Editor, Concern #1: ***Can you please justify why you have chosen discrete steps for the pitch realization rather than continuous (considering your specific contribution to improved tracking control)?***

Author Response: The authors are grateful for the comments of the editor and agree that this choice must be justified and explained in the manuscript. The motivation for realizing pitch control with discrete steps rather than continuous control is to allow for simple playability for accessibility and demonstration purposes. Continuous pitch control, as with the Theremin, requires extensive experience and muscle memory to manage even simple musical phases with proper intonation. Hence, we provide the discrete note selection to provide an enjoyable experience for the less-experienced user and allow for simple demonstration without expert musicianship. However, our implementation can be easily modified to provide continuous note selection for advanced musicians, if desired. Section IV-B of the updated manuscript provides a detailed explanation of our choice for pitch realization and comparison with the Theremin.

From Section IV-B of the updated manuscript:

> "Unlike the Theremin, which allows for continuous note selection, our interface quantizes the user input into predefined subregions corresponding to notes defined by the user. To play the desired note, the user must move their hand vertically to the position corresponding to that note. The subregions and allowed notes can be programmed by the user in the interactive MATLAB GUI. Continuous control, similar to the Theremin, can be easily implemented using our user interface. However, true continuous pitch control, as with an analog Theremin, is achievable only to a certain extent as discretization is required at some point in a digital software. We have selected discrete pitch control as the default for our framework for simple note selection to promote accessibility to numerous fields. The Theremin is commonly regarded as one of the most difficult instruments to control given its continuous note selection and requires extensive experience to manage even simple musical phases with proper intonation. Our solution allows advanced musicians to enable continuous pitch control while offering a simple interface for less-experienced users. Thus, our platform is accessible to many users providing an enjoyable experience and enabling less-effortful demonstration without expert musicianship."

Editor, Concern #2: ***On p12, col. 2, lines 10-14: it is not clear what you mean by "manual transcription".***
Author Response: The authors agree with the editor's suggestion. The term, "manual transcription" was meant to refer to manually transcribing MIDI sequences of notes and control parameters into a digital audio workstation (DAW), but its usage was rather nebulous in the previous version of the manuscript. We have corrected this issue by providing an additional explanation in the updated manuscript.

From Section VI of the updated manuscript:

> ". . . the proposed methods provide the musician a sufficient and consistent level of control and offer an elegant new musical interface capable of generating unique phrases previously only possible by transcribing MIDI notes and control parameters manually into a digital audio workstation (DAW) or other live digital synthesis platform."

Reviewer #1, Concern #1: ***In the revised manuscript, I do not see any comparison or discussion with the papers of Yimin, Gurbuz, Almeida, etc. that I had suggested. Please add the discussion to the manuscript as well and not just the response. Also, the motivation for using FCNN over other networks is not sufficiently described in detail.***

Author Response: The authors would like to thank reviewer #1 for their contribution and comments. In the updated manuscript, we have included a discussion on the works of Gurbuz [13], Liu [14], Almedia [19], as well as Brian Toomajian's seminal work on gesture recognition using micro-Doppler signatures with convolutional neural networks [16]. Additionally, we have referenced Jingkun Gao's paper on SAR image enhancement with complex-valued convolutional neural networks [20] and Hyungju Kim's recent work on improving azimuth angle resolution with neural networks [21] for comparison against the state-of-the-art radar resolution improvement techniques using deep learning. The relevant portion of the updated manuscript is included below:

From Section I:

> "Existing work on contactless gesture control, such as gesture radar, commonly applies machine learning and deep learning techniques. Gurbuz *et al.* employ a multi-frequency radio-frequency (RF) sensor to recognize American sign language patterns with high accuracy using several machine learning techniques such as support vector machines (SVM), random forest, linear discriminant analysis, and k-nearest neighbors [13]. Foot gestures are classified from EMG data using an SVM classifier [19]. In [14], Sang *et al.* compare the classification rates from several techniques from traditional machine learning approaches such as hidden Markov models (HMMs) to state-of-the-art deep learning models including convolutional neural networks (CNNs), recurrent neural networks (RNNs), and end-to-end networks. On mmWave radar, [16] proposes a CNN classifier for dynamic gestures using micro-Doppler signatures and [17] employs a sterile technique to improve the classification rate of static, non-moving hand poses. These previous efforts offer a sufficient solution to many classification applications where data is collected and used to determine the gesture or hand pose performed by a subject from a set of predefined classes.
>
> In this article, however, we consider a distinctly separate issue and offer a deep learning-based signal processing solution. Rather than classifying a sample from a set of classes, our proposed framework seeks to extract continuous spatial and temporal features from the hand's position and motion by improving the spatiotemporal image resolution. To achieve this goal, we apply a novel fully convolutional neural network (FCNN) to preserve the geometry of the image and perform super-resolution for improved localization. Radar signal processing using FCNNs is advantageous over other CNN techniques as it allows for data-driven "enhancement" rather than dimensionality reduction, as in classification. Hence, rather than suffering from information loss, the regressive FCNN provides additional "context" learned during the training phase to enhance the radar data. The enhanced data offer several advantages such as improved signal-to-noise-ratio (SNR), clutter removal, near-field image correction, aliasing suppression, and higher-resolution peaks. In this article, traditional radar signal processing algorithms are shown to achieve considerable performance gains when applied to enhanced data. In [20] and [21], an

FCNN and U-Net are employed to enhance the resolution of a radar image under far-field, plane-wave assumptions. Our novel approach unifies FCNN-based super-resolution with near-field imaging, which requires more difficult spherical-wave compensation, on a small (8-channel) array and is shown to improve hand-tracking performance significantly. To our knowledge, this article is the first documented effort towards near-field radar image super-resolution using an FCNN approach for improved localization. Incorporating our enhancement FCNN in the signal processing chain enables fine motion tracking unattainable by existing techniques."

Reviewer #1, Concern #2: ***It remains unclear what is the non-trivial novelty in the signal processing algorithm. Specifically, what is the non-trivial challenge in developing the modified particle filter? The addition of weights is hardly a non-trivial improvement. Further, the paper does not analyze the stability of this modified filter.***

Author Response: The authors are grateful for the reviewer's comment. The primary signal processing novelty and main contribution of our work is the super-resolution FCNN. For the particle filter, our modifications may be less groundbreaking but are still novel to the literature and provide details necessary for replicating and understanding our software implementation. The challenge in developing the modified particle filter is to incorporate both spatial and temporal features to improve localization. We agree that this extension to the particle filter is less substantial than our super-resolution FCNN, but we hold that its discussion should be included in the paper to provide an adequate description of our complete system-level design. Finally, the stability of the general particle filter is investigated elsewhere, and we believe our modifications will not impair the analysis therein [31]. The relevant portions of the updated manuscript are included below:

From Section III-B:

Our proposed extension of the particle filter introduces a novel particle resampling and weight calculation procedure. These modifications do not significantly alter the existing particle filter framework but are included to demonstrate their viability in hand-tracking with radar and adequately document our complete software implementation. Furthermore, the stability of the particle filter algorithm has been investigated elsewhere and will not be addressed in this article [31].

From Section III-B1:

In this section, we present a dynamic weighting technique for updating $\mathbf{a}$ in real-time by exploiting the dependence between position and velocity. Our approach considers corroboration between the Doppler velocity estimate and the velocity estimated from the range samples as a measure of the new measurement's reliability. Thus, the dependability of the Doppler velocity can improve tracking of the target position along the range z dimension even in the presence of noisy position estimates.

Reviewer #1, Concern #3: ***I still do not see the continuous-time signals in the paper. The paper starts directly with the discrete-time received signal. There is no mention of how this signal is mathematically obtained and what is the channel model.***

Author Response: The authors agree with the reviewer's remark and have implemented corrective actions to remedy the noted issues. In Section II-A of the updated manuscript, we have included a continuous-time model of the received signal, noted the ideal noiseless channel assumption, and mathematically detailed the process of obtaining the discrete-time signal. The relevant portion of the updated manuscript is included below:

From Section II-A:

> The FMCW chirp signal model is well documented in the literature [24]-[26] and is discussed in this section for reference and continuity throughout this paper. First, consider a single transmitter/receiver pair located at $(y_T, 0)$ and $(y_R, 0)$ in the y-z plane, respectively, and an ideal point target with reflectivity p located at (y, z). Assuming ideal propagation on a noiseless channel, the continuous-time signal can be modeled as
>
> $$s(y_T, y_R, k) = \frac{p}{R_T R_R} e^{jk(R_T+R_R)}, \quad (1)$$
>
> where R_T, R_R are the distances from the transmitter and receiver to the point target, respectively and $k = 2\pi f/c$ is the instantaneous wavenumber.
>
> The frequency of the chirp $f = f_0 + Kt$ increases linearly against time with slope K and starting frequency f_0.
>
> The continuous-time signal (1) is sampled with sampling frequency f_S by the radar analog-to-digital converter (ADC) and can be written in discrete time as
>
> $$s(y_T, y_R, n_k) = \frac{p}{R_T R_R} e^{j(k_0+\Delta n_k)(R_T+R_R)}, \quad (2)$$
>
> where n_k is the wavenumber index, $k_0 = 2\pi f_0/c$ is the starting wavenumber corresponding to the starting frequency f_0, and $\Delta = 2\pi K/(cf_S)$ is the wavenumber step size.

Reviewer #1, Concern #4: ***As per the updated algorithm chart, the only input to the algorithm is the vector r. But in Steps 2 and 3, A, \Sgma_w, \phi have been used without any prior values or inputs. Further, s_{n-1} has been used in Step 2 but it was never initialized in the algorithm.***

Author Response: The authors are grateful for the critique from the reviewer of the algorithm chart and description. We have included each of the algorithm inputs and a thorough description of each input, their initializations (if necessary), and usage prior to the algorithm chart. The relevant portions of the updated manuscript are included below:

From Section III-B:

Before executing the iterative algorithm, the initial particle states matrix, $\mathbf{X}_0$, and initial weights vector, $\mathbf{w}_0$, are initialized with random locations throughout the region of interest (ROI) and uniform weights, respectively. At each iteration, the particle filter receives $\mathbf{r} = [\hat{y}, \hat{z}]^T$, the newest location estimates; $\hat{\mathbf{X}}_{n-1}$ the previous particle filter locations, $\mathbf{X}_{n-1}$, sampled using weights $\mathbf{w}_{n-1}$; $\mathbf{a} = [a_y, a_z]^T$, the weighting vector whose two elements provide weight to the noisy estimates $\hat{y}$ and $\hat{z}$, respectively; $\mathbf{s}_{n-1}$, the vector of previously estimated states, $\mathbf{\Psi} = [\boldsymbol{\psi}_1, \boldsymbol{\psi}_2]^T$, a matrix consisting of two random vectors from the distribution $G(\mathbf{0}_N, \mathbf{\Sigma}_\psi)$; and $\mathbf{\Sigma}_w$, the covariance matrix for the weight distribution.

Algorithm: Modified Particle Filter Algorithm

input : $\mathbf{r} = [\hat{y}, \hat{z}]^T, \hat{\mathbf{X}}_{n-1}, \mathbf{a}, \mathbf{s}_{n-1}, \mathbf{\Psi}, \mathbf{\Sigma}_w$

output: $\mathbf{s}_n = [\tilde{y}, \tilde{z}]^T$

1 $\mathbf{X}_n \leftarrow \hat{\mathbf{X}}_{n-1} + \mathbf{1}_N \mathbf{a}^T (\mathbf{r} - \mathbf{s}_{n-1}) + \mathbf{\Psi}$

2 $\mathbf{w}_n \leftarrow e^{-\frac{1}{2}(\mathbf{X}_n - \mathbf{s}_{n-1})^T \mathbf{\Sigma}_w^{-1} (\mathbf{X}_n - \mathbf{s}_{n-1})}$

3 $\mathbf{s}_n \leftarrow \frac{1}{\mathbf{1}_N^T \mathbf{w}_n} \mathbf{X}_n^T \mathbf{w}_n$

Reviewer #1, Concern #5: ***References remain improperly formatted.***

Author Response: The authors appreciate the reviewer's attention to detail and constructive comments regarding the bibliography. We have conducted a thorough examination of the references to identify mistakes in the capitalization, abbreviation, and format of each reference.