \max_k (\hat{\Theta}_{i,j}(k)). \tag{34}$$

In (33), the vector  $\hat{M}_{i,j}$  represents the collection of parameters of  $h_k$ ,  $\Phi_{i,j}(k^*)$  are the values of  $\hat{\Theta}_{i,j}(k)$  around  $k^*$ , and  $\Xi_{i,j}(k^*)$  is the coefficient matrix; all for  $i, j$  object point and defined as follows:

$$\begin{aligned} \hat{M}_{i,j} &= \begin{bmatrix} \hat{a}_0 \\ \hat{a}_1 \\ \hat{a}_2 \\ \hat{a}_3 \end{bmatrix}, \quad \Phi_{i,j}(k^*) = \begin{bmatrix} \hat{\Theta}_{i,j}(k^*) \\ \hat{\Theta}_{i,j}(k^* - 1) \\ \hat{\Theta}_{i,j}(k^* - 2) \\ \hat{\Theta}_{i,j}(k^* + 1) \end{bmatrix}, \\ \Xi_{i,j}(k^*) &= \begin{bmatrix} 1 & k^* & k^{*2} & k^{*3} \\ 1 & (k^* - 1) & (k^* - 1)^2 & (k^* - 1)^3 \\ 1 & (k^* - 2) & (k^* - 2)^2 & (k^* - 2)^3 \\ 1 & (k^* + 1) & (k^* + 1)^2 & (k^* + 1)^3 \end{bmatrix}. \end{aligned}$$

The refined and filtered depth ( $KD_{i,j}$ ) for every  $P_{i,j}$  is then recovered by:

$$KD_{i,j} = \begin{pmatrix} -\hat{a}_1 \\ -2\hat{a}_2 \end{pmatrix}_{i,j}. \tag{35}$$

**VI. RESULTS AND DISCUSSION**

This section analyzes the experimental results, and discusses them in detail. The section is divided into three subsections.

First, details of the experimental setup are provided, followed by the depth map and shape assessment criteria, and the metric measures used. Later the detail analysis of the affects of Jitter noise on SFF is provided at the end of the section.

TABLE 2. Summary of experimented objects.

Name	Object Type	Image Dimensions	Number of Images
Simulated Cone	Simulated	360 × 360	97
Real Cone	Real	200 × 200	97
Real Plane	Real	200 × 200	87
LCD-TFT	Real	200 × 200	60
Groove	Real	200 × 200	60
Coin	Real	200 × 200	68
Image-I	Real	200 × 200	60

A. EXPERIMENTAL SETUP

Experiments for shape reconstruction analysis are performed on seven objects. Table 2, provides a summary of the objects used in the 3D shape analysis. Ten simulated datasets of simulated cone are generated with different lens positions and Jitter noise levels using camera simulation software (AVS). The details of AVS are provided in [8], [36], [37]. The Matlab code used can be downloaded from [8]. All the datasets consist of 97 images with 360 × 360 pixels. The AVS software is provided with the depth map, texture image, and camera parameters. The texture map consists of concentric circles of two 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 the same. The difference in each dataset is the uncertainty in step size Δu to generate the sequences, in order to study the effect of Jitter on shape reconstruction. The values of variance σϵ in Δu are (0, 0.1, 0.25, 0.5, 0.75, 1.0, 1.25, 1.5, 1.75, and 2.0).

The real datasets contain real objects, Real Cone, Real Plane, LCD-TFT Filter, Groove, Coin, and Image-I. These image sequences were originally in gray-scale. Fig. 4 provides the ground truths of Simulated and Real cones. Fig. 5 shows the 10th frame of each image sequence. These image sequences have been widely used by many researchers including [19], [38]–[42].

Images of Real Cone were taken using the CCD camera system, [13], with dimensions of 200 × 200 × 97. The Real Plane image sequence is also obtained in a similar way, and contains 87 image frames each with a size of 200 × 200 pixels. Sixty images of the LCD-TFT filter were taken by the microscopic control system (MCS), with each image consisting of 300 × 300 pixels.

The coin sequence consists of magnified images of Lincoln’s head from the back of a US penny. The LCD-TFT filter images were microscopic images of an LCD color filter. These images were also obtained by means of MCS. The system consisted of a personal computer integrated with a frame grabber board (Matrox Meteor-II) and a CCD camera (SAMSUNG CAMERA SCC-341) mounted on a microscope (NIKON OPTIPHOT-100S). Computer software acquired the images by controlling the lens position through a stepper motor driver (MAC 5,000), possessing a 2.5 nm minimum step length. All the images being stored in sequence at every step were captured by varying the object plane.

The sequence of Image-I is the letter I engraved in a metallic surface. This sequence consists of 60 images, and

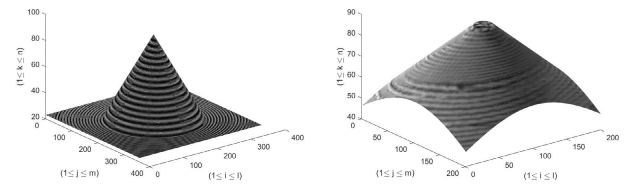


FIGURE 4. Ground truth of simulated Cone and real Cone.

was also obtained via the same system under magnification of 10×. The dimensions of this image sequence are 300 × 300 pixels.

The Groove image sequence is of a V-groove engraved in a metallic surface. The dimensions of this image sequence are 300 × 300 pixels, with 60 images.

B. METRIC MEASURES

The shape reconstruction quality is the characteristic that measures the perceived difference between the reconstructed shape and the ideal shape. As the difference increases, the quality of shape reconstruction reduces. In this article, the quality of the depth map obtained by using different focus measures under various levels of Jitter noise is analyzed. In the ideal case, the obtained depth map is indistinguishable from the original map, and the difference is zero, hence, the quality of the map is at its maximum. Several quality metrics have been provided in the previous literature [43]. In this manuscript, RMSE and correlation are used to compare the proposed method combined with various FM operators under different levels of Jitter noise.

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. A lower value of RMSE indicates better results, written as:

$$RMSE = \sqrt{\frac{1}{(l \times m)} \sum_i^l \sum_j^m (G_{i,j}^{true} - D_{i,j}^{obtained})^2} \quad (36)$$

where  $G^{true}$  is the ground truth,  $D^{obtained}$  is the obtained depth map, and  $l \times m$  are the dimensions of the depth maps. The higher the value of RMSE, the larger the error in shape reconstruction; for better results, the value should be close to zero.

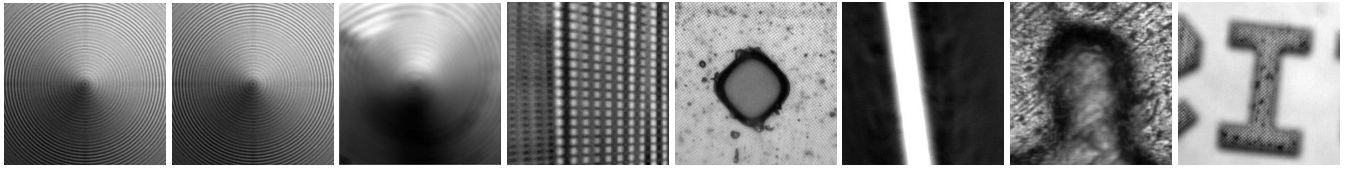
Correlation, or Pearson Correlation, is a linear relationship or similarity measure between two shapes [44], given by:

$$\rho_{GD} = \frac{cov(G^{true}, D^{obtained})}{\sigma_G \sigma_D} \quad (37)$$

where cov is the covariance, and  $\sigma_G^2$  and  $\sigma_D^2$  are the variances of  $G^{true}$  and  $D^{obtained}$ , respectively.

C. EXPERIMENTAL RESULTS AND DISCUSSION

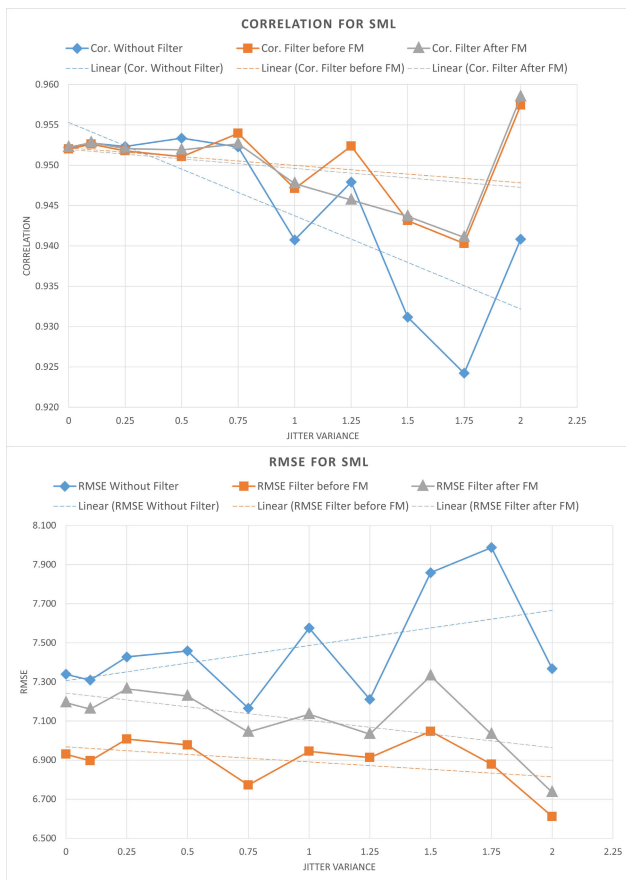
For experimentation, ten well-known FMs, namely, Sum of Modified Laplacian (SML), Tenenbaum (TENGL), Grey Level Variance (GLVA), Image Curvature (CURV), Image Contrast



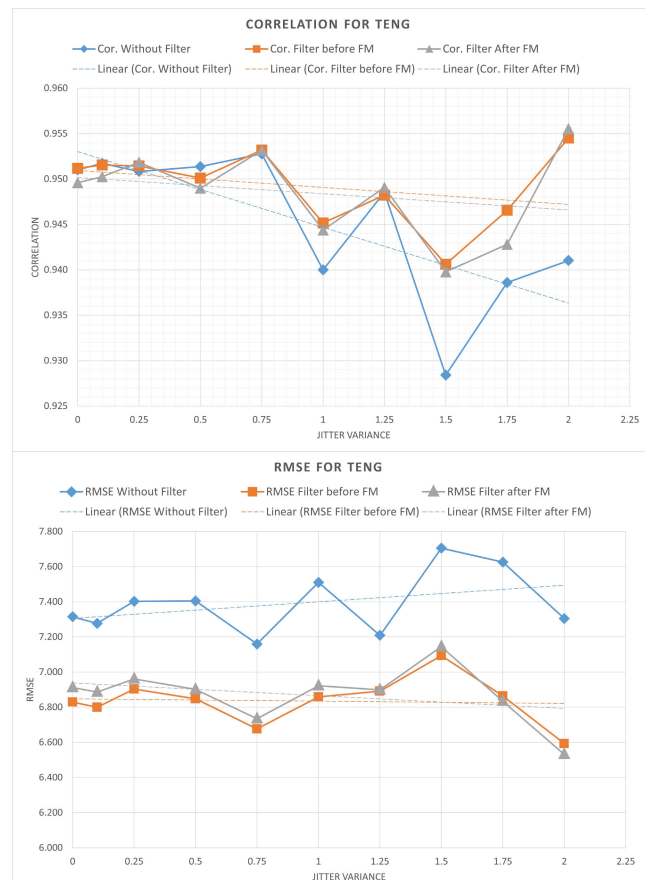
**FIGURE 5.** 10th frame of image sequence of each experimented object. (top row: right to left) Simulated Cone with  $\sigma_\epsilon = 0$  and  $\sigma_\epsilon = 2.0$ , Real Cone, Real Plane and LCD-TFT, Groove, Coin and Image-I.

**TABLE 3.** Comparison of proposed method with previous methods using SML, GLVA and TENG with Jitter noise ( $\sigma_\eta^2 = (2a_2k + a_1)^2\sigma_\epsilon^2 + 2a_2^2\sigma_\epsilon^4$ ),  $\sigma_\epsilon^2 = 0.1$ .

Methods	SML		GLVA		TENG		Iterations per step
	RMSE	Cor.	RMSE	Cor.	RMSE	Cor.	
Scalar KF	11.426	0.849	11.375	0.826	9.442	0.805	100
MKF	8.625	0.864	8.639	0.876	9.311	0.873	100
ANNF	7.689	0.955	8.145	0.949	8.507	0.927	100
MCKKF	7.470	0.956	7.957	0.949	8.420	0.928	100
IMCKKF	7.047	0.968	7.565	0.972	7.577	0.970	100
Proposed KF before FM	6.897	0.953	6.848	0.952	6.800	0.952	1
Proposed KF after FM	7.161	0.953	6.926	0.951	6.886	0.950	1



**FIGURE 6.** Correlation and RMSE using SML as FM with different values of variance of jitter noise  $\sigma_\eta^2 = (2a_2k + a_1)^2\sigma_\epsilon^2 + 2a_2^2\sigma_\epsilon^4$ ; without using Kalman filter, with filter as pre-FM step and filter as post-FM step. The dotted line shows the linear trend of the data.



**FIGURE 7.** Correlation and RMSE using TENG as FM with different values of variance of jitter noise  $\sigma_\eta^2 = (2a_2k + a_1)^2\sigma_\epsilon^2 + 2a_2^2\sigma_\epsilon^4$ ; without using Kalman filter, with filter as pre-FM step and filter as post-FM step. The dotted line shows the linear trend of the data.

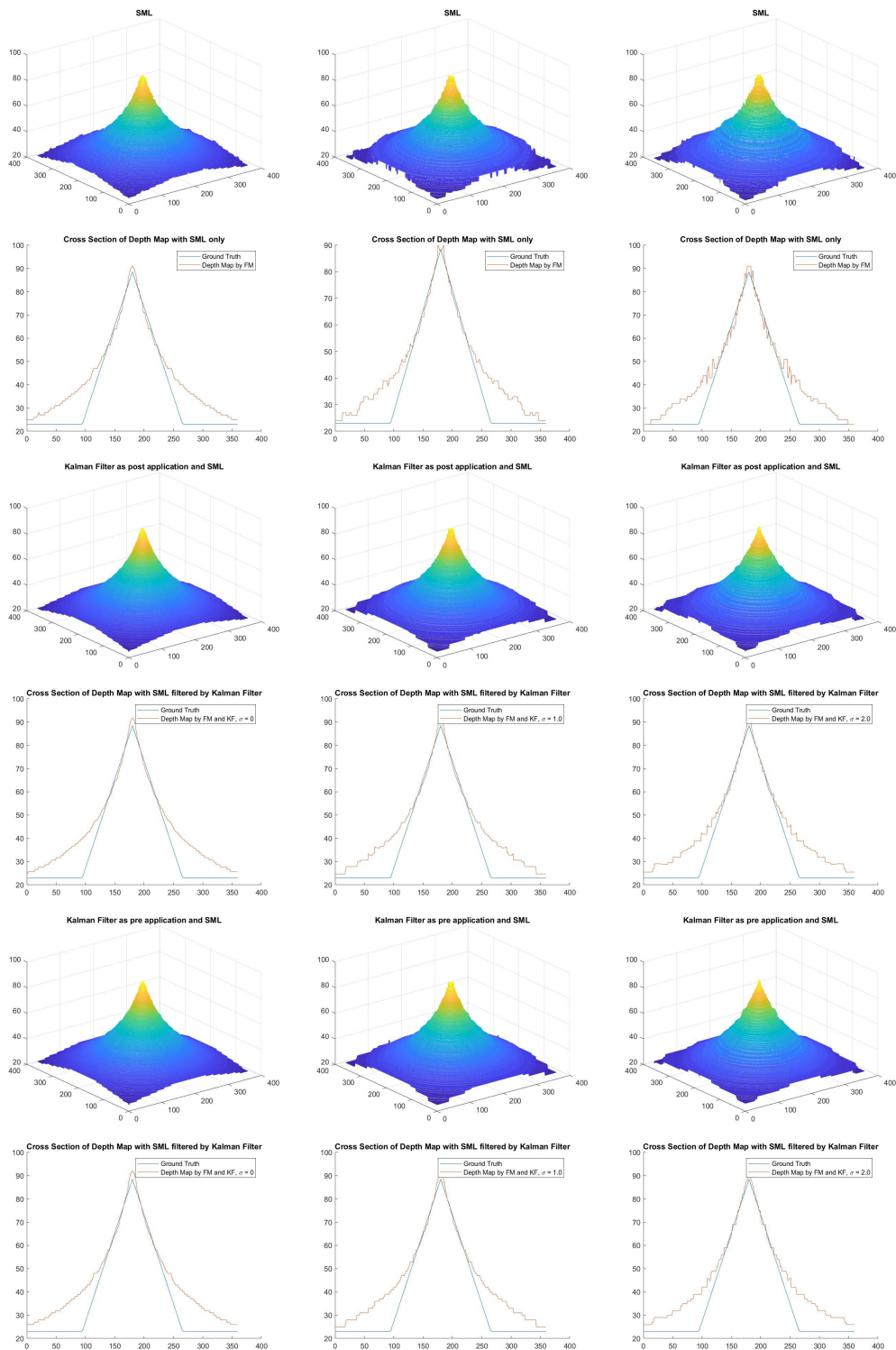
(CONT), 3D Laplacian (GRA3), DCT Energy Ratio (DCTE), Spatial Frequency (SFRQ), Sum of Wavelet Coefficients (WAVS), and Steerable Filters (SFIL) are used.

The experiments are first performed on the Simulated Cone image sequence with values of variance  $\sigma_\epsilon^2$  in  $\Delta u$  of (0, 0.1, 0.25, 0.5, 0.75, 1.0, 1.25, 1.5, 1.75, and 2.0), with all the ten

above-mentioned FMs. Tables 4 to 13 list the results of these experiments.

In Table 3 the proposed method is compared with previous methods provided by Jang *et al.* in [28]–[33]. The results are provided for  $\sigma_\epsilon^2 = 0.1$ . The first method is a scalar Kalman filter method, followed by scalar version of modified

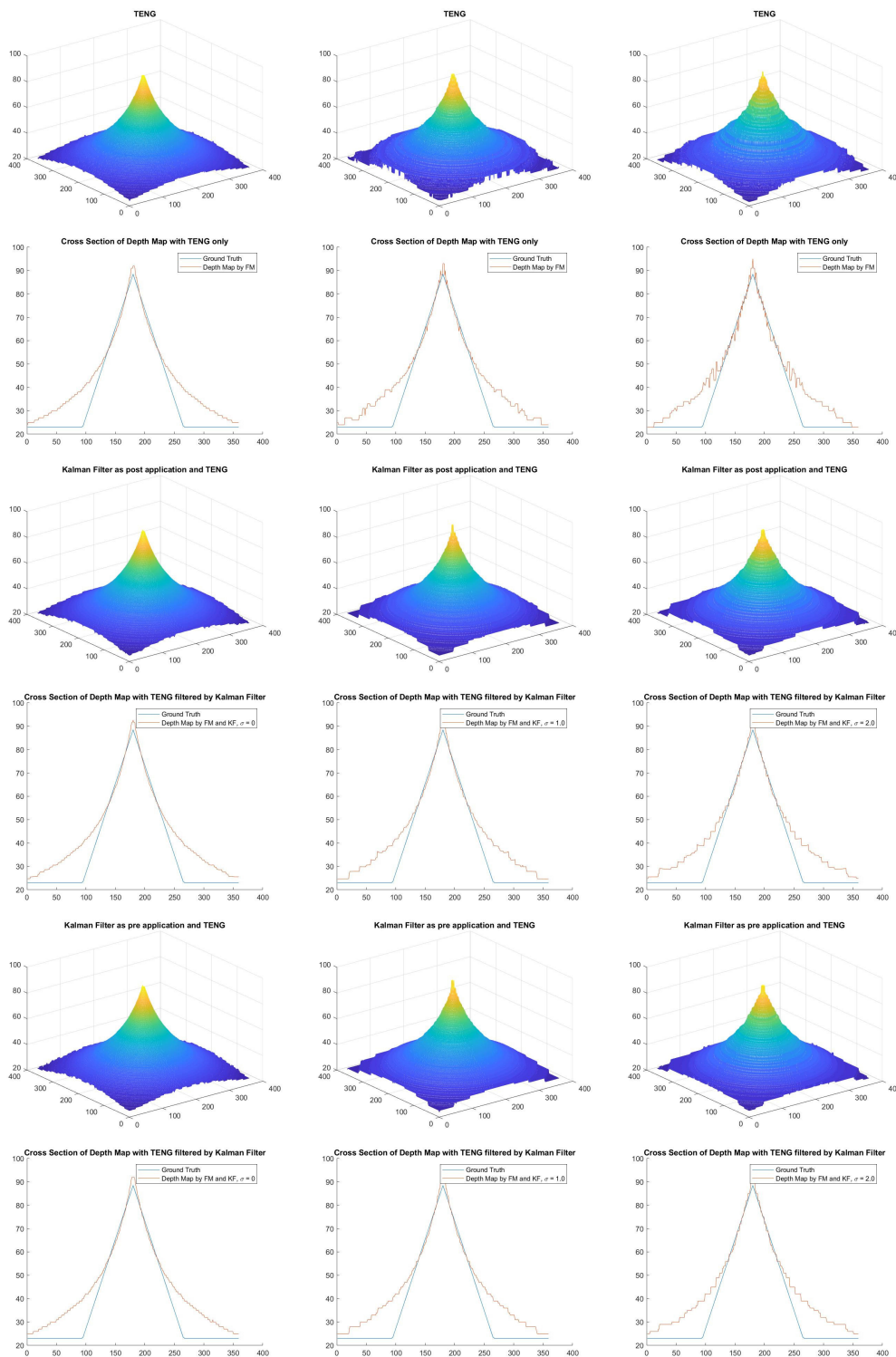




**FIGURE 8.** Shape reconstruction of Simulated Cone using SML as FM. (left-right) Jitter variance 0, 1.0 and 2.0 (top-bottom) SML only (3D shape and its cross-section), Kalman Filter before SML (3D shape and its cross-section), Kalman Filter after SML (3D shape and its cross-section).

Kalman filter method, ANNF method, MCCKF method and IMCKKF method. All of these methods use 100 iterations per step. RMSE and correlation results are provided in the table use SML, GLVA and TENG FMs. Results of proposed

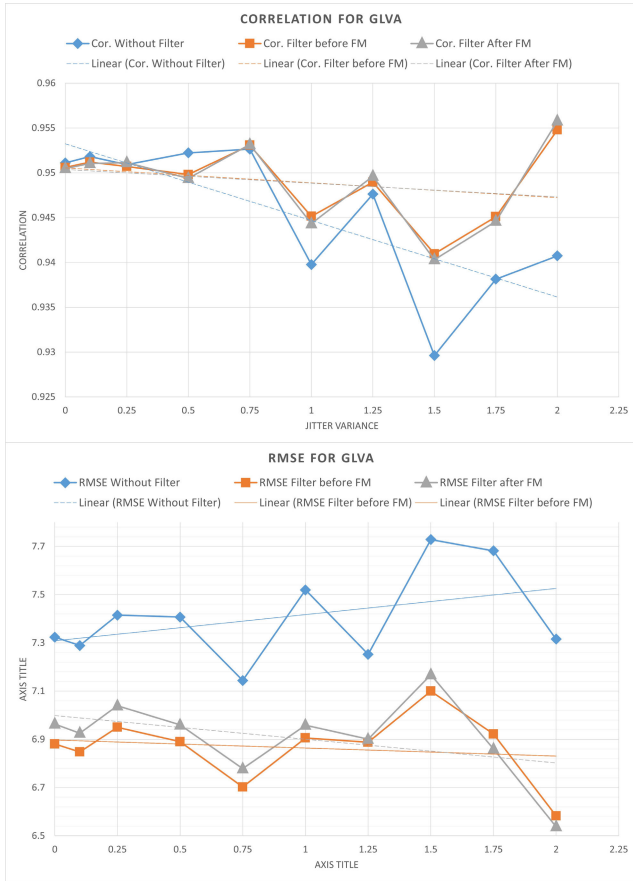
filtering technique (before and after FMs) are also shown. The proposed scheme utilizes only single measurement for each step. The RMSE results of proposed technique are better for all the FMs as compared to previous methods. Whereas,



**FIGURE 9.** Shape reconstruction of Simulated Cone using TENG as FM. (left-right) Jitter variance 0, 1.0 and 2.0 (top-bottom) TENG only (3D shape and its cross-section), Kalman Filter before TENG (3D shape and its cross-section), Kalman Filter after TENG (3D shape and its cross-section).

the correlation results of the proposed scheme are better or comparable to other methods. Only the correlation values of IMCCKF method are better than proposed method as they use more data, but their RMSE results are poor in comparison to

proposed scheme. This is due to the fact that the proposed scheme utilizes only the basic Kalman filter model. The results can be further improved if advance variants of Kalman filter are used with proposed scheme.



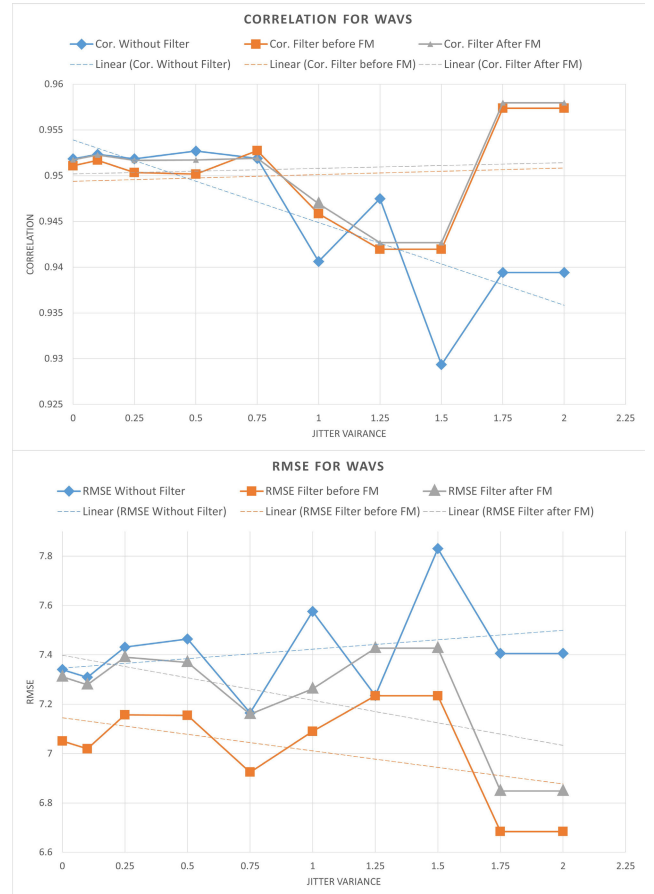
**FIGURE 10.** Correlation and RMSE using GLVA as FM with different values of variance of Jitter noise  $\sigma_\eta^2 = (2a_2k + a_1)^2\sigma_\epsilon^2 + 2a_2^2\sigma_\epsilon^4$ ; without using Kalman filter, with filter as pre-FM step and filter as post-FM step. The dotted line shows the linear trend of the data.

**TABLE 4.** RMSE and Cor. for shape reconstruction using SML with different levels of variance of Jitter noise ( $\sigma_\eta^2 = (2a_2k + a_1)^2\sigma_\epsilon^2 + 2a_2^2\sigma_\epsilon^4$ ).

$\sigma_\epsilon^2$	SML only		Kal. Fltr. before SML		Kal. Fltr. after SML	
	RMSE	Cor.	RMSE	Cor.	RMSE	Cor.
0	7.339	0.952	6.930	0.952	7.194	0.952
0.1	7.309	0.953	6.897	0.953	7.161	0.953
0.25	7.428	0.952	7.008	0.952	7.264	0.952
0.5	7.458	0.953	6.978	0.951	7.227	0.952
0.75	7.165	0.952	6.773	0.954	7.044	0.953
1	7.576	0.941	6.945	0.947	7.135	0.948
1.25	7.211	0.948	6.913	0.952	7.033	0.946
1.5	7.859	0.931	7.047	0.943	7.331	0.944
1.75	7.987	0.924	6.879	0.940	7.033	0.941
2	7.367	0.941	6.611	0.957	6.735	0.958

For further experimentation, each level of noise and FM, three scenarios are considered: FM only (i.e., without proposed filter), proposed filtering as a pre-processing step, (i.e. applied before FM), and proposed filtering as a post-processing step, (i.e. applied after FM). Each table has seven columns, with the  $\sigma_\epsilon^2$  in the first column, followed by one column each for correlation and RMSE for each scenario.

Table 4 along with Fig. 6 show the results of shape reconstruction for Simulated Cone under various levels of Jitter noise. The solid lines with markers show the correlation and RMSE values, whereas the dotted lines show



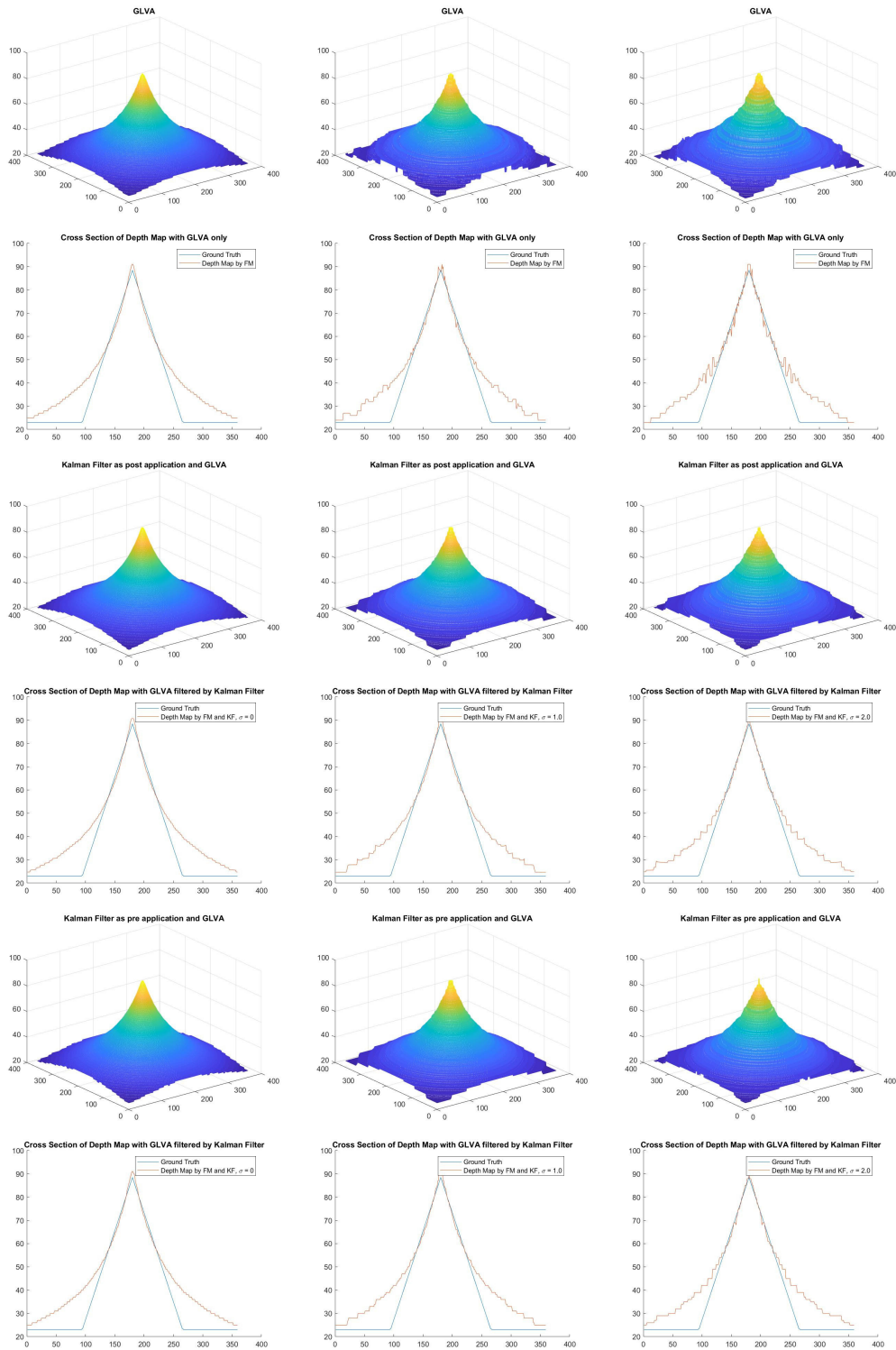
**FIGURE 11.** Correlation and RMSE using WAVS as FM with different values of variance of Jitter noise  $\sigma_\eta^2 = (2a_2k + a_1)^2\sigma_\epsilon^2 + 2a_2^2\sigma_\epsilon^4$ ; without using Kalman filter, with filter as pre-FM step and filter as post-FM step. The dotted line shows the linear trend of the data.

**TABLE 5.** RMSE and Cor. for shape reconstruction using TENG with different levels of variance of Jitter noise ( $\sigma_\eta^2 = (2a_2k + a_1)^2\sigma_\epsilon^2 + 2a_2^2\sigma_\epsilon^4$ ).

$\sigma_\epsilon^2$	TENG only		Kal. Fltr. before TENG		Kal. Fltr. after TENG	
	RMSE	Cor.	RMSE	Cor.	RMSE	Cor.
0	7.315	0.951	6.829	0.951	6.912	0.950
0.1	7.277	0.952	6.800	0.952	6.886	0.950
0.25	7.403	0.951	6.903	0.951	6.961	0.952
0.5	7.405	0.951	6.848	0.950	6.901	0.949
0.75	7.159	0.953	6.677	0.953	6.735	0.953
1	7.511	0.940	6.859	0.945	6.922	0.944
1.25	7.209	0.949	6.891	0.948	6.899	0.949
1.5	7.705	0.928	7.093	0.941	7.144	0.940
1.75	7.626	0.939	6.864	0.947	6.834	0.943
2	7.304	0.941	6.593	0.955	6.532	0.956

the general linear trend of the data. From the table and the figure, it is clear that when SML is used without the proposed scheme, as the noise level increases, the results become poorer. At lower noise levels, the values of correlation are similar for all three scenarios, whereas, when the noise levels increase, the correlation values without the proposed filter decrease sharply at 11.6%. The decrease in correlation values with the use of filter as pre- or post-step is merely 0.22% and 0.23%, respectively. The RMSE values of shape reconstruction when using SML start with 7.339 and increase at high rate of 17.95%, whereas the RMSE values



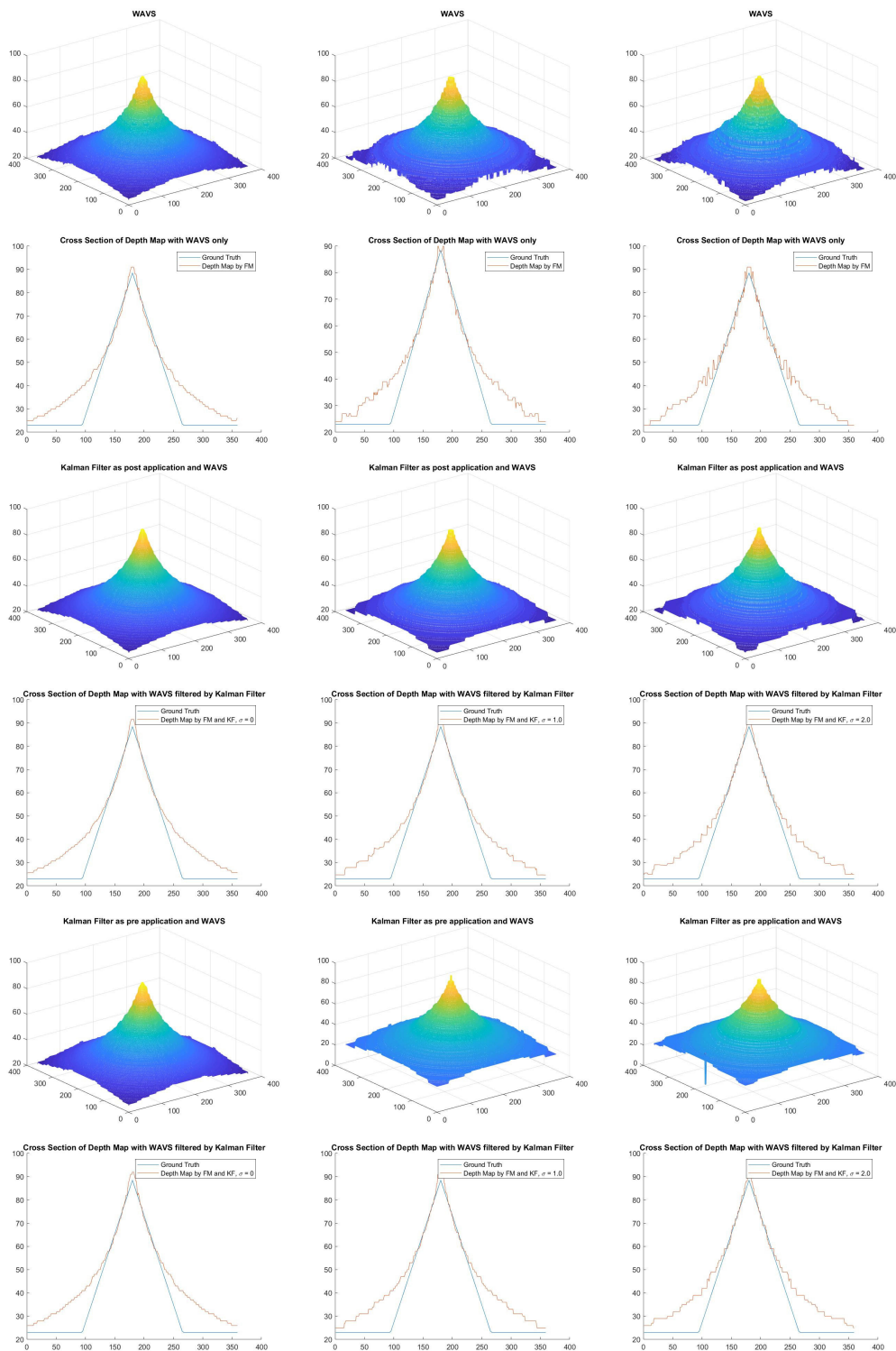


**FIGURE 12.** Shape reconstruction of Simulated Cone using GLVA as FM. (left-right) Jitter variance 0, 1.0 and 2.0 (top-bottom) GLVA only (3D shape and its cross-section), Kalman Filter before GLVA (3D shape and its cross-section), Kalman Filter after GLVA (3D shape and its cross-section).

and when proposed technique is used, about 3% and 10% decrease.

Table 7 and Fig. 11 provide the results of WAVS as FM under all three scenarios of FM and the proposed method

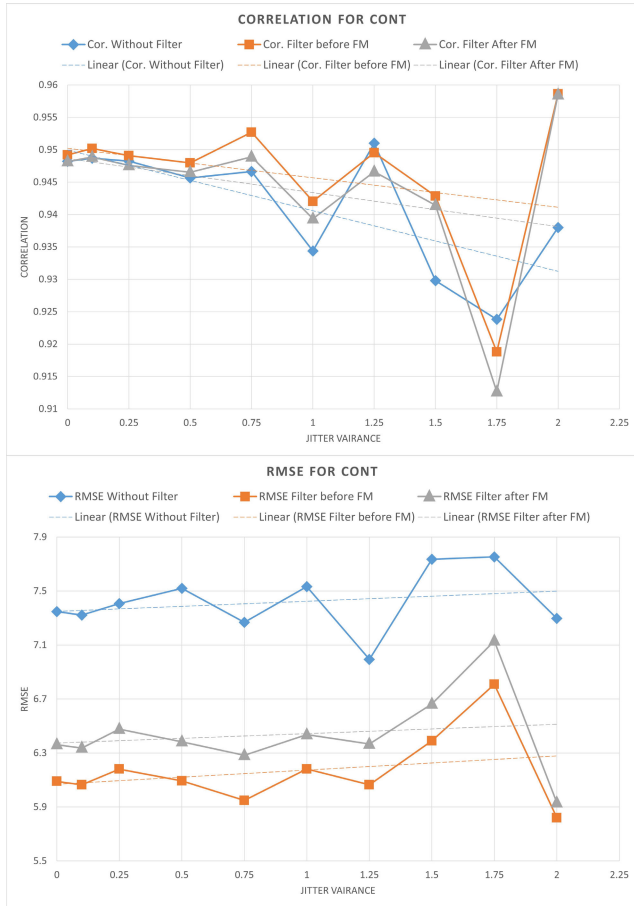
application. As noise increases, the correlation of WAVS decrease at the rate of 1%, but when combined with the proposed technique, with the increase in noise, it increases at 0.7%. However, at lower values of noise, the WAVS with



**FIGURE 13.** Shape reconstruction of Simulated Cone using WAVS as FM. (left-right) Jitter variance 0, 1.0 and 2.0 (top-bottom) WAVS only (3D shape and its cross-section), Kalman Filter before WAVS (3D shape and its cross-section), Kalman Filter after WAVS (3D shape and its cross-section).

or without filtering behaves similar. The trend in RMSE values also suggests that WAVS combined with the proposed technique offers better performance. The shape reconstruction results are provided in Fig. 12 and Fig. 13, respectively.

Table 8 and Fig. 14 show the RMSE and correlation results when using CONT as an FM. In all three scenarios, it behaves poorly. Table 9 and Fig. 15 provide the results of the shape reconstruction of simulated cone when GRA3 is used as the

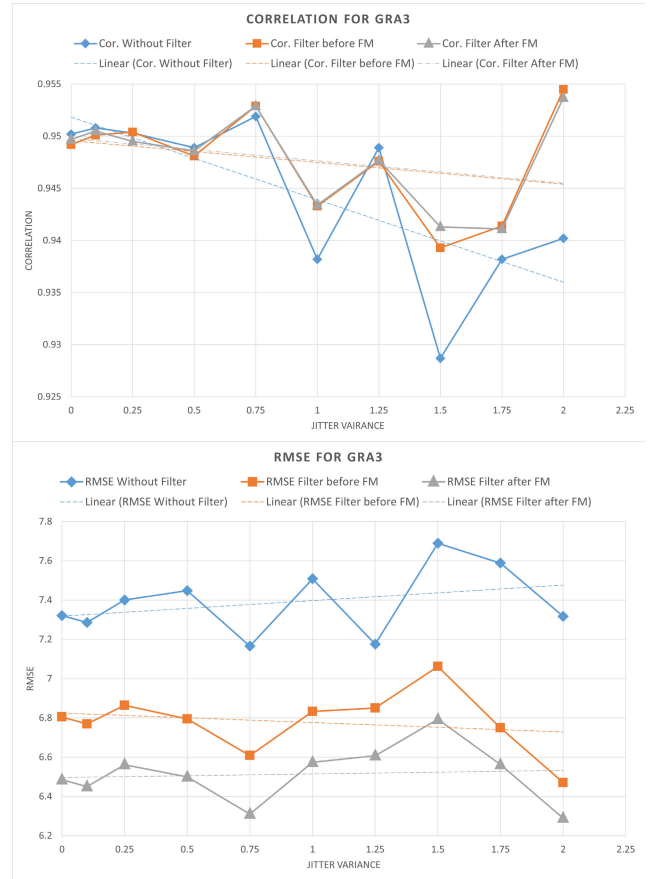


**FIGURE 14.** Correlation and RMSE using CONT as FM with different values of variance of Jitter noise  $\sigma_\eta^2 = (2a_2k + a_1)^2\sigma_\epsilon^2 + 2a_2^2\sigma_\epsilon^4$ ; without using Kalman filter, with filter as pre-FM step and filter as post-FM step. The dotted line shows the linear trend of the data.

**TABLE 12.** RMSE and Cor. for shape reconstruction using CURV with different levels of variance of Jitter noise ( $\sigma_\eta^2 = (2a_2k + a_1)^2\sigma_\epsilon^2 + 2a_2^2\sigma_\epsilon^4$ ).

$\sigma_\epsilon^2$	CURV only		Kal. Fltr. before CURV		Kal. Fltr. after CURV	
	RMSE	Cor.	RMSE	Cor.	RMSE	Cor.
0	7.359	0.949	6.517	0.949	6.808	0.948
0.1	7.319	0.950	6.487	0.949	6.772	0.949
0.25	7.433	0.949	6.595	0.948	6.902	0.948
0.5	7.460	0.949	6.540	0.948	6.835	0.947
0.75	7.191	0.951	6.353	0.952	6.633	0.952
1	7.528	0.938	6.621	0.940	6.885	0.939
1.25	7.259	0.947	6.623	0.946	6.871	0.945
1.5	7.702	0.928	6.859	0.938	7.107	0.936
1.75	7.694	0.932	6.917	0.926	7.137	0.925
2	7.322	0.940	6.329	0.951	6.508	0.952

FM. As the noise increases and only GRA3 is used, the correlation values decrease at the rate of 0.8%. When GRA3 is combined with the proposed scheme, the correlation values almost remain consistent, with the trend of 0.2% decrease over increase in noise levels. When the proposed method is used, the RMSE values for GRA3 are lower. Tables 10 to 13 provide the results of DCTE, SFRQ, CURV, and SFIL FMs. The RMSE and correlation values have similar trend to the other FMs. The combinations of all these FMs with the proposed method show better correlation values and lower



**FIGURE 15.** Correlation and RMSE using GRA3 as FM with different values of variance of Jitter noise  $\sigma_\eta^2 = (2a_2k + a_1)^2\sigma_\epsilon^2 + 2a_2^2\sigma_\epsilon^4$ ; without using Kalman filter, with filter as pre-FM step and filter as post-FM step. The dotted line shows the linear trend of the data.

**TABLE 13.** RMSE and Cor. for shape reconstruction using SFIL with different levels of variance of Jitter noise ( $\sigma_\eta^2 = (2a_2k + a_1)^2\sigma_\epsilon^2 + 2a_2^2\sigma_\epsilon^4$ ).

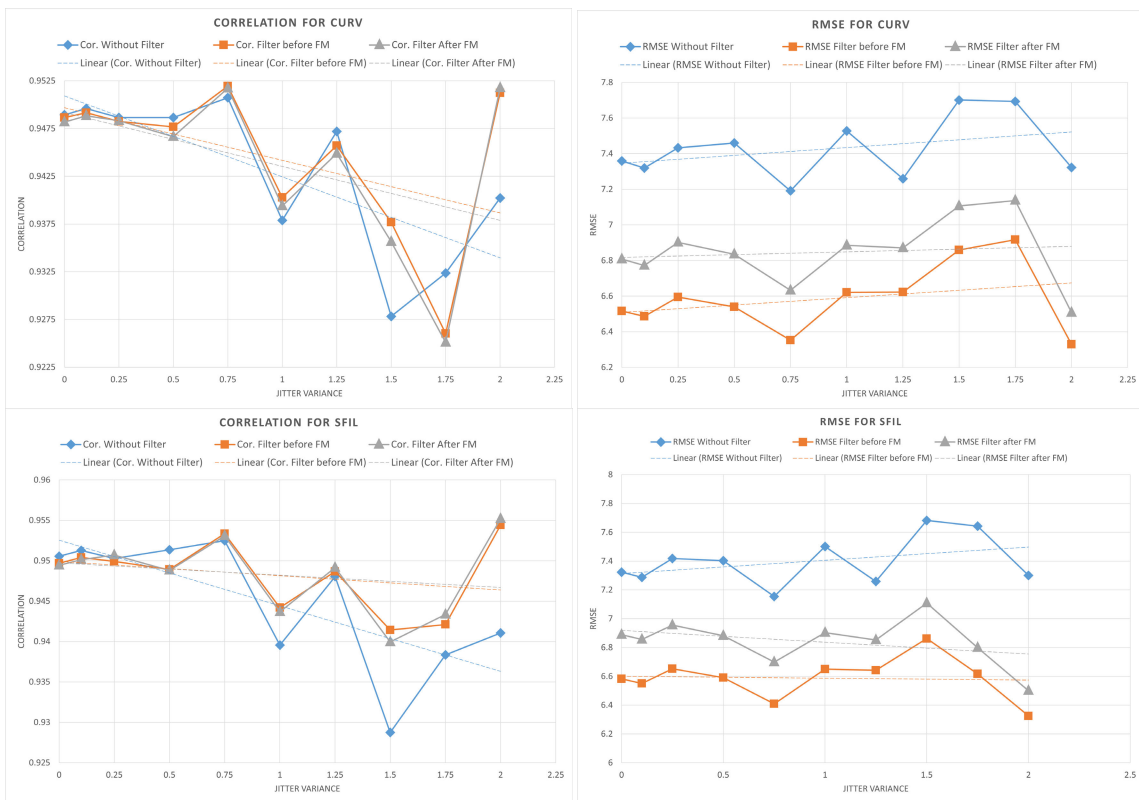
$\sigma_\epsilon^2$	SFIL only		Kal. Fltr. before SFIL		Kal. Fltr. after SFIL	
	RMSE	Cor.	RMSE	Cor.	RMSE	Cor.
0	7.322	0.951	6.583	0.950	6.889	0.949
0.1	7.287	0.951	6.551	0.950	6.856	0.950
0.25	7.418	0.950	6.653	0.950	6.956	0.951
0.5	7.402	0.951	6.591	0.949	6.882	0.949
0.75	7.155	0.952	6.409	0.953	6.699	0.953
1	7.501	0.940	6.651	0.944	6.903	0.944
1.25	7.259	0.948	6.643	0.949	6.852	0.949
1.5	7.682	0.929	6.862	0.941	7.109	0.940
1.75	7.643	0.938	6.617	0.942	6.799	0.943
2	7.301	0.941	6.325	0.954	6.500	0.955

RMSE values, as compared to using FMs only. The RMSE and Cor. graphs are shown in Fig. 16 and Fig. 17, respectively.

Table 14 with Fig. 18 and Fig. 19, show the similar comparison of Real Cone with and without the proposed scheme for Jitter noise variance  $\sigma_\eta$  affected by  $0 \leq \sigma_\epsilon^2 \leq 0.75$ . The table and graphs clearly suggest the effectiveness of the proposed scheme. The correlation values across noisy conditions for the proposed scheme are higher as compared to using FM(s) only, whereas the RMSE values are lower. Figures 20 to 22 represent the shape construction of Real Cone with and without the proposed scheme. The reconstructed shapes when

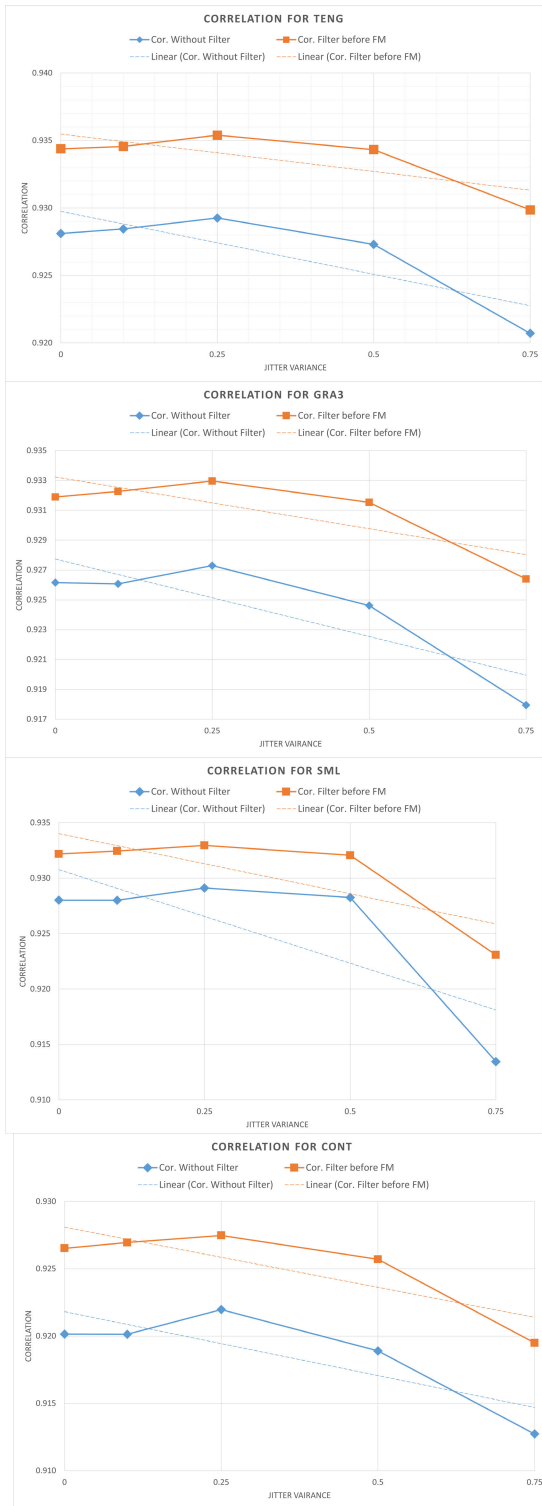


**FIGURE 16.** Correlation and RMSE using DCTE and SFRQ as FM with different values of variance of Jitter noise  $\sigma_{\eta}^2 = (2a_2k + a_1)^2\sigma_{\epsilon}^2 + 2a_2^2\sigma_{\xi}^4$ ; without using Kalman filter, with filter as pre-FM step and filter as post-FM step. The dotted line shows the linear trend of the data.



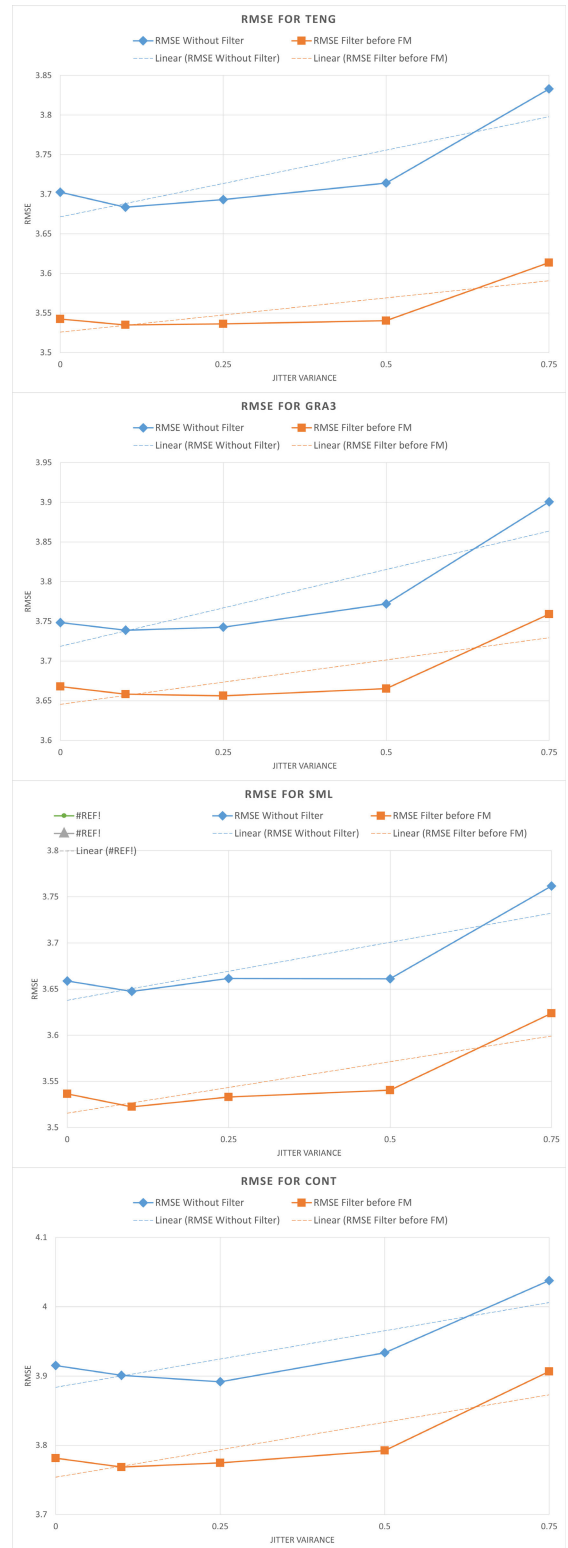
**FIGURE 17.** Correlation and RMSE using CURV and SFIL as FM with different values of variance of Jitter noise  $\sigma_{\eta}^2$ ; without using Kalman filter, with filter as pre-FM step and filter as post-FM step. The dotted line shows the linear trend of the data.





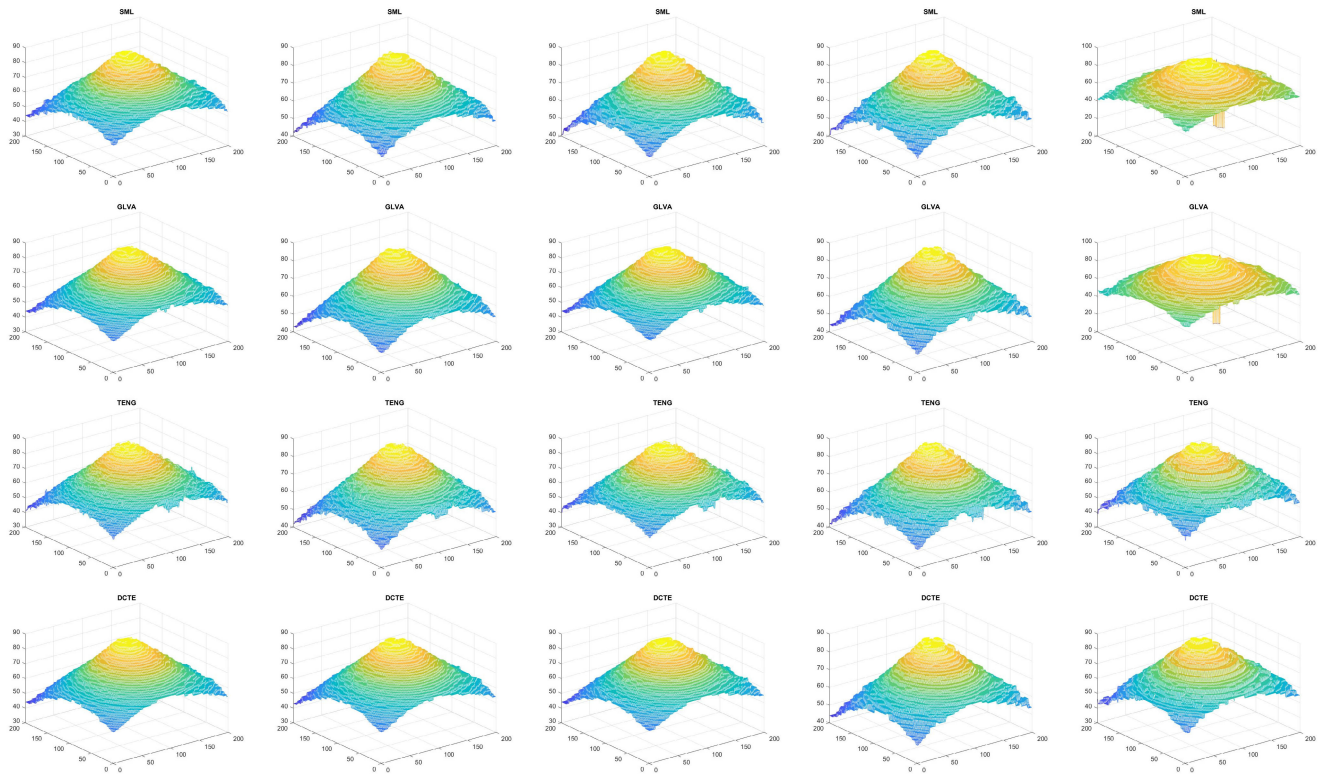
**FIGURE 18.** Correlation for Real Cone with different TENG, SML, GRA3 and CONT with different values of variance of Jitter noise  $\sigma_y^2 = (2a_2k + a_1)^2\sigma_\epsilon^2 + 2a_2^2\sigma_\epsilon^4$ ; with and without using proposed filter. The dotted line shows the linear trend of the data.

using the proposed scheme are smoother as compared to the ones reconstructed using FM(s) only, as these shapes have surface-roughness produced due to jitter, and FMs cannot remove this alone.

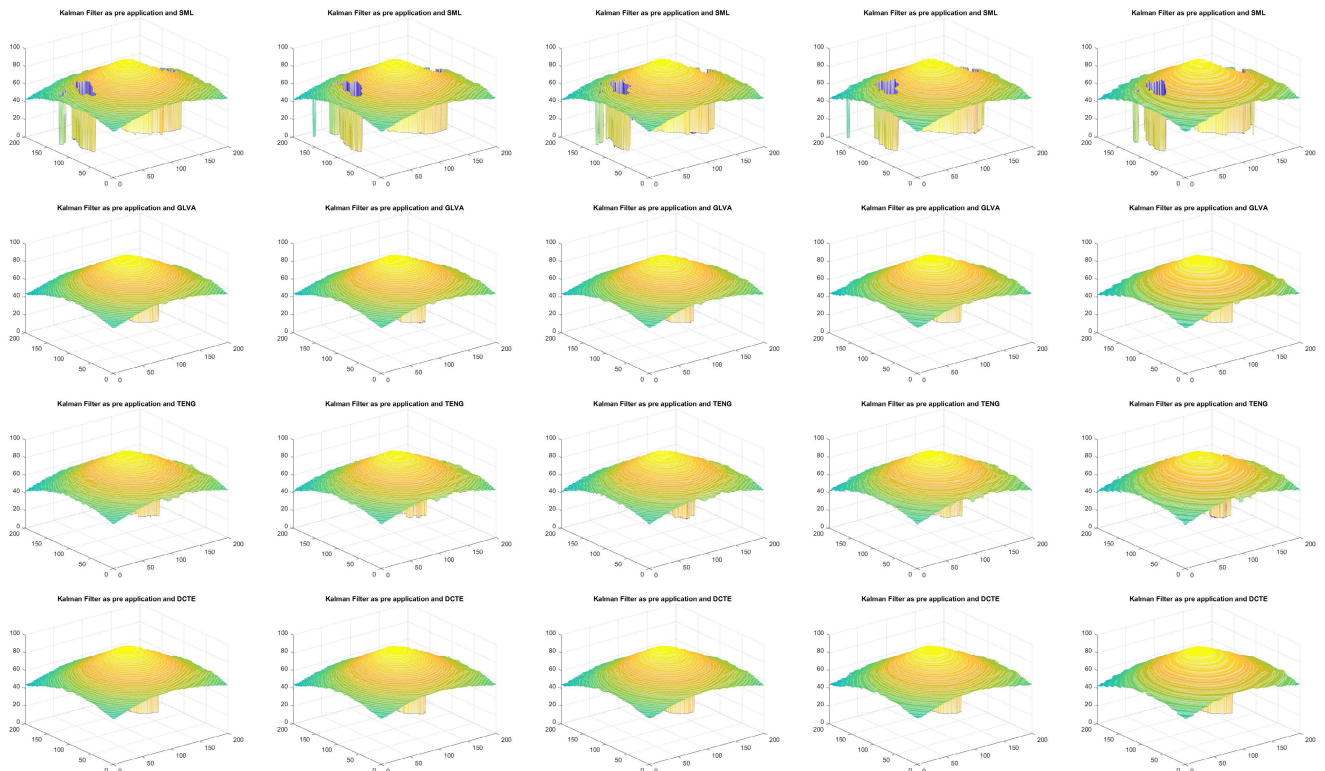


**FIGURE 19.** RMSE for Real Cone with different TENG, SML, GRA3 and CONT with different values of variance of Jitter noise  $\sigma_y^2 = (2a_2k + a_1)^2\sigma_\epsilon^2 + 2a_2^2\sigma_\epsilon^4$ ; with and without using proposed filter. The dotted line shows the linear trend of the data.

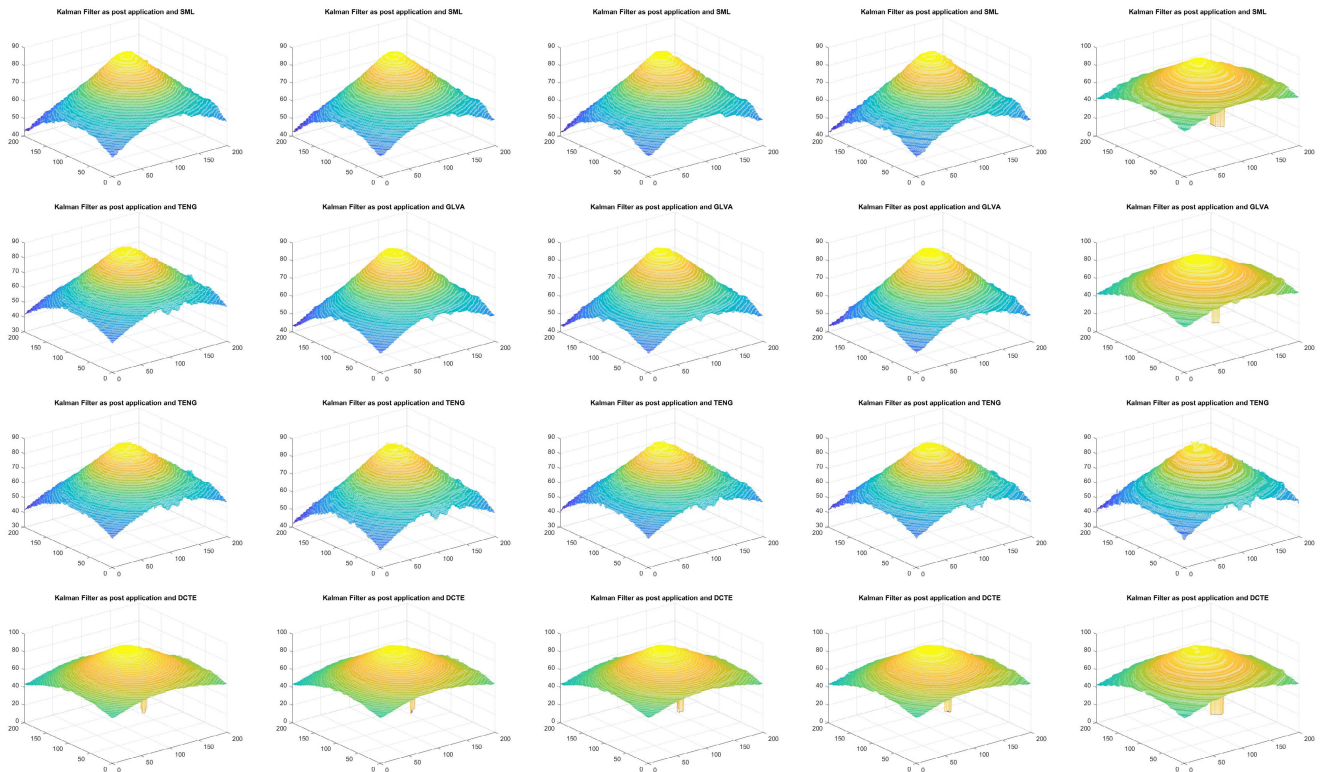
Fig. 23 and Fig. 24 represent the shape reconstruction for Real Plane, LCD-TFT filter, Image-I, and Coin, using SML, GLVA, TENG, DCTE, and WAVS FMs, with FM



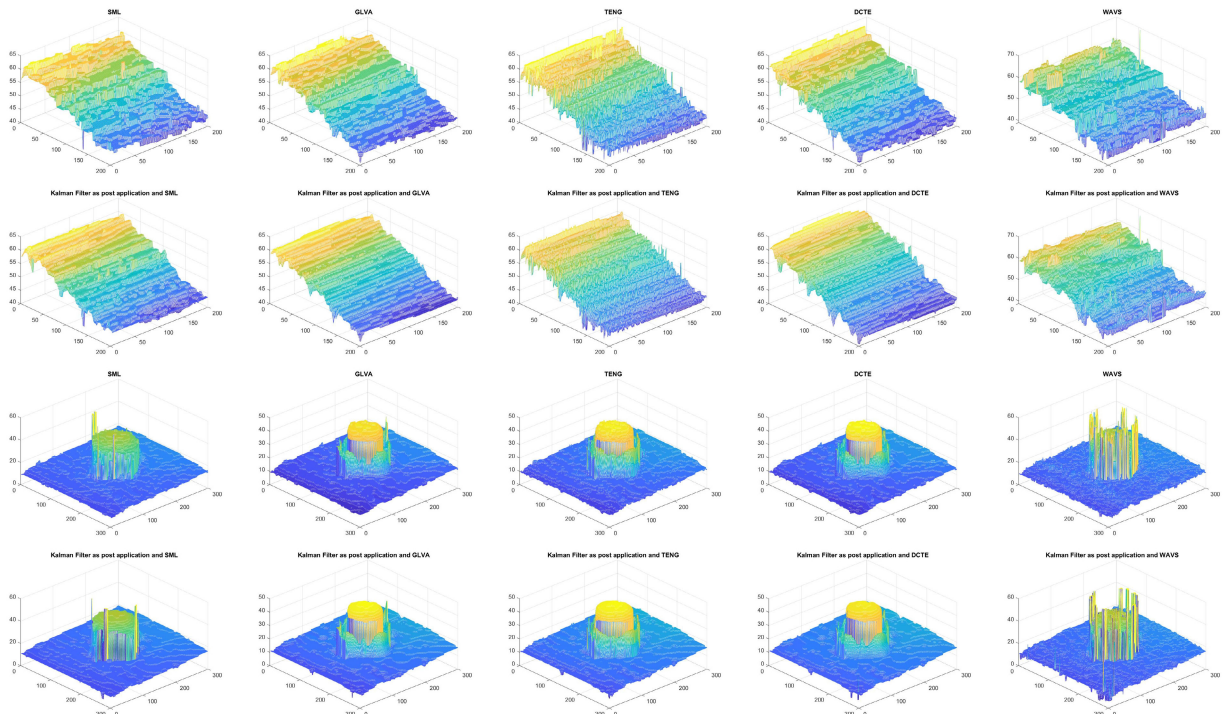
**FIGURE 20.** Shape reconstruction of Real Cone using different FMs, without using proposed filtering technique. (left-right)  $\sigma_\epsilon^2 = 0, \sigma_\epsilon^2 = 0.1, \sigma_\epsilon^2 = 0.25, \sigma_\epsilon^2 = 0.5, \sigma_\epsilon^2 = 0.75$ . (top-bottom) SML, GLVA, TENG, DCTE.



**FIGURE 21.** Shape reconstruction of Real Cone using different FMs, with using proposed filtering technique as pre-step. (left-right)  $\sigma_\epsilon^2 = 0, \sigma_\epsilon^2 = 0.1, \sigma_\epsilon^2 = 0.25, \sigma_\epsilon^2 = 0.5, \sigma_\epsilon^2 = 0.75$ . (top-bottom) SML, GLVA, TENG, DCTE.



**FIGURE 22.** Shape reconstruction of Real Cone using different FMs, without using proposed filtering technique as post-step. (left-right)  $\sigma_\epsilon^2 = 0$ ,  $\sigma_\epsilon^2 = 0.1$ ,  $\sigma_\epsilon^2 = 0.25$ ,  $\sigma_\epsilon^2 = 0.5$ ,  $\sigma_\epsilon^2 = 0.75$ . (top-bottom) SML, GLVA, TENG, DCTE.



**FIGURE 23.** Shape reconstruction of (top-bottom) Real Plane and LCD-TFT using (left-right) SML, GLVA, TENG, DCTE and WAVS.

only and the proposed filter used as post-FM application. In Real Plane reconstructed shape, the similar smoothness phenomenon (as Real Cone) can be observed in Fig. 23. The

roughness in shape, due to jittering when using only FM is smoothed by the filter i.e., the jitter effect is removed. In LCD-TFT filter, the cylindrical shape of the filter is

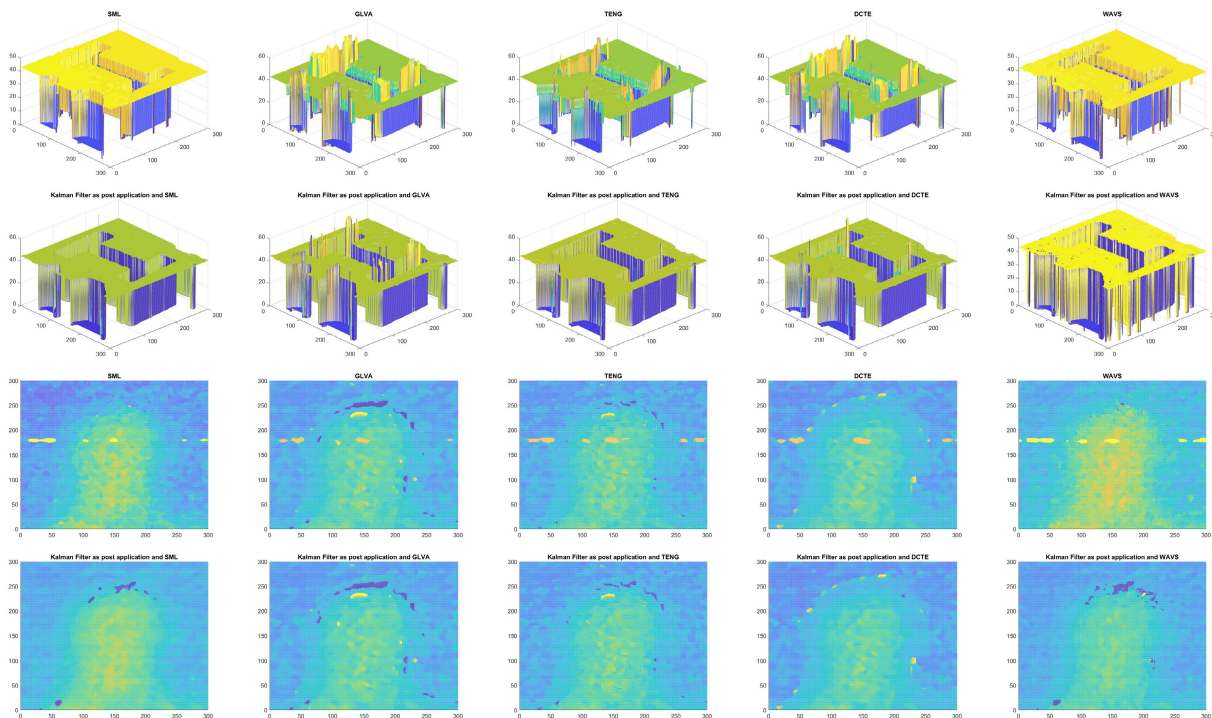


FIGURE 24. Shape reconstruction of (top-bottom) Image-I and Coin using (left-right) SML, GLVA, TENG, DCTE and WAVS.

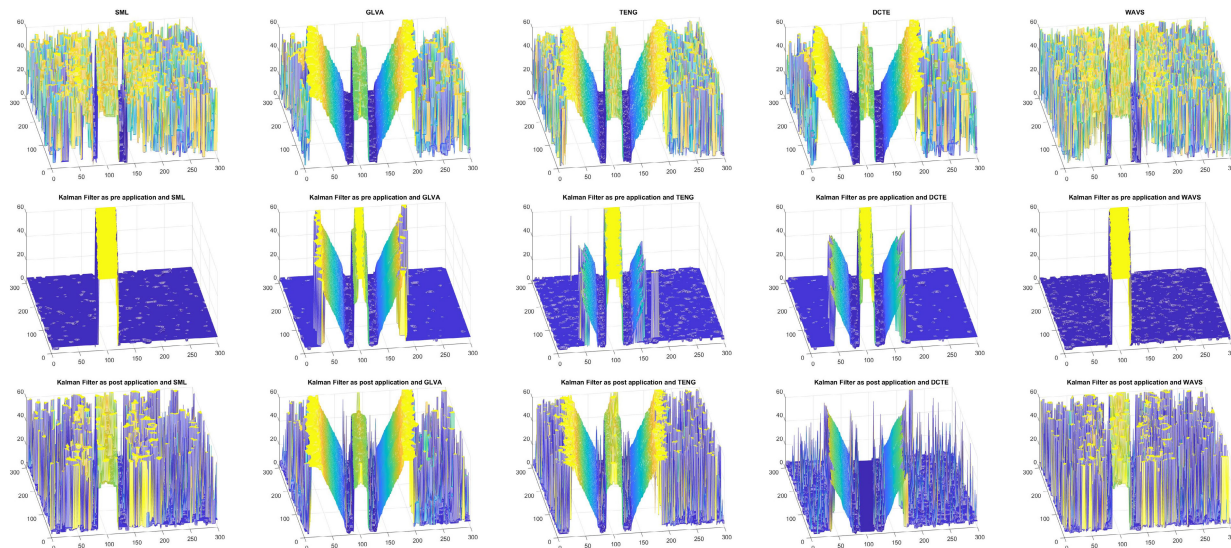


FIGURE 25. Shape reconstruction of Groove using proposed technique as pre/post-filtering, (left-right) SML, GLVA, TENG, DCTE and WAVS.

preserved, and the surface around it is smoothed by the filtering process. In the case of Image-I dataset, not much difference can be observed visually, as the jitter in this sequence is quite low. In the case of Coin sequence a depth abnormality can be observed also in Fig. 23 near the vertical axis of 175 value in shape reconstruction using all FMs, whereas, when the proposed filter is applied, it is removed.

The Groove image sequence is a challenging problem in shape reconstruction [2]. The sides and center of this image sequence are over-exposed, resulting in texture degradation,

which is critical in SFF systems. The slopes in the middle are the only ones that exhibit the change in focus levels. Fig. 25 shows the shape reconstruction of Groove using different FMs, along with the proposed filter applied as pre- and post-FM.

The results presented in the tables, graphs and reconstructed shapes in the manuscript clearly show that the Jitter noise affects the overall accuracy of the SFF systems, and can be removed by applying the proposed filtering technique using Kalman filter. The proposed scheme shows promising results.

**TABLE 14. RMSE and Cor. for Real Cone shape reconstruction using FMs with different levels of variance of Jitter noise**  
 $(\sigma_{\eta}^2 = (2a_2k + a_1)^2\sigma_{\epsilon}^2 + 2a_2^2\sigma_{\epsilon}^4)$ .

Focus Measure	$\sigma_{\epsilon}^2$	FM only		Kal. Fltr. and FM	
		RMSE	Cor.	RMSE	Cor.
TENG	0	3.703	0.928	3.542	0.934
	0.1	3.684	0.928	3.535	0.935
	0.25	3.693	0.929	3.536	0.935
	0.5	3.714	0.927	3.540	0.934
	0.75	3.833	0.921	3.614	0.930
SML	0	3.659	0.928	3.536	0.932
	0.1	3.648	0.928	3.522	0.932
	0.25	3.661	0.929	3.533	0.933
	0.5	3.661	0.928	3.541	0.932
	0.75	3.762	0.913	3.624	0.923
GLVA	0	3.685	0.929	3.532	0.935
	0.1	3.650	0.929	3.519	0.935
	0.25	3.665	0.930	3.524	0.936
	0.5	3.679	0.929	3.531	0.935
	0.75	3.765	0.890	3.727	0.889
SFRQ	0	5.749	0.855	5.136	0.873
	0.1	5.604	0.860	5.171	0.871
	0.25	5.559	0.862	5.146	0.873
	0.5	5.706	0.856	5.156	0.871
	0.75	7.046	0.790	6.398	0.812
WAVS	0	3.664	0.928	3.524	0.932
	0.1	3.634	0.929	3.545	0.931
	0.25	3.814	0.922	3.629	0.929
	0.5	3.757	0.923	3.645	0.929
	0.75	4.681	0.738	4.513	0.799
CONT	0	3.915	0.920	3.781	0.927
	0.1	3.901	0.920	3.769	0.927
	0.25	3.892	0.922	3.775	0.927
	0.5	3.934	0.919	3.792	0.926
	0.75	4.038	0.913	3.907	0.919
DCTE	0	4.483	0.898	3.793	0.925
	0.1	4.380	0.902	3.755	0.926
	0.25	4.610	0.894	3.775	0.926
	0.5	4.559	0.895	3.783	0.925
	0.75	4.718	0.893	3.905	0.918
GRA3	0	3.749	0.926	3.668	0.932
	0.1	3.739	0.926	3.658	0.932
	0.25	3.743	0.927	3.656	0.933
	0.5	3.772	0.925	3.665	0.932
	0.75	3.901	0.918	3.759	0.926
SFIL	0	9.691	0.648	7.390	0.757
	0.1	10.113	0.628	7.420	0.754
	0.25	9.550	0.658	7.444	0.757
	0.5	9.614	0.652	7.413	0.756
	0.75	9.651	0.642	7.813	0.731

**VII. CONCLUSION**

In SFF when the shape of the object is discretised into image frames, constant inter-frame distance is assumed. However, in practice, this inter-frame distance is prone to errors. This is due to mechanical errors in gear assembly of the translational-stage or lens-focusing-assembly of the imaging device. This can cause errors, which are referred to as Jitter noise in the literature. Jitter noise is not visible in images, because each pixel in an image will be subjected to the same error in focus. Thus, using traditional techniques of the denoising of images will not work.

In this paper, Jitter noise is first modeled and the *mean* and *variance* of this noise are formulated. It is also shown that this Jitter noise is dependent on the first and second derivatives of the focus/pixel curves, and follows gamma distribution. The

design of the system and measurement models for the proposed scheme using Kalman filter are presented and applied to the focusing curves to remove this noise. The proposed scheme can be applied in two ways, as pre-FM application or post-FM application. Unlike previously proposed techniques for Jitter noise removal in SFF systems, the proposed scheme utilizes single measurement for each step, and utilizes the dynamic approach with Kalman filter. Thus, it is faster and more accurate as compared to its predecessors.

The experiments are performed on seven objects: *one* simulated and *six* real. *Ten* noise levels are tested on the simulated object, and *four* levels on the real objects. Both pre- and post-applications are tested and the results are presented. The RMSE and correlation are used as metric measures. The experiments show the effectiveness of the proposed scheme.

**REFERENCES**

- [1] M. B. Ahmad and T. S. Choi, "Application of three dimensional shape from image focus in LCD/TFT displays manufacturing," *IEEE Trans. Consum. Electron.*, vol. 53, no. 1, pp. 1–4, Feb. 2007.
- [2] H. Mutahira, M. S. Muhammad, M. Li, and D. Shin, "A simplified approach using deep neural network for fast and accurate shape from focus," *Microsc. Res. Technique*, vol. 84, no. 4, pp. 656–667, Apr. 2021.
- [3] J.-B. Perraud, J.-P. Guillet, O. Redon, M. Hamdi, F. Simoens, and P. Mounaix, "Shape-from-focus for real-time terahertz 3D imaging," *Opt. Lett.*, vol. 44, no. 3, pp. 483–486, 2019.
- [4] J. Tang, Z. Qiu, and T. Li, "A novel measurement method and application for grinding wheel surface topography based on shape from focus," *Measurement*, vol. 133, pp. 495–507, Feb. 2019.
- [5] M. Wendland, "Shape from focus image processing approach based 3D model construction of machined part," Ph.D. dissertation, School Eng. Comput. Sci. Mech. Eng., Univ. Pacific, Stockton, CA, USA, 2015.
- [6] S. Zhou and M. Seita, "Large-area surface topography analysis of additively manufactured metallic materials using directional reflectance microscopy," *Mater. Sci. Eng., A*, vol. 760, pp. 489–497, Jul. 2019.
- [7] P. Favaro, A. Mennucci, and S. Soatto, "Observing shape from defocused images," *Int. J. Comput. Vis.*, vol. 52, no. 1, pp. 25–43, 2003.
- [8] S. Pertuz, D. Puig, and M. A. Garcia, "Analysis of focus measure operators for shape-from-focus," *Pattern Recognit.*, vol. 46, no. 5, pp. 1415–1432, 2013.
- [9] M. S. Muhammad and T.-S. Choi, "Sampling for shape from focus in optical microscopy," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 34, no. 3, pp. 564–573, Mar. 2012.
- [10] M. S. Muhammad and T. S. Choi, "A novel method for shape from focus in microscopy using Bezier surface approximation," *Microscopy Res. Techn.*, vol. 73, no. 2, pp. 140–151, Feb. 2010.
- [11] S. K. Nayar and Y. Nakagawa, "Shape from focus," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 16, no. 8, pp. 824–831, Aug. 1994.
- [12] S. K. Nayar, "Shape from focus," Dept. Elect. Comput. Eng., Robot. Inst., Carnegie Mellon Univ., Pittsburgh, PA, USA, Tech. Rep., 1989.
- [13] M. Subbarao and T. Choi, "Accurate recovery of three-dimensional shape from image focus," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 17, no. 3, pp. 266–274, Mar. 1995.
- [14] T.-S. Choi and J. Yun, "Three-dimensional shape recovery from the focused-image surface," *Opt. Eng.*, vol. 39, no. 5, pp. 1321–1326, 2000.
- [15] M. Asif and T.-S. Choi, "Shape from focus using multilayer feedforward neural networks," *IEICE Trans. Inf. Syst.*, vol. 83, no. 4, pp. 946–949, 2000.
- [16] M. Asif and T.-S. Choi, "Shape from focus using multilayer feedforward neural networks," *IEEE Trans. Image Process.*, vol. 10, no. 11, pp. 1670–1675, Nov. 2001.
- [17] H.-J. Kim, M. T. Mahmood, and T.-S. Choi, "An efficient neural network for shape from focus with weight passing method," *Appl. Sci.*, vol. 8, no. 9, p. 1648, 2018.
- [18] U. Ali, V. Pruks, and M. T. Mahmood, "Image focus volume regularization for shape from focus through 3D weighted least squares," *Inf. Sci.*, vol. 489, pp. 155–166, Jul. 2019.

- [19] U. Ali and M. T. Mahmood, "3D shape recovery by aggregating 3D wavelet transform-based image focus volumes through 3D weighted least squares," *J. Math. Imag. Vis.*, vol. 62, no. 1, pp. 54–72, Jan. 2020.
- [20] U. Ali, I. H. Lee, and M. T. Mahmood, "Guided image filtering in shape-from-focus: A comparative analysis," *Pattern Recognit.*, vol. 111, Mar. 2021, Art. no. 107670.
- [21] U. Ali and M. T. Mahmood, "Combining depth maps through 3D weighted least squares in shape from focus," in *Proc. Int. Conf. Electron., Inf., Commun. (ICEIC)*, Jan. 2019, pp. 1–4.
- [22] T. Fan and H. Yu, "A novel shape from focus method based on 3D steerable filters for improved performance on treating textureless region," *Opt. Commun.*, vol. 410, pp. 254–261, Mar. 2018.
- [23] M. T. Mahmood and I. H. Lee, "Shape from focus based on 3D structure tensor using optical microscopy," *Microsc. Res. Technique*, vol. 83, no. 1, pp. 48–55, Jan. 2020.
- [24] Z. Ma, D. Kim, and Y.-G. Shin, "Shape-from-focus reconstruction using nonlocal matting Laplacian prior followed by MRF-based refinement," *Pattern Recognit.*, vol. 103, Jul. 2020, Art. no. 107302.
- [25] T. Yan, P. Wu, Y. Qian, Z. Hu, and F. Liu, "Multiscale fusion and aggregation PCNN for 3D shape recovery," *Inf. Sci.*, vol. 536, pp. 277–297, Oct. 2020.
- [26] H.-S. Jang, M. S. Muhammad, and T.-S. Choi, "Optimizing image focus for shape from focus through locally weighted non-parametric regression," *IEEE Access*, vol. 7, pp. 74393–74400, 2019.
- [27] D.-C. Tsai and H. H. Chen, "Focus profile modeling," *IEEE Trans. Image Process.*, vol. 25, no. 2, pp. 818–828, Feb. 2016.
- [28] H. S. Jang, M. S. Muhammad, and T. S. Choi, "Removal of jitter noise in 3D shape recovery from image focus by using Kalman filter," *Microsc. Res. Technique*, vol. 81, no. 2, pp. 207–213, Feb. 2018.
- [29] S.-A. Lee, H.-S. Jang, and B.-G. Lee, "Jitter elimination in shape recovery by using adaptive neural network filter," *Sensors*, vol. 19, no. 11, p. 2566, Jun. 2019.
- [30] H.-S. Jang, M. S. Muhammad, and T.-S. Choi, "Bayes filter based jitter noise removal in shape recovery from image focus," *J. Imag. Sci. Technol.*, vol. 63, no. 2, 2019, Art. no. 20501.
- [31] H. Jang, M. S. Muhammad, and T. Choi, "Optimal depth estimation using modified Kalman filter in the presence of non-Gaussian jitter noise," *Microsc. Res. Technique*, vol. 82, no. 3, pp. 224–231, Mar. 2019.
- [32] H.-S. Jang, M. S. Muhammad, G. Yun, and D. H. Kim, "Sampling based on Kalman filter for shape from focus in the presence of noise," *Appl. Sci.*, vol. 9, no. 16, p. 3276, Aug. 2019.
- [33] H.-S. Jang, M. S. Muhammad, and M.-K. Kang, "Removal of non-Gaussian jitter noise for shape from focus through improved maximum correntropy criterion Kalman filter," *IEEE Access*, vol. 8, pp. 36244–36255, 2020.
- [34] M. Pollefeys, L. Van Gool, and M. Proesmans, "Euclidean 3D reconstruction from image sequences with variable focal lengths," in *Proc. Eur. Conf. Comput. Vis.* Berlin, Germany: Springer, 1996, pp. 31–42.
- [35] A. S. Malik and T.-S. Choi, "Effect of noise and source illumination on 3D shape recovery," *Int. J. Pattern Recognit. Artif. Intell.*, vol. 22, no. 5, pp. 945–958, 2008.
- [36] M.-C. Lu, "Computer modeling and simulation techniques for computer vision problems," Ph.D. dissertation, State Univ., Stony Brook, NY, USA, 1993. [Online]. Available: <http://www.ece.sunysb.edu/~murali/>
- [37] M. T. Mahmood and T.-S. Choi, "Nonlinear approach for enhancement of image focus volume in shape from focus," *IEEE Trans. Image Process.*, vol. 21, no. 5, pp. 2866–2873, May 2012.
- [38] M. Subbarao and J. K. Tyan, "Selecting the optimal focus measure for autofocusing and depth-from-focus," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 20, no. 8, pp. 864–870, Aug. 1998.
- [39] A. S. Malik and T.-S. Choi, "A novel algorithm for estimation of depth map using image focus for 3D shape recovery in the presence of noise," *Pattern Recognit.*, vol. 41, no. 7, pp. 2200–2225, 2008.
- [40] R. Minhas, A. A. Mohammed, and Q. M. J. Wu, "An efficient algorithm for focus measure computation in constant time," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 1, pp. 152–156, Jan. 2012.
- [41] S.-O. Shim, W. Aziz, A. Banjar, A. Alamri, and M. Alqarni, "Improving depth computation from robust focus approximation," *IEEE Access*, vol. 7, pp. 20144–20149, 2019.
- [42] R. Minhas, A. A. Mohammed, and Q. M. J. Wu, "Shape from focus using fast discrete curvelet transform," *Pattern Recognit.*, vol. 44, no. 4, pp. 839–853, Apr. 2011.

- [43] A. M. Eskicioglu and P. S. Fisher, "Image quality measures and their performance," *IEEE Trans. Commun.*, vol. 43, no. 12, pp. 2959–2965, Dec. 1995.
- [44] D. Ghadiyaram and A. C. Bovik, "Massive online crowdsourced study of subjective and objective picture quality," *IEEE Trans. Image Process.*, vol. 25, no. 1, pp. 372–387, Jan. 2016.



Her research interests include 3-D shape recovery and vision-based robot navigation.



His research interests include estimation, filtering, tracking, data fusion, identification, and other multidimensional data processing methods.



His research interests include robot vision, image processing, robot navigation, and robots in health care.



His research interests include artificial intelligence, neural networks, wireless communication and networks, ubiquitous computing and networking, sensor networks, and big data.

...