GPS First Path Detection Network based on MLP-Mixers

Seung-Hyun Kong†, *Senior Member, IEEE*, Sangjae Cho†, and Euiho Kim*

***Abstract*—BPSK modulated GPS L1 CA signal is the most widely used GNSS signal to date, and the first path detection (FPD) of the conventional GPS L1 CA signals is the most challenging problem to ensure reliable GPS positioning in multipath environments. In this paper, we propose an FPD network (FPDN) based on multi-layer perceptron (MLP)-Mixer to extract the first path from the discrete autocorrelation function (ACF) output accurately with low computational cost. In addition, the proposed FPDN is useful in practice because it is robust to noise and achieves a high FPD performance without any prior assumption on the number of total incoming multipath, which is required for conventional signal processing-based FPD techniques. We compare the performance of the proposed FPDN to that of diverse conventional techniques, such as techniques based on narrow correlator, super-resolution, and some widely used CNNs such as VGGNet, ResNet, and U-Net, through simulations and field tests. As demonstrated, the proposed FPDN outperforms all of the compared FPD techniques in terms of the computational cost and accuracy for wide range of carrier-to-noise (C/N_0) ratios.**

***Index Terms*—GPS, Multipath, First path, DNN, MLP-Mixer.**

I. Introduction

Pseudorandom-noise (PRN) sequences are widely used for direct sequence spread spectrum (DSSS) systems including Global Navigation Satellite System (GNSS), because they allow a multiple access and an accurate code phase estimation of the received signals. Currently, the most widely used GNSS signal is the GPS coarse-acquisition (C/A) signal at L1 frequency (i.e., 1575.42 MHz), which uses the binary phase-shift keying (BPSK)-modulated Gold code as a PRN sequence. To detect ranging measurements from DSSS signals, a receiver first obtains autocorrelation function (ACF) output by correlating the received signal with a receiver-generated replica PRN sequence. Then, the receiver measures the path delay of the received signal based on the code phase of the peak of the ACF output, which becomes the range between the receiver and the transmitter. However, radio signals are subjected to refraction and reflection due to various obstacles, for example, high rise buildings and cars in the urban environments, where the line-of-sight (LOS) path signal can be significantly attenuated or may not be received, and multipath (echo) arrives at the receiver with an excess delay and a significantly decreased power in comparison to the LOS path [1]–[3]. As a result, we often experience a large ranging error from several meters to hundreds of meters. A GPS receiver typically uses an early-late (EL) correlator to measure the code delay of the first path (FP) and an EL correlator-based delay-locked loop (DLL) [4] to track the FP signal. In an open sky environment, where a strong LOS path is present, an accurate code delay can be estimated because the correlation peak of the FP (i.e., the LOS path) is significantly higher than those of multipath. However, in Non-LOS (NLOS) urban environments, the ACF output is constructed with multiple overlapping autocorrelation outputs of a number of multipath, in which case a correlator-based FP detection (FPD) technique may result in large ranging errors. To improve the code delay estimation accuracy, the Strobe correlator using multiple correlators has been proposed [5], but this technique does not improve the FPD performance in NLOS multipath channels. In addition, the multipath elimination (ME) technique utilizes the slope estimation around the detected path in the ACF output, but has performance degradation for large number of multipath signals [6].

In [7]–[10], maximum likelihood (ML)-based FPD techniques for multipath environments are proposed. In [7], a multicorrelator-based technique is proposed to estimate the delay and phase of individual paths from an ACF output in multipath channels: however, it is impractical to require prior knowledge on the number of total incoming multipath. Studies in [8], [9], and [10] introduce a grid search-based technique, a Rao-Blackwellization [11] technique, and Newton's method to reduce the computational costs of ML-based FPD techniques, respectively. The space-alternation generalized expectation-maximization (SAGE) algorithm [12] reduces the dimensions of channel parameters for the fast convergence of iterative estimation techniques, and, in [13], SAGE is applied to estimate the channel impulse response (CIR) in multipath channels. However, those ML-based FPD techniques require high computational costs in general, which makes them impractical for real-time applications. Meanwhile, super-resolution (SR)-based FPD techniques have been developed to estimate the FP from ACF output, such as multiple signal classification (MUSIC) [14] and estimation of signal parameters via rotation investigation (ESPRIT) [15]. These SR techniques require complex computations, such as the subspace decomposition of the signal correlation matrix, which makes their real-time implementation still difficult [16]–[18]. The least-square-based iterative multipath super-resolution (LIMS) technique tries to reduce computational cost by implementing iterative FPD utilizing the fact that the ACF output of a path is an equilateral triangle [19]. However, LIMS still requires a

†Co-first authors of equal contribution.

Seung-Hyun Kong and Sangjae Cho are with the CCS Graduate School of Green Transportation, Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, 305-701 (e-mail: {skong, sanje}@kaist.ac.kr)

*Corresponding author is Euiho Kim with the Department of Mechanical & System Design Engineering, Hongik University, Seoul, Korea, 121-791 (e-mail: euihokim@hongik.ac.kr)

number of iterations to find the FP, and the performance of MUSIC, ESPRIT and LIMS depends on prior knowledge on the number of multipath in the received signal. Recently, Deep Neural Networks (DNNs), such as Convolutional Neural Network (CNN) [20] and Multi-layer Perceptron (MLP) [21], have shown innovative performance in a wide range of tasks, including classification, detection, and localization. However, DNNs require a much larger computational cost than signal processing techniques, since there are millions of parameters used for final computation in the DNNs while the signal processing techniques such as LMIS require $O(P^3)$ multiplications [19], where P is the assumed number of incoming multipath. Despite the huge computational cost, DNNs have gained a lot of attention for their amazing performance and strong robustness against noise.

In this paper, we propose a novel FPD network (FPDN) to detect the first path in mulitpath environments such as urban and dense urban areas, where the LOS path is often blocked or seriously attenuated while a number of multipaths are arriving at the receiver. The proposed FPDN employs MLP-Mixer [22] block(s) as a global feature extractor. This is because we transform the ACF output into an image input and the MLP-Mixer has a strong capability in detecting global features (e.g., a large size feature on the input image) with high resolution, whereas CNN looses resolution when detecting global features at deeper layers and MLP is generally inferior to CNN in the detection tasks on an image. In addition, MLP-Mixer requires less computational cost than CNN. To demonstrate the superior performance of the proposed FPDN, we compare its FPD performance with a number of diverse GPS FPD techniques, including narrow correlator; super-resolution-based techniques, such as SAGE and LIMS; and DNN-based techniques, which are MLP and well-known CNN models, such as VGGNet [23], ResNet [24], and U-Net [25]. The FPD performance is compared through both simulations for various LOS and NLOS channels and field experiments in urban canyon environments.

The remainder of this paper is organized as follows. Section II defines ACF output in terms of multipath signal parameters and develops a mathematical expression for the discrete ACF output with respect to the sampling rate. Section III introduces the MLP-Mixer employed in the proposed FPDN. Section IV introduces the preprocessing scheme for the discrete ACF output in order to generate input images to the proposed technique (i.e., FPDN), and describes the FPDN in detail. In Section V, we compare the FPD performance of various neural networks with similar hyper-parameter configurations to show the advantage of the proposed FPDN in both the computational cost and FPD accuracy. Furthermore, we compare the performance of the proposed FPDN to that of ML-based and super resolution-based FPD techniques to prove that the proposed technique is valid and effective for real-world environments. Additionally, we apply the proposed FPDN to the measurements obtained from a software defined GPS receiver to test the positioning performance improvements in real multipath environments. Section VI draws the conclusion of this paper.

II. Discrete ACF Output

This section derives a mathematical expression for the discrete ACF output in terms of path delay, phase, and amplitude of multipath. In the bit interval $(l-1)T_b \le t < lT_b$, the data bit signal $D(t)$ with a bit rate of $R_b = 1/T_b$ is constant (i.e., $D(t) = D = +1$ or -1), and the baseband binary phase-shift keying (BPSK) modulated signal spread by a PRN sequence can be written as

$$s(t) = D \sum_{k=-\infty}^{\infty} \sum_{n=0}^{N_{code}-1} c\left[n - kN_{code}\right] \left[u\left(t - nT_c - kT_p\right) - u\left(t - (n+1)T_c - kT_p\right)\right], \tag{1}$$

where $c[n]$ is the $(n+1)$th code value of the binary phase PRN sequence having N_{code} code lengths with a code rate (i.e., chip rate) of $1/T_c$ (T_c is the BPSK chip width), $u(t)$ is a unit step function, and $s(t)$ has the period of T_p (i.e., $T_p = N_{code}T_c$) [26].

In the receiver, the autocorrelation function performs a correlation between the incoming signal $s(t)$ and a receiver replica code signal $c_R^*(t-\tau)$ for a coherent correlation interval of T_{co} that is much smaller than T_b. The normalized expected ACF output can be expressed as

$$\begin{aligned} R_0(\tau) &= \frac{1}{T_{co}} \int_0^{T_{co}} s(t) c_R^*(t-\tau) dt \\ &= \begin{cases} \frac{T_c - |\tau|}{T_c}, & 0 \le |\tau| < T_c \\ 0, & \text{otherwise,} \end{cases} \end{aligned} \tag{2}$$

where $(\cdot)^*$ is the complex conjugate function, τ is the code delay hypothesis being tested, and we assume $T_{co} \ll T_b$ and $D = 1$. The transmitted signal $x(t)$ is the signal $s(t)$ modulated with a carrier at carrier frequency f_c with an unknown phase θ_0 at the time of antenna transmission as

$$x(t) = s(t) \cos\left(2\pi f_c t + \theta_0\right). \tag{3}$$

As the signal $x(t)$ in (3) arrives at the receiver in urban environments, it experiences a multipath channel whose the time domain CIR can be modeled as

$$h(t) = \sum_{p=0}^{P-1} C_p \delta\left(t - \tau_p\right), 0 \le t < T_{co}, \tag{4}$$

where P is the total number of paths,

$$C_p = a_p e^{j\theta_p} \tag{5}$$

is the complex channel coefficient of the $(p+1)$th path with amplitude a_p ($\le$ 1) and phase θ_p, $\delta(t)$ is the dirac delta function, and $\tau_p (> \tau_{p-1})$ is the $(p+1)$th path delay. Using (4), the baseband equivalent received signal in the continuous time domain can be found as

$$\begin{aligned} y_B(t) &= s(t) * h(t) + n(t), 0 \le t < T_{co} \\ &= \int_{-\infty}^{\infty} s(\tau) h(t-\tau) d\tau + n(t), \end{aligned} \tag{6}$$

where '$*$' is the convolution operation, and $n(t)$ is a complex additive white Gaussian noise (AWGN) process with two-sided power spectral density (PSD) $N_0/2$. The signal $y_B(t)$ is then despread by the receiver generated spreading code signal $c_{R(t)}$.

Then, the normalized ACF output in the continuous form can be expressed as

$$\begin{aligned} R(\tau) &= \frac{1}{T_{co}} \int_0^{T_{co}} y_B(t) c_R^*(t-\tau) dt \\ &= \sum_{p=0}^{P-1} C_p R_0 \left(\tau - \tau_p\right) + w(\tau), \end{aligned} \tag{7}$$

where $w(\tau)$ is a zero-mean Gaussian process with autocorrelation $E\{w(\tau)w(\lambda)\} = (N_0/2T_{co})R_c(\tau-\lambda)$, and R_0 is the noise-free autocorrelation function of the code signal. When a sampling frequency $f_s = 1/T_s$ is used for sampling the continuous signal $y_B(t)$, the sampled ACF output $R(nT_s)$ becomes

$$R\left(nT_s\right) =$$

$$\begin{cases} \sum\limits_{p=0}^{P-1} C_p R_0 \left(nT_s - \tau_p\right) \\ \qquad + w\left(nT_s\right), & \left\lceil \frac{\tau_0 - T_c}{T_s} \right\rceil \le n \le \left\lfloor \frac{\tau_{P-1} + T_c}{T_s} \right\rfloor, \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

where n is the sample index defined in a range where the sampled ACF output $R(nT_s)$ is nonzero, and τ_p can be further expressed with a quotient d_p and a remainder ϵ_p ($0 \le \epsilon_p < T_s$) of T_s such that

$$\tau_p = d_p T_s + \epsilon_p, \tag{9}$$

where

$$d_p = \left\lfloor \frac{\tau_p}{T_s} \right\rfloor \tag{10}$$

is the path delay of the p-th path in the unit of samples, and

$$\epsilon_p = \tau_p - \left\lfloor \frac{\tau_p}{T_s} \right\rfloor T_s \tag{11}$$

is the residual delay of the p-th path caused by the finite sampling rate. The magnitude $|C_p|$ of the first path ($p = 0$) in the LOS channel is assumed to be the maximum value of $R(\tau)$ in (7), because the LOS path is generally by far the strongest path. In general, $\epsilon_p \neq 0$, and the peak value of the discrete ACF output $R(nT_s)$ is smaller than the peak value of the continuous ACF output $R(\tau)$. For this reason, measurement accuracy of the FP from the discrete ACF output can be improved for smaller T_s, but increasing the sampling rate f_s results in an increased complexity of the receiver.

III. MLP-Mixer for Feature Extractor

In many applications that require feature detection and object recognition, CNN has shown the highest performance [20] so far. Starting with AlexNet's ILSVRC-2012 victory [27], CNN has been one of the most researched areas in the world. VGGNet achieves a good performance with a simple architecture using a CNN-based feature detector and a fully connected (FC) layer-based classifier. For example, VGGNet has been widely used as a backbone of many DNNs [23]. GoogLeNet, introduced in year 2014, has deeper layers than VGGNet and achieves the higher recognition performance than VGGNet, but it requires less computation by sequentially utilizing inception modules. An inception module reduces the amount of computation by concatenating multiple feature maps produced by various convolution filters in a parallel structure. ResNet [24] presents a method to obtain a better recognition by adding skip (or residual) connections to compensate for the possible degradation that occurs at a deeper neural network. U-Net [25] has a symmetric U-shape structure consisting of the context path (encoding path) that extracts the context of the input image and the extracted context (decoding path), which improves the context recognition of the input image and the localization performance of the object simultaneously. The U-shape structure provides a high localization performance, which is still used as a backbone for many DNNs for semantic segmentation [28]–[31].

MLP-Mixer [22] divides the input images into multiple patches and extracts the global features through correlations between all tokens and between all channels without reducing the data size, which leads to a superb performance in detecting global features (such as the ACF envelope) with high resolution. An MLP-Mixer network can have mulitple blocks (i.e., the number of MLP-Mixer blocks $N_M \ge 1$), but Fig. 1 illustrates a single MLP-Mixer block (N_M=1) as an example, where the input image has a size $H = W = 160$ and the patch size $Q \times Q = 8 \times 8$. Therefore, there are $HW/Q^2 = S = 400$ tokens, and each token is produced by an MLP of size $2Q^2 \times N_{CH}$ for the two patches at the same location of the I/Q-channel images, where $(2Q^2 = 128)$, $N_{CH} = 256$ is the number of channels, and the MLP is shared for all pair of patches. Note that H and W are determined by the image input M_F, and that Q, N_{CH}, and N_M are hyperparameters of the network to be determined through numerous simulations.

However, in [32], the best performance is found when the input image is divided into 16×16=256 patches for image classification tasks, in comparison to other cases when the input image is divided into a smaller number of patches than 256. Exploiting this ablation study in [32] and considering the size of M_F ([160×160]), we may use $Q = 8$ to produce 400 patches or $Q = 10$ to produce 256 patches. On the other hand, since it is useful to represent a patch in a semantic dimension larger than the number of pixels in a patch (i.e., $2Q^2$), we use $N_{CH} = 2 \times 2Q^2$ so that a patch can be represented over a 2 times wider dimension than the number of pixels of the patch, which can be large enough for the black and white images like M_F. As a result, the I/Q-channel input images can be converted into a 'patch × channel' table of a size $S \times N_{CH} = 400 \times 256$ for $Q = 8$ or $S \times N_{CH} = 256 \times 400$ for $Q = 10$. From simulations, we observe that $Q = 8$ results in only slightly better performance, so we use $Q = 8$ in the paper.

Each MLP-Mixer block has Token Mixing and Channel Mixing steps for the global feature extraction as shown in Fig. 1. The Token Mixing step employs two sequential MLPs of sizes 400×256 and 256×400 to perform correlations between any two patches across the S patches in the same channel, which should be performed for each of N_{CH} channels. After the Token Mixing, we apply the skip (i.e., residual) connection (SC) [24] and the layer normalization (LN) to prevent the gradient vanishing problem and Internal Covariance Shift (ICS), respectively. The Channel Mixing step utilizes two sequential

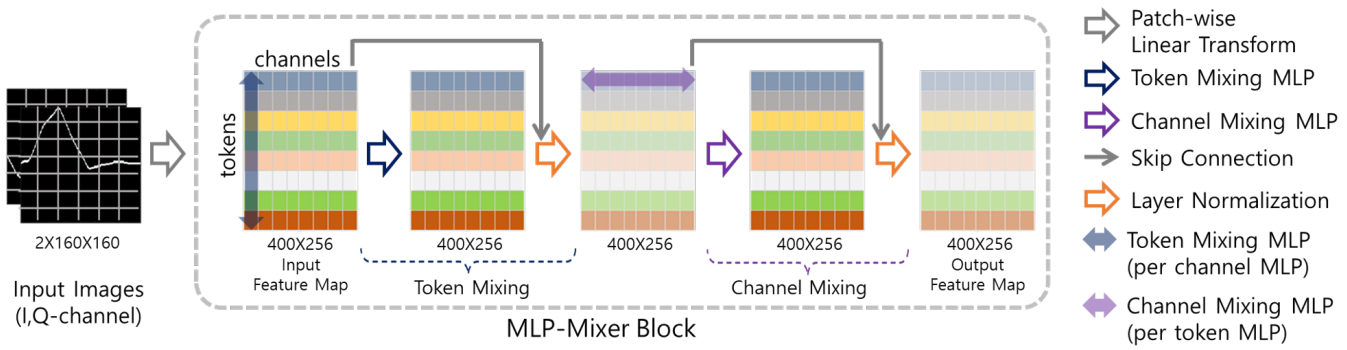


Fig. 1. MLP-Mixer block used as the feature extractor of the proposed FPDN.

MLPs of sizes 256×512 and 512×256 to perform correlations between any two patches across the N_{CH} channels in the same token, which should be performed for each of $S=400$ tokens, and followed by SC and LN layers. As a result, the output of the MLP-Mixer block is a feature map, of the same size 400×256 to the input, that shows the correlation between patches across the tokens and channels.

In general, an MLP-Mixer-based feature extractor (FE) has two strong advantage over the CNN-based FEs. Firstly, MLP-Mixer can detect global features with higher resolution than CNN. This is because CNN extracts local features at shallow layers and recognizes global features by combining the local features at deeper layers, where the image data (i.e., feature-map) resolution becomes poor, whereas an MLP-Mixer does not reduce the feature-map size so that there is no loss of resolution. Therefore, considering that the ACF output envelope shown in the I/Q-channel ACF output image is a thin line of subtle shape and that the FP can be detected by investigating the overall and detail shape of the ACF output envelope, the MLP-Mixer can be an adequate FE for FPD. Secondly, while CNN-based FEs do not have any self-attention mechanism introduced in ViT [32] so that they focus on every region of the input image with equal attention whether the region is from foreground or background, the MLP-Mixer has a low computational self-attention realized with a single MLP. In the literature, the self-attention mechanism has been found very useful for a high object detection performance, since ViT achieves a superior performance of 88.55% top-1 accuracy on ImageNet. However, ViT requires quadratic computational costs due to the self-attention using three MLPs, while MLP-Mixer has a similar performance of 87.94% top-1 accuracy on ImageNet with much less computational costs. Therefore, the MLP-Mixer achieves one of the top object detection performance for an image input with low computational cost, which is important and necessary for low cost GPS receivers.

IV. FPDN BASED ON MLP-MIXER

In this section, we propose the MLP-Mixer-based FPD network (FPDN), which detects the FP from the sampled ACF output. The proposed FPDN is designed to have a low computational complexity because it does not require any prior knowledge regarding the number of multipaths or any iteration process. In the following subsections, we introduce the preprocessing technique for the ACF output samples to build input images to the FPDN and then FPDN based on the MLP-Mixer.

A. Preprocessing ACF output image

A discrete ACF output produced by a GPS receiver is two $N\times1$ vectors showing the I/Q-channel ACF output envelopes at consecutive sampling points as in Fig. 2 (a) and (b). To detect the FP from the two ACF output envelopes, the discrete ACF output of the I/Q-channel is converted into a sparse binary image matrix M_B of size $[m_r \times m_c]$ as shown in Fig. 2 (c) and (d). That is, a sparse matrix M_B has non-zero (i.e., ones) values at the matrix element corresponding to the ACF output envelopes in (a) and (b), where m_c is the temporal window size that should be set wide enough to contain the whole ACF envelope. Therefore, m_c should contain the samples within a temporal range $[-T_c, T_c]$ (i.e., 2-chip interval) of the discrete ACF output peak. Since it is found that multipath arrives with excess delays seldom later than 1 chip from the LOS path [1], [26], a total of 4 chips (i.e. 1.5 chips before and 2.5 chips after the ACF peak delay) are used as the ACF output window width, which results in $m_c = 4 \times T_c/T_s$. On the other hand, m_r represents discrete levels to express the amplitude of the ACF output. Because there are T_c/T_s ACF output samples per chip, m_r must be bigger than T_c/T_s to express the various slopes along the ACF output envelope. Therefore, in the proposed technique, we set $m_r = 4T_c/T_s$ to represent 4 possible slopes between the neighboring sample points along the ACF output envelope. In the following, $m_r = m_c$ is assumed and set to m.

In summary, the preprocessing to develop input images from the I/Q-channel ACF outputs consists of the following steps. The first step generates an all-zero square matrix M_Z of size $[m \times m]$, and the second step quantizes the amplitude of the normalized ACF output R(nTs) into m levels to obtain a quantized discrete ACF output $R_Q[n]$ such that $R_Q[n] \in \{0, 1, \ldots, m-1\}$ for all $n \in \{0, 1, \ldots, m-1\}$. In the third step, let $M_Z[n, R_Q[n]] = 1$ (for all n) and save the result to a sparse binary image matrix M_B so that the nonzero (i.e., 1) elements of M_B depicts the ACF output envelope. In the fourth step, we insert K_z zero-columns and K_z zero-rows after each column and row of the M_B, respectively, to develop the final input image matrix M_F, where

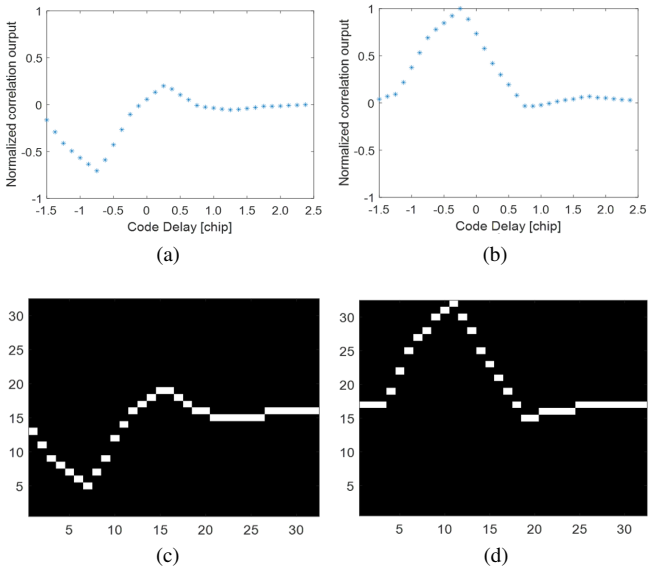


Fig. 2. Sampled ACF outputs for I/Q-channel shown in (a) and (b), respectively, and resulting sparse matrix M_B for I/Q-channel shown in (c) and (d), respectively.

$$K_z = \frac{40T_s}{T_c} \tag{12}$$

is to make M_F K_z^2 times more sparse than M_B and to provide M_F enough spacing between the non-zero (i.e., 1) elements. In this paper, the size of the input image matrix (i.e., zero-padded M_F) is fixed to $[160 \times 160]$ (that is, $m \times K_z = 160$).

For example, for ACF output obtained with $f_s = 10/T_c$, M_B has a size $[40 \times 40]$, and $K_z = 4$ based on (12). Each column and row of M_B is added with K_z zero-columns and K_z zero-rows, respectively, so that M_F has a size $[160 \times 160]$ as shown in Fig. 3 (a). In this generated image, non-zero elements occupy only 0.19% of the total image, and this sparsity of the input image causes a poor learning performance. To mitigate the sparsity, in the fifth step of the preprocessing, we apply a linear interpolation between non-zero samples of M_F, as shown in Fig. 3 (b), so that there are 160 non-zero pixels in M_F. In the sixth (last) step, to improve the recognition of the ACF envelope shown in the interpolated image in the neural network, we thicken the ACF envelope by changing the value of the pixels within 2 pixels below the ACF envelope to one in the interpolated image. The final input image M_F to the proposed FPDN would look like the one shown in Fig. 3 (c).

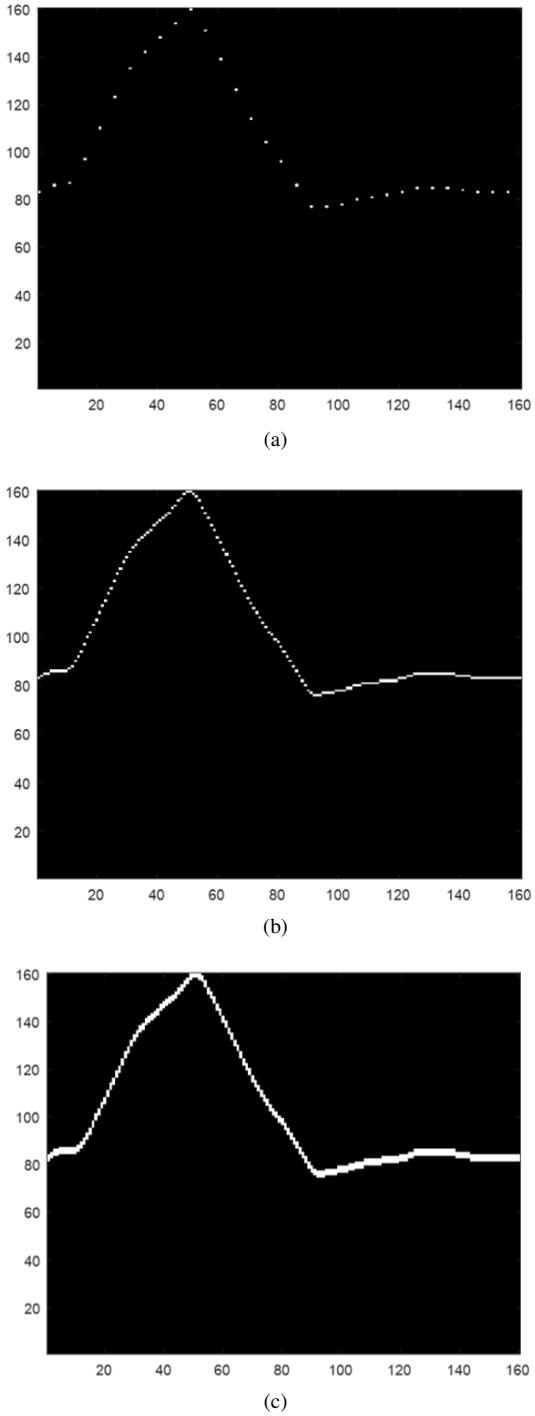


Fig. 3. $[160 \times 160]$ size images after zero-padded (a), interpolated (b), and thickened (c).

B. Proposed FPDN based on MLP-Mixer

The overall structure of the proposed FPDN is shown in Fig. 4. The preprocessed ACF outputs for I/Q-channels, M_{F_I} and M_{F_Q}, respectively, are concatenated and passed through an initial convolutional layer with 256 filters of size equal to the patch size (Q^2) and stride size Q, i.e., patch-wise linear transform. The result is a table (i.e., feature-map) of size 400 patches $\times$ 256 channels that becomes the input to the MLP-Mixer blocks. The table is processed through N_M sequential MLP-Mixer blocks, where each MLP-Mixer block is illustrated in Fig. 4. The number N_M is a network hyperparameter to be determined to satisfy the required performance and computational cost. After N_M sequential MLP-Mixer blocks, the final table of size 400×256 passes through a global average pooling (GAP) layer [33] to yield a vector of size 400×1, which is to extract per-patch estimate by compressing over all channels. Then, the vector is passed through 2 consecutive MLPs of size 400×50 and 50×1 to extract a compressed feature and to regress the final single estimate, respectively. Because we design the final layer to produce the output $\hat{y}$

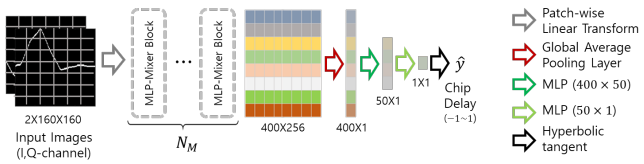


Fig. 4. Proposed FPDN using Sequential MLP-Mixers.

within a range $[-1, 1]$ (in chips), we employ the hyperbolic tangent (tanh) function as the activation function that has the same output range. Note that we apply the same hyperbolic tangent function for the activation function of the other DNNs considered in this paper due to the same reason. The loss function J to train and evaluate the proposed network is based on the root mean square error (RMSE) as

$$J\left(\theta_j\right) = \sqrt{\frac{1}{\mathrm{n}} \sum_{i=1}^{n} \left(y_i - \hat{y}_i\right)^2}, \tag{13}$$

where n(=64) is the batch size of the training data, and θ_j is the set of learning parameters of the FPDN tuned at the j-th training epoch. The training of the network is executed on the NVIDIA RTX 3080, with an Adam optimizer [34] and a learning rate of 0.001. We use Xavier normal as the kernel initializer [35].

When FP delay is estimated from I/Q pair of input images, the estimation error due to the sampling resolution can be as large as $T_s/2$, which can be 14.7m for a GPS receiver with $T_s = T_c/10$. However, in this paper, the network is trained to enhance the resolution in the FP delay estimation to as small as 1ns (30cm).

V. Performance Evaluation

In this section, we evaluate the FP delay-estimation performance of the proposed FPDN against multipath in mobile environments, such as urban and dense urban, using various simulations and field tests. First, we model the distortion of ACF outputs (such as rounded top and smooth edge corners in the ACF output) due to the pre-correlation bandwidth (PCBW) B_W for accurate performance evaluation [36]. Generally, a signal passing through a bandlimited filter loses high-frequency components, which smooths the sharp triangular ACF peak to a rounded peak and smoothed edge corners. Therefore, we consider the effect of PCBW on the FPD performance. Note that since the ACF output distortion we investigate in this section is not due to the multipath, we assume that $y_B(t)$ is from a single path (i.e., LOS path signal) and $P = 1$ in (4), in the following analysis.

Let $H(f)$ and $Y_B(f)$ be the Fourier transforms of the impulse response of an ideal low-pass filter $h(t)$ with PCBW B_W and the baseband signal $y_B(t)$ in (6), respectively. Since we analyze the distortion of the ACF output according to PCBW, we assume that $y_B(t)$ is the LOS path signal (i.e., $P = 1$ in (4)). The Fourier transform of the ACF output $R(t)$ in (7) can be expressed as

$$\begin{aligned} R_F(f) &= Y_B(f) S^*(f) \\ &= H(f)|S(f)|^2 + W(f), \end{aligned} \tag{14}$$

where $(\cdot)^*$ represents the complex conjugate,

$$H(f) = \begin{cases} 1, & |f| < B_w \\ 0, & \text{otherwise} \end{cases} \tag{15}$$

is a rectangular function with a bandwidth B_W in the frequency domain, and $S(f)$ is the Fourier transform of the ranging signal $s(t)$ in (1). When the noise spectral density, $W(f)$, is low enough, $R(t)$ can be simplified as

$$\begin{aligned} R(t) &= \mathcal{F}^{-1}\left\{H(f)|S(f)|^2\right\} + \mathcal{F}^{-1}\{W(f)\} \\ &\cong \mathcal{F}^{-1}\left\{H(f)|S(f)|^2\right\} \\ &\cong B_W \,\mathrm{sinc}\left(\pi B_W t\right) * R_0(t), \end{aligned} \tag{16}$$

where $\mathcal{F}^{-1}\{\cdot\}$ represents the inverse Fourier transform, $R_0(t)$ is the ideal ACF output with infinite bandwidth, $\mathrm{sinc}(\pi B_W t)$ is a normalized sinc function with a null-to-null bandwidth B_W, and , and the sinc function $\mathrm{sinc}(\cdot)$ is defined as $\mathrm{sinc}(x) = \sin(x)/x$.

Using the analysis in [37], $R(t)$ can be approximated as

$$R(t) \cong 2\beta \int_{-1}^{1} \mathrm{sinc}(2\beta\tau) R_0(\mathrm{t} - \tau) d\tau \tag{17a}$$

$$\cong \begin{cases} \alpha \exp\left\{-t^2/\delta^2\right\}, & \text{for } \beta = 1 \\ R_0(t), & \text{for } \beta \gg 1, \end{cases} \tag{17b}$$

where $\beta = B_W T_C$ so that $\beta = 1$ is for a receiver that has B_W equal to the chip rate of the spreading code, and $\beta = 10$ is for the typical GPS C/A signal receivers [37]. The quantities $\alpha = 0.903$ and $\delta = 0.63$ are found in [37] for the Gaussian approximation ($\beta = 1$) in (17b). In the simulations, we use various sampling frequencies f_s such as $2/T_C$ (i.e., 2 samples per chip (SPC)), $4/T_C$, $8/T_C$ and $10/T_c$, for (null-to-null) B_W of 2.046MHz, 4.092MHz, 8.184MHz, and 10.23MHz, respectively. Note that we assume $\beta = 1$ in the case of B_W=2.046MHz and $\beta \gg 1$ for B_W larger than 2.046MHz.

In general, since the code phase search resolution of the GPS C/A signal acquisition function is $0.5T_c$, the delay of the true FP to be estimated by the FPDN can be uniformly distributed over the interval $[-0.25T_c, 0.25T_c]$ around the detected FP delay by the acquisition function. And since multipath delays are rarely larger than $1T_c$ (of the GPS L1 C/A code) as found in [1] and [26], we can marginally define that the true FP delay τ_0 is not earlier than -0.5T_c from the detected FP delay and the last path delay $\tau_{(P-1)}$ is not later than $1.5T_c$ from the detected FP delay. Considering that an ideal ACF output of a path spans from $-T_c$ to $+T_c$ around the peak and that $\tau_{(P-1)} - \tau_0 < 2T_c$, the ACF output affected by the incoming signal paths is well within the window of $4T_c$ width, and the total number of samples N_{sample} within the window becomes

$$N_{sample} = 4 \times \frac{T_c}{T_s}. \tag{18}$$

In the following subsections, we use the following assumptions for simulations:

- Short correlation interval T_{co} = 1ms is assumed for FPDN and other DNN-based techniques, whereas, for LIMS and SAGE, T_{co} = 10ms is assumed as in [19]. This provides a strong advantage to LIMS and SAGE as the SNR in the ACF output increases by 10 times.

- We train the DNN-based FPD techniques (i.e., MLP, VGGNet, U-Net, ResNet, and MLP-Mixer) for all of the channels considered in this section, and, then, the trained DNN-based techniques are tested to develop the performance plots for specific channels (fixed number of multipath channels and stochastic channels) considered in the following subsections. Note that the test dataset is developed separately so that it contains no data that belong to the training dataset.
- Considering the sampling rate is a hardware-fixed parameter, we develop separate DNN-based techniques for different SPCs. For example, we develop three separate FPDNs for 2, 4, and 8 SPCs and use them to develop performance plots in the following subsections.

For the training dataset, we develope 5,000 Monte Carlo realizations of ACF output image (M_F) for each of five C/N$_0$ values, each of five channels (used in the following subsections), and each of three sampling rates. Therefore, the training dataset is composed of 5,000×5×5×3=375,000 images of M_F. For test dataset, we develope only additional 10% more of the training dataset.

A. Multipath Channels

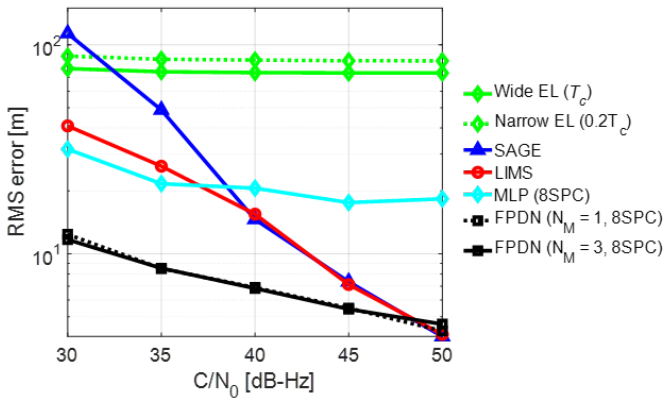


Fig. 5. RMS errors for two-path non-fading channel.

The first test channel for a performance evaluation is a simple two-path non-fading channel [19] with channel parameters of $C_{CH} = [1/\sqrt{2}, 1/\sqrt{2}]$ and $\tau_{CH} = [\tau_0, \tau_0 + T_c/2]^T$, where C_{CH} and τ_{CH} are a set of real amplitudes and delays, respectively. We exploit the two-path channel that is useful to show the intelligence of the proposed FPDN while other techniques show a strong degradation. In the delay set, it is assumed that τ_0 is uniformly distributed over $[-T_c, T_c)$, which corresponds to the subrange $[0.5T_c, 1.5T_c)$ within the full temporal window, $[0, 4T_c]$, of the ACF output image input to the proposed FPDN. In addition, the simulations described in this subsection assume that the carrier-to-noise density ratio (C/N$_0$) of the GPS L1 C/A signal is uniformly distributed from 30dB-Hz (weak) to 50dB-Hz (strong). The signal-to-noise ratio (SNR) of the signal for a given C/N$_0$, PCBW B_W, and the receiver processing gainP_G can be expressed as

$$\mathrm{SNR[dB]} = \mathrm{CN_0[dB-Hz]} + P_G - 10\log_{10} B_W\mathrm{[Hz]}, \quad (19)$$

TABLE I
COMPARISON OF SIZE AND COMPLEXITY OF MLP, MLP-MIXER1, AND MLP-MIXER3

	Number of Parameters	FLOPS
MLP	470k	1.2M
MLP-Mixer1	520k	170M
MLP-Mixer3	1.46M	485M

where $P_G = 10\log_{10}(T_{co}/T_s)$. As found in [19], the EL discriminator is suitable for the equilateral triangular shape ACF envelope in the open-sky LOS environments, but it shows a severe performance degradation for the two-path non-fading channel. Fig. 5 shows the FP delay estimation performance of the narrow correlator, wide correlator, LIMS, SAGE [11], MLP-based technique, and two FPDNs with $N_M = 1$ and 3 for the two-path non-fading channel, where the PCBW of 10.23MHz is assumed for SAGE and LIMS and PCBW is 8.184MHz for DNN-based techniques. The conventional narrow correlator and wide correlator show RMS errors of approximately $T_c/4$ regardless of C/N$_0$. This is because the two-path channel causes an ACF output that has a flat top between the two paths so that the EL discriminators could choose any points around the middle of the flat top with uniform probability. For LIMS and SAGE, it is found that the FPD performance monotonously degrades as C/N$_0$ decreases. Note that as discussed in [19], SAGE and LIMS shows almost the same performance to the CRLB (Cramer-Rao Lower Bound) for C/N$_0 >$ 40dB-Hz and the CRLB becomes the minimum at C/N$_0$ = 50dB-Hz, where the effect of noise becomes almost negligible. In general, DNN-based techniques can achieve better performance than the mathematical rule-based techniques such as narrow and wide correlators, SAGE, and LIMS, because they recognize the FP based on not only the detected peak of the ACF output but also understanding the overall shape of the ACF output envelope. In other words, DNN-based techniques learn to figure out the FP from various ACF output envelopes produced by various multipath channels and noise levels, which is possible as the techniques experience a huge number of ACF output envelopes. The performance of the proposed FPDN shown in Fig. 5 supports this claim.

However, the proposed FPDN shows much better performance than the MLP technique (described in Appendix), because MLP-based techniques have inferior performance to CNN-based techniques when recognizing features from an image input in general. Note that the proposed FPDN shows slightly lower accuracy than the SAGE and LIMS when C/N$_0$ = 50dB-Hz. This is because SAGE and LIMS are based on mathematical rules to estimate the peak with infinite precision so that the accuracy is increased monotonously with higher C/N$_0$, whereas the FPDN is data-driven (i.e., experience-based) by the input with fixed image resolution so that the precision is limited. Note also that even if there is a slight accuracy degradation at C/N$_0$ = 50dB-Hz, the proposed FPDN shows a strong robustness against noise so that there is a lowest degradation at lower C/N$_0$ levels, which is because of the data-driven approach (i.e., experience with huge

amount of various ACF output data) and FP detection based on recognition of various detected features of the ACF output envelope. In the result shown in Fig. 5, we can find that MLP-Mixer with $N_M = 1$ is enough to detect the FP in the simple two-path channel, as MLP-Mixer with $N_M = 3$ doesn't show a noticeable improvement. Table I summarizes the number of network parameters and floating-point operations (FLOPS) that define the amount of computation for MLP [21], MLP-Mixer1 (i.e., FPDN with $N_M = 1$), and MLP-Mixer3 (i.e., FPDN with $N_M = 3$) [22].

Table II summarizes two types of multipath channels [19] tested for the second performance comparison. Channel-A represents an LOS channel with two additional low power paths with relative delays and random phases. We assume that the phases of the three paths are independent, since the phase of the LOS path is uniformly distributed due to the unknown carrier phase of the receiver replica signal, and that the delay distributions of the two paths are uniform within a small range. Channel-B is a four-path NLOS channel with relative delays and random phases, where the 3rd path has the highest power than other three paths. In both channels, the FP delay τ_0 is uniformly distributed over $[-T_c, T_c)$ as assumed in the two-path channel simulations. The FPD performance comparison for these two channel environments is performed for SAGE, LIMS, and MLP, MLP-Mixer with $N_{Mixer} = 1$, and a few well-known CNN-based networks such as VGGNet [23], ResNet [24], and U-Net [25]. We test SAGE and LIMS for two PCBWs and corresponding SPCs: 2 and 8 SPCs for 2.046MHz and 8.184MHz PCBWs, respectively, and all DNN-based techniques for three PCBWs and the corresponding SPCs: 2, 4 and 8 SPCs for 2.046MHz, 4.092MHz, and 8.184MHz PCBWs, respectively. The details of the MLP and CNN-based networks are provided in the Appendix. Table III lists the parameters and FLOPS for VGGNet, ResNet, and U-Net used for the comparison.

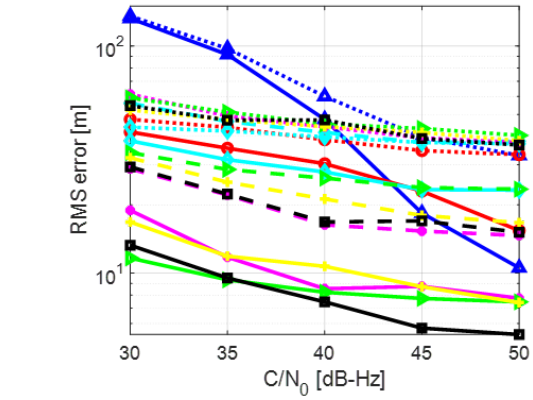
TABLE II
MULTIPATH CHANNEL PARAMETERS

Path	Channel-A			Channel-B		
	Relative power (dB)	Delay	Random Phase	Relative power (dB)	Delay	Random Phase
1	0	τ_0	Y	-7.0	τ_0	Y
2	-5.0	$\tau_0 + [0.1T_c, 0.3T_c)$	Y	-7.0	$\tau_0 + [0.1T_c, 0.3T_c)$	Y
3	-10.0	$\tau_0 + [0.3T_c, 0.5T_c)$	Y	0	$\tau_0 + [0.3T_c, 0.4T_c)$	Y
4	-	-	-	-2.2	$\tau_0 + [0.5T_c, 0.7T_c)$	Y

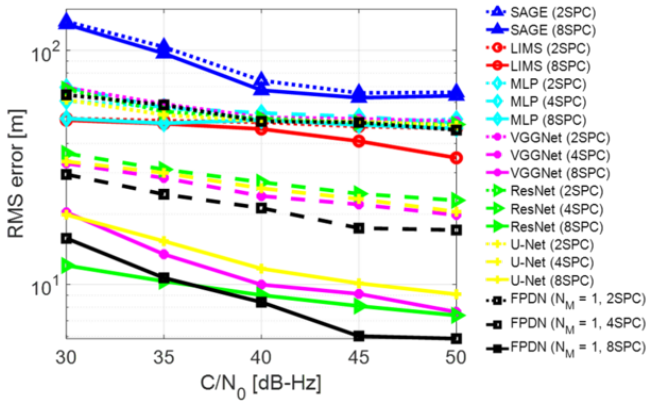
Fig. 6 shows the comparison of FPDN performance with other techniques for the multipath Channel-A and Channel-B summarized in Table II. In general, the FPD performance is generally better for Channel-A due to the presence of strong LOS path, and the FPD performance of all of the tested techniques degrades as PCBW and C/N$_0$ decrease for both Channel-A and Channel-B. As shown, LIMS, SAGE, and MLP

TABLE III
COMPARISON OF MODEL SIZE AND COMPLEXITY OF CNN-BASED MODELS

	Number of Parameters	FLOPS
VGGNet	2M	0.9G
ResNet	2M	1.0G
U-Net	2.2M	9.1G



(a)



(b)

Fig. 6. RMS errors for (a) channel-A and (b) channel-B.

show worse performance regardless of the sampling rate and C/N$_0$ than DNN-based techniques, even if the performance of LIMS and SAGE for Channel-A shows an improvement at C/N$_0 \geq 40$. Similar to the results shown in Fig. 5, the poor performance of LIMS and SAGE is because they are based on mathematical rules that depend on a number of assumptions and conditions, for example, assumed number of multipath and accuracy of the estimated parameters, and the poor performance of MLP is expected as the MLP-based techniques have inferior performance to CNN-based techniques when recognizing features from image inputs.

In contrast, DNN (MLP, VGGNet, ResNet, U-Net, MLP-Mixer1)-based (FPD) techniques achieve superior performance over the entire C/N$_0$ range than SAGE, LIMS, and MLP of the same sampling rate. This is because DNN-based techniques are data-driven (i.e., experience with huge amount of various ACF output data) and their FPD is based on recognition of various detected features of the ACF output envelope. Among the CNN-based (FPD) techniques (i.e., VGGNet, ResNet, and U-Net), ResNet-based technique shows slightly better performance than VGGNet and UNet-based techniques only for a high sampling rate (e.g., 8SPC). This is because the ResNet utilizes the residual (or skip) connection to maintain the details (i.e., resolution) of the input image to the deeper layers, which leads to a more accurate feature extraction performance than

both VGGNet and UNet which loose the resolution of the input image at deeper layers due to the compression. However, when the sampling rate is not high (e.g., 2SPC or 4SPC), there is no advantage from using the residual connection in the ResNet so that the performance of VGGNet, ResNet, and U-Net becomes similar.

Comparing the proposed FPDN (i.e., MLP-Mixer-based technique) to other CNN-based techniques, the proposed FPDN shows a better accuracy. This is expected, since MLP-Mixer has an enhanced capability to recognize global features (such as the overall envelope of ACF output) on the input image without loss of the input image resolution. In fact, most of the conventional CNNs are good at recognizing local features and trying to fit the detected local features into a global feature at deeper layers, where the resolution becomes poorer. Note that the proposed FPDN is only slightly better for high $C/N_0(\geq 40$dB-Hz) and high sampling rate, however, as shown in Table I and Table III, the number of parameters and the computational cost of the proposed FPDN are significantly lower than the ResNet. Based on the performance in Fig. 6 and the computational cost analysis in the tables, we demonstrate the strong advantage of the proposed FPDN.

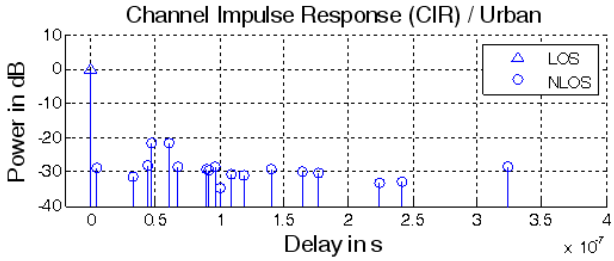
It should be noticed that the performance of all of the DNN-based techniques (i.e., MLP, VGGNet, ResNet, U-Net, and MLP-Mixer1) for Channel-A shows almost the same performance to Channel-B, even if Channel-B is a more harsh channel than Channel-A. This is an interesting observation of the DNN-based techniques, which appears at the results for stochastic multipath channels in the next subsection as well. This can be explained from the fact that the performance of a neural network is mainly determined by the amount of experience (i.e., training), and, therefore, when a network is sufficiently trained, the performance for the both channels has to be similar.

B. ITU-R P.681 Earth-Space Land Mobile Channel Model

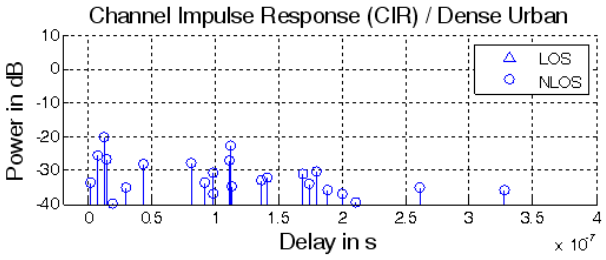
To compare the FPD performance in stochastic multipath channels similar to those in urban environments, we use the earth-space land mobile multipath channel (LMMC) model, which is recommended by the International Telecommunication Union (ITU) for mobile GNSS channels and standardized in ITU-R P.681-7 [38]. This model produces multipath based on ray-tracing in a virtual urban canyon environment, where the layouts of buildings, trees, and poles on the two sides of roads are located and sized according to its specified probability distribution. We consider two scenarios, urban and dense urban, which are classified according to the stochastic height and density of buildings in the virtual urban environment, and the parameters for both environments are summarized in Table IV. Fig. 7 shows examples of CIRs for the two considered environments. Unlike the two-path channel model and channel-A and channel-B described in subsection V-A, the number of multipath, their amplitudes, delays, and phases are generated in the LMMC model based on the probabilistically defined surrounding environments for a given vehicle speed. In both channels, LOS paths are seldom observed, but they occur relatively more often in the urban environment than the dense urban environment because of lower building heights used for simulations.

TABLE IV
EARTH-SPACE LAND MOBILE CHANNEL MODEL DENSE URBAN, URBAN PARAMETER [38]

Parameter	Value (max, min, mean, sigma)
Car maximum speed [km/h]	50
Satellite elevation [Deg]	30
Satellite azimuth [Deg]	-45
Carrier frequency [GHz]	1.57542
Antenna Height [m]	2
Road width [m]	15
Building Width [m]	-, 20, 30, 10
Building Height (Dense urban) [m]	150, 50, 100, 30
Building Height (Urban) [m]	100, 10, 50, 30
Gap between buildings [m]	-, 10, 30, 20
Building gap likelihood (Dense urban)	0.1
Building gap likelihood (Urban)	0.4
Tree height (Dense urban) [m]	8
Tree height (Urban) [m]	7
Tree diameter (Dense urban) [m]	5
Tree diameter (Urban) [m]	4
Tree trunk length [m]	2
Tree trunk diameter [m]	2
Tree Attenuation [dB/m]	1.1
Tree distance [m]	-, -, 40, 5
Pole height (Dens urban) [m]	10
Pole height (Urban) [m]	9
Pole diameter [m]	0.2
Pole distance [m]	-, -, 25, 10



(a)



(b)

Fig. 7. Channel Impulse Responses for (a) urban environment and (b) dense urban environment

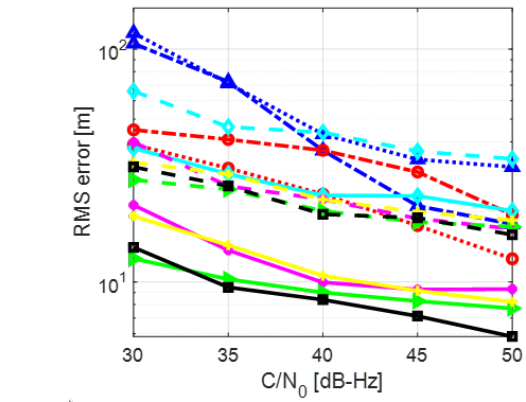
Fig. 8 shows the FPD performance results of LIMS, SAGE and MLP, VGGNet, ResNet, U-Net, and the proposed FPDN for the LMMC of urban and dense urban environments. First of all, despite the wide PCBW (i.e., 10MHz) and longer correlation interval (i.e., 10ms), SAGE and LIMS show poor

performance in most C/N_0 range in comparison to the DNN-based techniques, and MLP (-based FPD technique) shows similar performance to the SAGE and LIMS. The poor performance of the SAGE, LIMS, and MLP is because of the same reason explained for the results in Fig. 6.

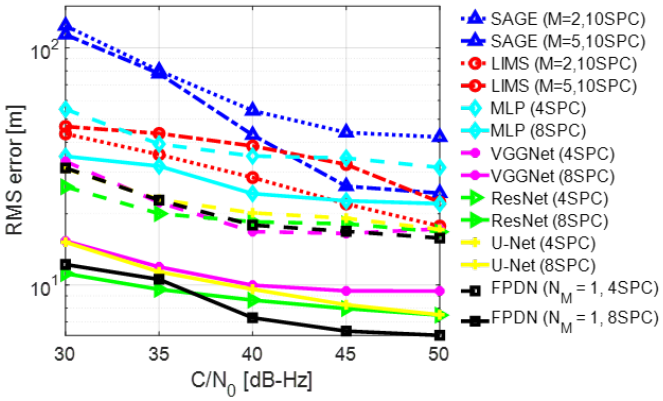
Similar to the results in Fig. 6, DNN (MLP, VGGNet, ResNet, U-Net, MLP-Mixer1)-based (FPD) techniques achieve superior performance for the entire C/N_0 range than SAGE, LIMS, and MLP of the same sampling rate. This is because DNN-based techniques are data-driven and their FPD is based on recognition of various detected features of the ACF output envelopes. Again, ResNet shows slightly better performance than VGGNet and UNet for a high sampling rate (e.g., 8SPC), which is because the ResNet utilizes the residual (or skip) connection to maintain the details (i.e., resolution) of the input image at deeper layers, whereas VGGNet and UNet loose the resolution of the input image at deeper layers. Notice again that the DNN-based techniques show similar performance for urban and dense urban environments and that the performance of the DNN-based techniques in Fig. 6 and that in Fig. 8 are also very similar. This is because the performance of a neural network is mainly determined by the amount of experience (i.e., training), and, therefore, when a network is sufficiently trained, the performance for the both channels has to be similar.

Comparing the proposed FPDN and other CNN-based techniques, the proposed FPDN shows a better accuracy for a high sampling rate. This is because the MLP-Mixer can recognize global features on the input image without loss of the image resolution, unlike other CNNs. The proposed FPDN is only slightly better for high C/N_0($\geq$ 40dB-Hz) and high sampling rate, however, as shown in Table I and Table III, the computational cost for the proposed FPDN is much lower than the ResNet and other CNNs, which can be critical for cheap GPS receivers. Therefore, it is demonstrated that the proposed FPDN achieves the most accurate FPD in urban and dense urban environments, while its memory usage and computational cost are smaller than other CNN-based techniques.

Additionally, we compare the performance of the proposed FPDN for sampling rate, MLP-Mixer blocks of various depths (i.e., N_M equal to 1, 2, 3, and 8), and the urban and dense urban LMMC models. The results are shown in Fig. 9, which is to testify the effect of the depth of the MLP-Mixer blocks on the FP delay detection performance. As shown, the detection performance of the FPDN does not noticeably change with respect to the number of Mixer blocks but the performance strongly depends on the sampling rate. This does not agree with the general observation that the detection performance of DNN is expected to improve as the number of blocks increases. This is because the MLP-Mixer is more capable of recognizing the global feature (such as the ACF envelope) on the ACF output image M_F at shallow layers than CNN-based techniques where a global feature is extracted after the input image is compressed through multiple layers. Therefore, the MLP-Mixer is a unique choice and suitable to the problem of FPD from an ACF output image in terms of FPD accuracy and computational cost.



(a)



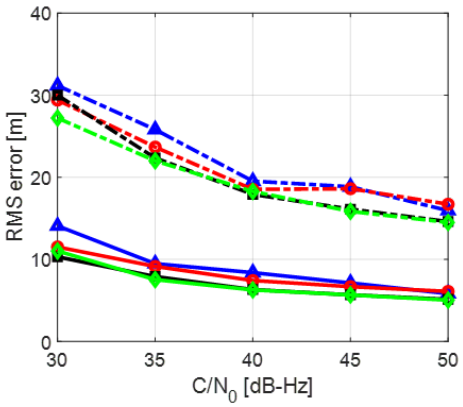
(b)

Fig. 8. Comparison of the proposed FPDN to other techniques for various C/N_0 and PCBW in (a) urban and (b) dense urban environments

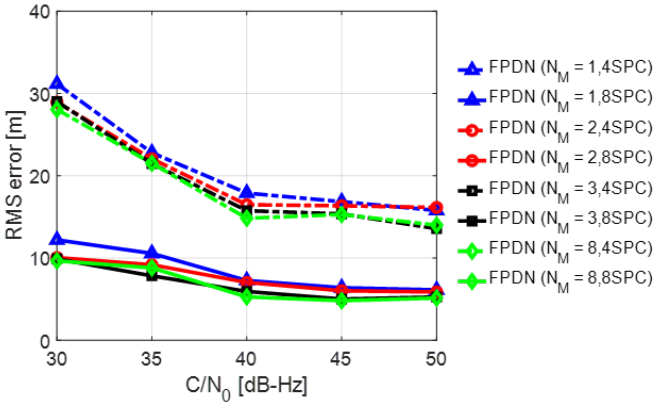
C. Positioning Performance Improvement in the Field

Finally, we test the performance of the proposed FPDN using real-world GPS measurements. We use a software-defined receiver [39] with a PCBW of 8.184MHz to receive GPS L1 frequency (1575.42MHz) signals. To precisely simulate the effect of multipath on positioning, we utilize observed LOS GPS measurements to generate various two-path multipath channels with complex amplitudes. The FP of the multipath channel is set to a power of -5dB and the 2nd path has a power of 0dB with $0.7T_c$ excess delay. The two paths are set to have a random relative phase with each other.

We compare the FPD performance of the proposed FPDN with the conventional EL discriminator as shown in Fig. 10. Fig. 10(a) depicts the satellite constellation used in the field test experiment; we only use four LOS GPS satellites, 5, 13, 15, and 29, for positioning to clearly show the effect of multipath and the improvement by the proposed technique. Fig. 10(b) shows the positioning results when the signal measurements of the GPS satellite-13 located at 35° azimuth and 60° elevation are used for the two-path multipath channel. Fig. 10(c) shows the positioning errors when the signals of the GPS satellite-15, located at azimuth 306° and elevation 72°, are used for the two-path multipath channel. Fig. 10(d) shows the results of the positioning when both the signals of GPS



(a)



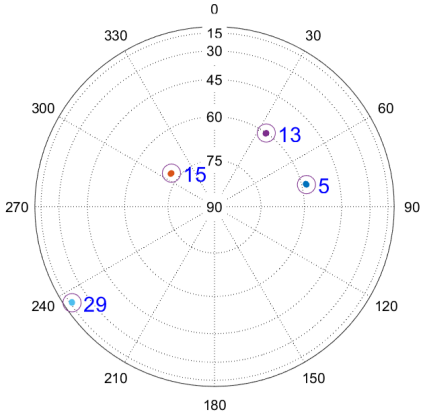
(b)

Fig. 9. Performance comparison of FPDN for different number of MLP-Mixer layers (a) urban and (b) dense urban environments

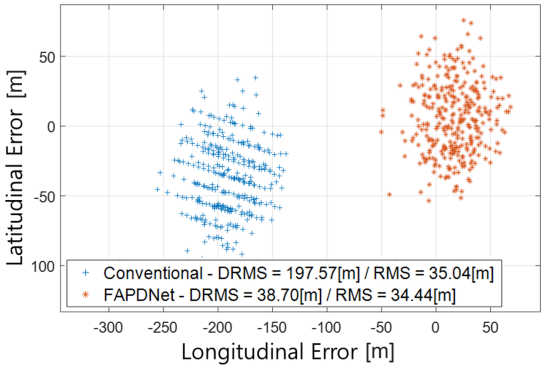
satellite-13 and satellite-15 are used for the two-path multipath channel. Fig. 10 (b), (c), and (d) show the distance root-mean square (DRMS) errors to indicate the accuracy and RMS errors to indicate the precision of the positioning technique. As shown, it is clear that the proposed FPDN can detect the FP reliably and can improve positioning accuray in the multipath environments, which is not possible for the conventional EL discriminator.

VI. Conclusion

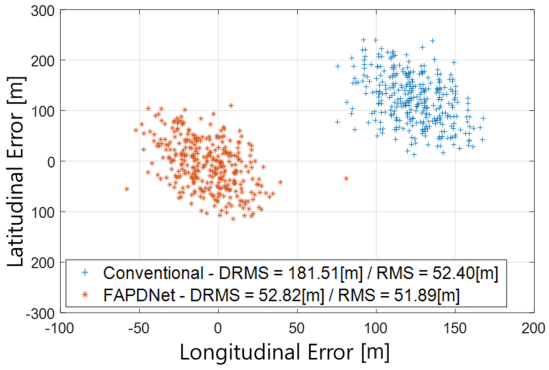
In this paper, we have investigated the first path detection (FPD) problem in multipath environments, and we have proposed the FPDN, which is based on MLP-Mixers to estimate the FP delay. We have presented a pre-processing scheme to build input image to the proposed FPDN from the discrete ACF output and the complete architecture of the proposed FPDN. The performance of the proposed FPDN has been compared to that of various path detection techniques, such as conventional EL discriminators, SAGE, LIMS and MLP, VGGNet, ResNet and U-Net, for various channels, such as 2-path, 3-path, 4-path, and the LMMC of urban and dense urban environments. It has been demonstrated that the proposed FPDN not only achieves strong FPD performance for all multipath channels, but also has important practical advantages



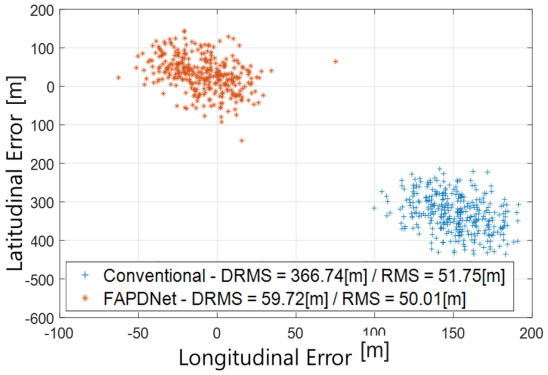
(a)



(b)



(c)



(d)

Fig. 10. Positioning Errors with GPS measurements (a) satellite constellation, (b) and (c) multipath channel for GPS satellite-13 and 15 respectively, (d) multipath channel for both GPS satellites

in comparison to other techniques. The proposed FPDN does not require any assumption on the number of multipaths in advance of the FPD, which is a strong advantage in practice over SAGE and LIMS. And the proposed FPDN requires lower memory usage and lower computational cost than other CNN-based techniques, which shows that the proposed FPDN is suitable for low-cost GPS receivers. Finally, we have tested the proposed FPDN using real GPS measurements and demonstrated a significant positioning accuracy improvement compared to the conventional EL discriminators in multipath channels. Overall, the proposed FPDN is suitable for real-time mobile GPS receivers in multipath environments for its superior accuracy and moderate computational cost.

APPENDIX

In this section, we describe MLP, VGGNet, ResNet, and U-Net used for performance comparison with the proposed FPDN.

Unlike the proposed FPDN, an MLP network receives an input in a column-wise vector format. In this paper, the I/Q-channel ACF outputs are separately fed into the MLP network as shown in Fig. 11. First, there are hidden layers of size 160, 320, 320, 320, 160, 160, 160, and 160 for I/Q-channels processed in parallel, then the two channels are concatenated, flatened, and processed together by the hidden layers of size 320, 160, and 50. Since it is designed to output the chip delay, the final output is a single value. Batch normalization and ReLU are placed equally in each layer, and only the tanh activation function is used in the last layer.

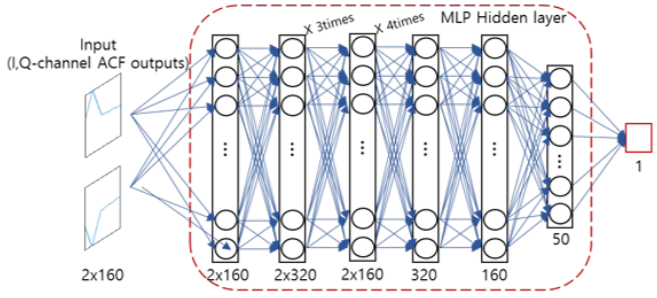


Fig. 11. MLP network used for comparison in subsection V-A

We utilize two VGGNets to process the I/Q-channel ACF outputs separately as shown in Fig. 12, which illustrates the modified VGGNet (which we call VGGNet in the paper) with the kernel size of 3×3. VGGNet is called VGG16 or VGG19, depending on the depth of the layer [23], and, in this paper, we utilize two VGG16 models in parallel to process the I/Q-channel ACF outputs (of size 160×160 each) and the two final feature-maps (of size $128 \times 5 \times 5$ each) of the two VGG16s are concatenated to build an overall feature-map of size $256 \times 5 \times 5$. After that, the merged feature map is passed through one additional convolution layer and then flattened for input to the fully connected layer. Although not specified in Fig. 12, we designed a model that consistently uses batch normalization and ReLU for all convolutional layers and outputs the final value through the tanh activation function in the fully-connected layer.

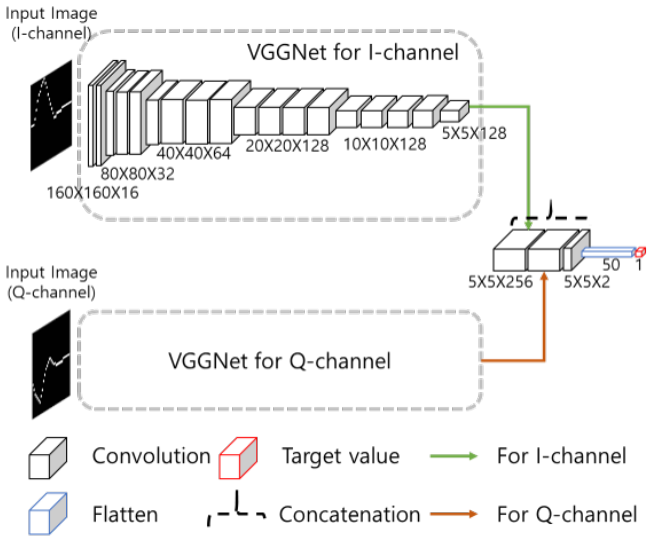


Fig. 12. Modified VGGNet used for comparison in subsection V-A, B

We utilize two ResNets to process the I/Q-channel ACF outputs separately as shown in Fig. 13, which shows the modified ResNet (that we call ResNet in the paper). Basically, ResNet utilizes the VGGNet as a skeleton model, but ResNet employs residual (skip) connections to lessen the gradient vanishing problem at the deep layers. The feature map size at each layer of the modified ResNet is set as shown in Fig. 13. Note that, to keep the number of parameters similar to other CNN-based techniques and the proposed FADNet compared in this paper, the feature map size of the 4th and 5th convolution layers is set smaller than that of VGGNet. The final feature-maps for I/Q-channels are merged when each feature-map size is $128 \times 5 \times 5$ as in VGGNet. Although not specified in Fig. 13, we add batch normalization and ReLU for all convolutional layers and outputs the final value through the tanh activation function in the fully-connected layer.

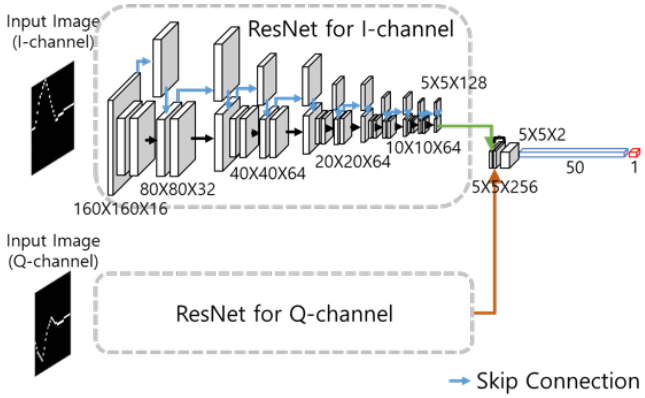


Fig. 13. Modified ResNet used for comparison in subsection V-A, B

The Modified U-Net (which we call U-Net in the paper) based on two U-Nets to process I/Q-channel ACF outputs separately as shown in Fig. 14, where a U-Net consists of a contraction path to extract the context of the input image and an expansiion path to combine the context found in

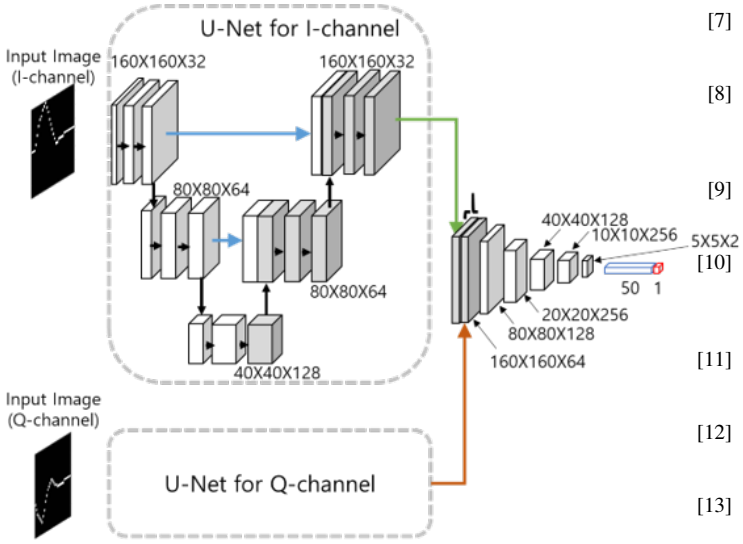


Fig. 14. Modified U-Net used for comparison in subsection V-A, B

the contraction path and upsampled feature map from the previous expansion path layer. In general, the encoder/decoder-based segmentation networks employs a similar architecture (i.e., contraction path and expansion path), which results in the loss of sophisticated pixel information. However, U-Net utilizes a skip connection to directly hand over the feature map from the Contracting path to the expansion path of the same depth, so that the features of the image can be elaborately segmented. The modified U-Net used in this paper has the parallel structure [25], so that the four steps in [25] is reduced in half to reduce computational costs. Again, we use the Batch Normalization and ReLU in the same way as other CNN models. In the modified U-Net, the final feature-maps for I/Q-channels are concatenated when the feature-map size becomes $64 \times 160 \times 160$, and, after passing through convolution layers, the chip delay estimate is generated through a fully connected layer as shown in Fig. 14.

REFERENCES

[1] A. Steingass and A. Lehner, "Measuring the navigation multipath channel-A statistical analysis," in *Proceedings of the 17th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2004)*, 2004, pp. 1157–1164.

[2] M.-D. Kim, J. Lee, J. Liang, and J. Kim, "Multipath channel characteristics for propagation between mobile terminals in urban street canyon environments," in *2015 17th International Conference on Advanced Communication Technology (ICACT)*. IEEE, 2015, pp. 511–516.

[3] S.-H. Kong, "TOA and AOD statistics for down link Gaussian scatterer distribution model," *IEEE transactions on wireless communications*, vol. 8, no. 5, pp. 2609–2617, 2009.

[4] A. Van Dierendonck, P. Fenton, and T. Ford, "Theory and performance of narrow correlator spacing in a GPS receiver," *Navigation*, vol. 39, no. 3, pp. 265–283, 1992.

[5] L. Garin, F. van Diggelen, and J.-M. Rousseau, "Strobe & edge correlator multipath mitigation for code," in *Proceedings of the 9th International Technical Meeting of the Satellite Division of the Institute of Navigation (ION GPS 1996)*, 1996, pp. 657–664.

[6] B. Townsend and P. Fenton, "A practical approach to the reduction of pseudorange multipath errors in a L1 GPS receiver," in *Proceedings of the 7th International Technical Meeting of the Satellite Division of the Institute of Navigation, Salt Lake City, UT, USA*. Citeseer, 1994, pp. 20–23.

[7] B. R. Townsend, P. C. Fenton, K. J. Van Dierendonck, and D. R. Van Nee, "Performance evaluation of the multipath estimating delay lock loop," *Navigation*, vol. 42, no. 3, pp. 502–514, 1995.

[8] N. Blanco-Delgado and F. D. Nunes, "Multipath estimation in multicorrelator GNSS receivers using the maximum likelihood principle," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 48, no. 4, pp. 3222–3233, 2012.

[9] M. Lentmaier, B. Krach, and P. Robertson, "Bayesian time delay estimation of GNSS signals in dynamic multipath environments," *International Journal of Navigation and Observation*, vol. 2008, March 2008.

[10] M. Sahmoudi and M. G. Amin, "Fast iterative maximum-likelihood algorithm (FIMLA) for multipath mitigation in the next generation of GNSS receivers," *IEEE Transactions on Wireless Communications*, vol. 7, no. 11, pp. 4362–4374, 2008.

[11] A. Doucet, N. De Freitas, K. Murphy, and S. Russell, "Rao-Blackwellised particle filtering for dynamic bayesian networks," *arXiv preprint arXiv:1301.3853*, 2013.

[12] J. A. Fessler and A. O. Hero, "Space-alternating generalized expectation-maximization algorithm," *IEEE Transactions on signal processing*, vol. 42, no. 10, pp. 2664–2677, 1994.

[13] D. Shutin and B. H. Fleury, "Sparse variational bayesian SAGE algorithm with application to the estimation of multipath wireless channels," *IEEE Transactions on signal processing*, vol. 59, no. 8, pp. 3609–3623, 2011.

[14] R. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE transactions on antennas and propagation*, vol. 34, no. 3, pp. 276–280, 1986.

[15] R. Roy and T. Kailath, "ESPRIT-estimation of signal parameters via rotational invariance techniques," *IEEE Transactions on acoustics, speech, and signal processing*, vol. 37, no. 7, pp. 984–995, 1989.

[16] A. Bruckstein, T.-J. Shan, and T. Kailath, "The resolution of overlapping echos," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 33, no. 6, pp. 1357–1367, 1985.

[17] J. Kusuma, I. Maravic, and M. Vetterli, "Sampling with finite rate of innovation: Channel and timing estimation for UWB and GPS," in *IEEE International Conference on Communications, 2003. ICC'03.*, vol. 5. IEEE, 2003, pp. 3540–3544.

[18] F. Bouchereau, D. Brady, and C. Lanzl, "Multipath delay estimation using a superresolution PN-correlation method," *IEEE transactions on signal processing*, vol. 49, no. 5, pp. 938–949, 2001.

[19] W. Nam and S.-H. Kong, "Least-squares-based iterative multipath super-resolution technique," *IEEE Transactions on signal processing*, vol. 61, no. 3, pp. 519–529, 2012.

[20] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel, "Backpropagation applied to handwritten zip code recognition," *Neural computation*, vol. 1, no. 4, pp. 541–551, 1989.

[21] M.-C. Popescu, V. E. Balas, L. Perescu-Popescu, and N. Mastorakis, "Multilayer perceptron and neural networks," *WSEAS Transactions on Circuits and Systems*, vol. 8, no. 7, pp. 579–588, 2009.

[22] I. Tolstikhin, N. Houlsby, A. Kolesnikov, L. Beyer, X. Zhai, T. Unterthiner, J. Yung, D. Keysers, J. Uszkoreit, M. Lucic *et al.*, "MLP-Mixer: An all-MLP architecture for vision," *arXiv preprint arXiv:2105.01601*, 2021.

[23] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556*, 2014.

[24] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.

[25] O. Ronneberger, P. Fischer, and T. Brox, "U-net: Convolutional networks for biomedical image segmentation," in *International Conference on Medical image computing and computer-assisted intervention*. Springer, 2015, pp. 234–241.

[26] S.-H. Kong, "Statistical analysis of urban GPS multipaths and pseudo-range measurement errors," *IEEE transactions on aerospace and electronic systems*, vol. 47, no. 2, pp. 1101–1113, 2011.

[27] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Advances in neural information processing systems*, vol. 25, pp. 1097–1105, 2012.

[28] A. Madani, J. R. Ong, A. Tibrewal, and M. R. Mofrad, "Deep echocardiography: data-efficient supervised and semi-supervised deep learning towards automated diagnosis of cardiac disease," *NPJ digital medicine*, vol. 1, no. 1, pp. 1–11, 2018.

[29] V. Moskalenko, N. Zolotykh, and G. Osipov, "Deep learning for ECG segmentation," in *international conference on Neuroinformatics*. Springer, 2019, pp. 246–254.

[30] S. L. Oh, E. Y. Ng, R. San Tan, and U. R. Acharya, "Automated beat-wise arrhythmia diagnosis using modified u-net on extended electrocardiographic recordings with heterogeneous arrhythmia types," *Computers in biology and medicine*, vol. 105, pp. 92–101, 2019.

[31] T. He, Y. Liu, C. Xu, X. Zhou, Z. Hu, and J. Fan, "A fully convolutional neural network for wood defect location and identification," *IEEE Access*, vol. 7, pp. 123 453–123 462, 2019.

[32] A. Dosovitskiy, L. Beyer, A. Kolesnikov, D. Weissenborn, X. Zhai, T. Unterthiner, M. Dehghani, M. Minderer, G. Heigold, S. Gelly *et al.*, "An image is worth 16x16 words: Transformers for image recognition at scale," *arXiv preprint arXiv:2010.11929*, 2020.

[33] M. Lin, Q. Chen, and S. Yan, "Network in network," *arXiv preprint arXiv:1312.4400*, 2013.

[34] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[35] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proceedings of the fourteenth international conference on artificial intelligence and statistics*. JMLR Workshop and Conference Proceedings, 2011, pp. 315–323.

[36] J. J. Spilker Jr, P. Axelrad, B. W. Parkinson, and P. Enge, *Global Positioning System: theory and applications, volume I*. American Institute of Aeronautics and Astronautics, 1996.

[37] K. Yu, I. Sharp, and Y. J. Guo, *Ground-based wireless positioning*. John Wiley & Sons, 2009, vol. 5.

[38] I. Recommendation, "Propagation data required for the design of earth-space land mobile telecommunication systems," *International Telecommunication Union*, pp. 681–692, 2019.

[39] Sparkfun Electronics (2012), "Sige GN3S sampler v3." [Online]. Available: http://www.sparkfun.com/products/10981

Seung-Hyun Kong (M'06–SM'16) is an Associate Professor in the CCS Graduate School of Green Transportation of Korea Advanced Institute of Science and Technology (KAIST), where he has been a faculty member since 2010. He received a B.S. degree in Electronics Engineering from Sogang University, Seoul, Korea, in 1992, an M.S. degree in Electrical and Computer Engineering from Polytechnic University (merged to NYU), New York, in 1994, a Ph.D. degree in Aeronautics and Astronautics from Stanford University, Palo Alto, in 2005. From 1997 to 2004 and from 2006 to 2010, he was with companies including Samsung Electronics (Telecommunication Research Center), Korea, and Qualcomm (Corporate R&D Department), San Diego, USA for advanced technology R&D in mobile communication systems, wireless positioning, and assisted GNSS. Since he joined KAIST as a faculty member in 2010, he has been working on various R&D projects in advanced intelligent transportation systems, such as robust GNSS-based navigation for urban environment, deep learning and reinforcement learning algorithms for autonomous vehicles, sensor fusion, and vehicular communication systems (V2X). He has authored more than 100 papers in peer-reviewed journals and conference proceedings and 12 patents, and his research group won the President Award (of Korea) in the 2018 international student autonomous driving competition host by the Korean government. He has served as an associate editor of IEEE T-ITS and IEEE Access, an editor of IET-RSN and the lead guest editor of the IEEE TITS special issue on "ITS empowered by AI technologie" and the IEEE Access special section on "GNSS, Localization, and Navigation Technologies". He has served as the program chair of IPNT from 2017 to 2019 in Korea and as a program co-chair of IEEE ITSC2019, New Zealand

Sangjae Cho received the B.S. degree in Department of Energy and Electrical Engineering from Korea Polytechnic University, Korea, in 2018 and M.S. degree in CCS Graduate School of Green Transportation from KAIST where he is currently pursuing the Ph.D. degree. His research interests include GNSS, Signal processing, Deep learning, Autonomous vehicle, and Wireless communication.

Euiho Kim received the B.S. degree from the Department of Aerospace Engineering, Iowa State University, Ames, IA, USA, and the Ph.D. and M.S. degree from the Department of Aeronautics and Astronautics, Stanford University, Stanford, CA, USA. He is currently an Assistant Professor in the Department of Mechanical & System Design Engineering, Hongik University, Korea. Prior to this, he was a Research Associate in the Department of Aerospace Engineering, University of Kansas; and the Technical Lead of the ground-based augmentation system of GPS and FAA's alternative position, navigation, and timing programs. His current research interests include satellite-based navigation, aircraft navigation using ground nav-aids, indoor navigation, and robotics.