

Article

Deep Learning Soft-Decision GNSS Multipath Detection and Mitigation

Fernando Nunes ^{1,2,*},[†]  and Fernando Sousa ^{1,3},[†] ¹ Instituto de Telecomunicações, 1049-001 Lisboa, Portugal; fsousa@isel.pt² Instituto Superior Técnico, Universidade de Lisboa, Torre Norte, Piso 10, 1049-001 Lisboa, Portugal³ Instituto Superior de Engenharia de Lisboa, Instituto Politécnico de Lisboa, 1959-007 Lisboa, Portugal

* Correspondence: nunes@lx.it.pt

† These authors contributed equally to this work.

Abstract: A technique is proposed to detect the presence of the multipath effect in Global Navigation Satellite Signal (GNSS) signals using a convolutional neural network (CNN) as the building block. The network is trained and validated, for a wide range of C/N_0 values, with a realistic dataset constituted by the synthetic noisy outputs of a 2D grid of correlators associated with different Doppler frequencies and code delays (time-domain dataset). Multipath-disturbed signals are generated in agreement with the various scenarios encompassed by the adopted multipath model. It was found that pre-processing the outputs of the correlators grid with the two-dimensional Discrete Fourier Transform (frequency-domain dataset) enables the CNN to improve the accuracy relative to the time-domain dataset. Depending on the kind of CNN outputs, two strategies can then be devised to solve the equation of navigation: either remove the disturbed signal from the equation (hard decision) or process the pseudoranges with a weighted least-squares algorithm, where the entries of the weighting matrix are computed using the analog outputs of the neural network (soft decision).

Keywords: multipath detection; multipath mitigation; deep learning; convolutional neural network; multilayer perceptron



Citation: Nunes, F.; Sousa, F. Deep Learning Soft-Decision GNSS Multipath Detection and Mitigation. *Sensors* **2024**, *24*, 4663. <https://doi.org/10.3390/s24144663>

Academic Editors: Alex Minetto and Beatrice Motella

Received: 29 May 2024

Revised: 13 July 2024

Accepted: 15 July 2024

Published: 18 July 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Multipath is one of the major sources of positioning errors in GNSS receivers operating near the Earth surface. It is due to the reception of one or more reflected rays besides (or instead of) the direct ray or line of sight (LOS). The reflected rays are characterized by extra delays relative to the arrival of the LOS signal. The reflected signals may also exhibit different Doppler frequencies, for instance when the receiver is not static. Multipath aspects affect both the code and carrier measurements, although the magnitudes of the errors differ significantly [1].

Many different methods have been proposed to mitigate the effect of multipath on the computation of the position, velocity, and time of the GNSS receiver at different stages, from the antenna to post-detection Receiver Autonomous Integrity Monitoring (RAIM) techniques [2,3]. An important class of mitigation techniques operate at the correlators level, including non-parametric and parametric techniques. Non-parametric processing, such as double-delta processing [4], resorts to code discriminator designs that are less sensitive to multipath-induced errors, while parametric processing, such as the Multipath Estimating Delay Lock Loop (MEDLL) [5], tries to estimate the parameters associated with the reflected rays. In any case, it will be beneficial if reliable side information, concerning the degradation of a given signal by multipath aspects, is available. In fact, the existence of a binary multipath/no multipath classifier permits the receiver to adopt one of the following alternatives. Strategy I: remove the signal affected by multipath from the navigation equation, provided that the number of visible satellites exceeds the minimum of 4. Strategy II: keep the multipath-disturbed signal in the navigation equation but process it in a

different way from the other (unaffected) signals, using, for instance, a weighted least-squares estimator.

In the last two decades, several authors have applied machine learning (algorithms that can learn from experience) with various degrees of success to detect and mitigate multipath in GNSS receivers using supervised or unsupervised learning and different datasets [6–13]. References [6,10] use multilayer perceptrons, [8] utilizes a K-means clustering technique with unsupervised learning, and the remaining works resort to convolutional neural networks (CNNs). In [14], the authors replace standard correlation schemes with deep neural network-based correlation schemes to learn the complexity of the multipath channels. A technique based on the rain forest learning algorithm is proposed in [15] to estimate the multipath parameters and remove the estimated reflected signal components. A thorough survey on the application of machine learning techniques in different aspects of GNSS signal processing, including multipath mitigation, is provided in [16]. Reference [17] reviews the previous work on multipath mitigation using machine learning techniques, with the received signal strength, elevation angle, and receiver correlator outputs constituting the most popular input features.

Herein, we propose a multipath/no multipath classifier based on a CNN with supervised learning that uses synthetic signals disturbed by additive correlated noise during the training stage. The dataset is generated by a 2D grid of correlators with different code delays and Doppler frequencies. The existence of multipath provokes the change in the correlator outputs, and these features can be captured by the neural network. Since the dataset is 2D, the problem is similar to the feature extraction in images, where the CNNs have been immensely successful [18]. The adopted CNN is trained and validated with noisy synthetic signals for a wide range of carrier-to-noise ratios, namely $30 \leq C/N_0 \leq 50$ dB-Hz. The dataset is generated using a blend of different environments: open, rural, suburban, urban, and highway. These environments constitute the DLR (German Aerospace Center) satellite-ground model proposed in [19,20]. Alternatively, the multipath environment could have been characterized by the more complicated model proposed in [21], which is adopted, for instance, in [14].

Extensive simulations have shown that the best results are obtained when the CNN inputs are pre-processed with a two-dimensional Discrete Fourier Transform. The proposed algorithms are well-suited for software receivers, which employ an analog-to-digital converter that captures all the channels and demodulates the channel waveforms using software on a general-purpose processor [22]. Depending on the type of neural network outputs (hard or soft decisions), either strategy I or II may be applied to solve the equation of navigation.

The paper is organized as follows. In Section 2, we characterize the outputs of the grid of correlators in the presence of multipath and Gaussian additive noise. Section 3 describes the DLR multipath model used to train the networks and addresses the training of two competing neural networks: the multilayer perceptron (MLP) and the convolutional neural network (CNN), with the MLP serving as the benchmark. In Section 4, the performance of the selected solution (CNN), using hard and soft decisions, is evaluated and a multipath mitigation technique is proposed. Finally, conclusions are drawn in Section 5.

2. Correlators Characterization

Assume that the receiver includes a front-end that heterodynes the GNSS signal, $r(t)$, transmitted by each satellite, to produce the complex baseband signal $y(t)$ and a grid of $N_f N_c$ correlators, as sketched in Figure 1. In the scheme, $2B$ is the front-end bandwidth. The frequency $f_c + \hat{f}_{d_0}$ is derived from locking the local oscillator to the incoming signal, with f_c denoting the nominal GNSS carrier frequency. The correlators are separated in frequency by $\Delta f = 2f_{dop}/N_f$ and in the code delay by $\Delta\tau = 2/N_c$ (in chip units). The maximum Doppler frequency range, $2f_{dop}$, should be adjusted for every scenario to encompass the different carrier frequencies of the multipath replicas that hit the receiver's antenna.

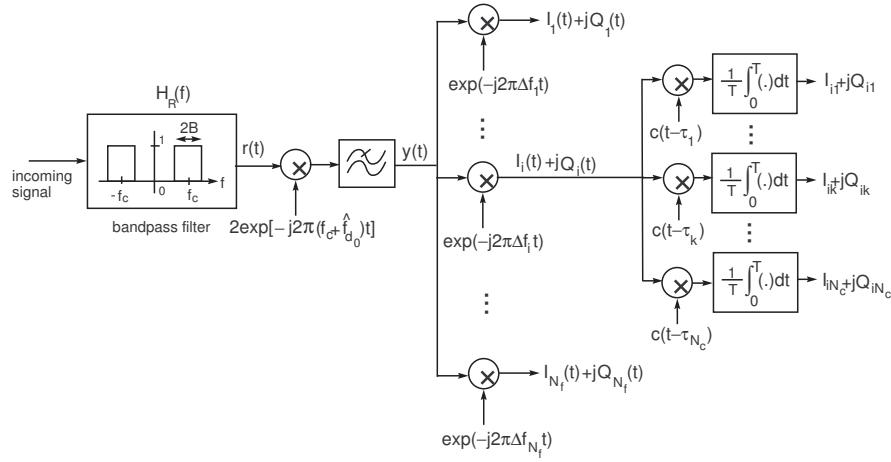


Figure 1. Grid of correlators.

Using a software receiver, the analog signal $y(t)$ is sampled and the rest of the operations leading to the correlators output will be performed with digital signals (although this is not shown in Figure 1 for simplicity). The correlators outputs are provided by

$$Z_{ik} = I_{ik} + jQ_{ik} = \frac{1}{T} \int_0^T y(t) \exp(-j2\pi\Delta f_i t) c(t - \tau_k) dt, \quad (1)$$

where T is the duration of the correlation interval and $c(t)$ is a pseudorandom (PRN) code sequence with chip duration T_c and $|c(t)| = 1$. Conducting $\Delta f_i = f_{min} + i\Delta f$, $i = 0, \dots, N_f - 1$, we obtain

$$Z_{ik} = \frac{1}{T} \int_0^T x_k(t) \exp(-j2\pi i \Delta f t) dt, \quad (2)$$

with $x_k(t) = y(t) \exp(-j2\pi f_{min} t) c(t - \tau_k)$. In software receivers, the quantities in (2) are calculated from the samples $x_k(t_m)$ as

$$Z_{ik} \approx \frac{1}{M} \sum_{m=0}^{M-1} x_k(t_m) \exp\left(-j\frac{2\pi im\mu}{M}\right), \quad (3)$$

in which $t_m = m\Delta t$, where $\Delta t = T/M$ is the sampling interval and $\mu = \Delta f T$.

Consider that, in the presence of multipath, the received GNSS signal $r(t)$ is constituted by a strong ray and N_r weaker rays plus additive Gaussian noise $w(t)$. The noise in-band power spectrum is $G_w(f) = N_0/2$ for $f_c - B \leq |f| \leq f_c + B$. The bandpass signal is

$$r(t) = \sum_{n=0}^{N_r} A_n D(t - \tau_n) c(t - \tau_n) \cos[2\pi(f_c + f_{d_n})t + \theta_n] + w(t). \quad (4)$$

Each ray of index n is characterized by amplitude A_n , propagation delay τ_n , Doppler frequency f_{d_n} , and phase θ_n . $D(t)$ is the navigation message, with $|D(t)| = 1$. In the case of a pilot channel, the navigation message is constant with $D(t) = 1$ (this condition will be assumed hereafter). The index $n = 0$ corresponds to the strongest ray, which coincides, in general, with the line-of-sight (LOS) ray when it is not obstructed. The bandwidth $2B$ is considered sufficient to accommodate the incoming signals in the presence of Doppler frequency shifts. Signal $r(t)$ is converted to baseband using the local oscillator complex signal $2 \exp[-j2\pi(f_c + \hat{f}_{d_0})t]$. For the sake of simplicity, it is assumed that the carrier

tracking loop is perfectly synchronized with the strongest incoming signal; that is, $\hat{f}_{d_0} = f_{d_0}$. The result is

$$y(t) = \sum_{n=0}^{N_r} A_n c(t - \tau_n) \exp[j(2\pi(f_{d_n} - f_{d_0})t + \theta_n)] + N(t), \quad (5)$$

where $N(t) = NI(t) + jNQ(t)$, with $NI(t)$ and $NQ(t)$ being independent, zero mean, inphase/quadrature components of $w(t)$ with equal powers $2N_0B$. The baseband signal $y(t)$ is then multiplied by a grid of $N_f N_c$ complex units, $\exp(-j2\pi\Delta f_i t) \cdot c(t - \tau_k)$, $i = 1, \dots, N_f$, $k = 1, \dots, N_c$, to yield a 2D correlation, as sketched in Figure 1.

For the signal model described in (4), the correlators outputs are

$$\begin{aligned} Z_{ik} &= \frac{1}{T} \int_0^T [I_i(t) + jQ_i(t)] c(t - \tau_k) dt \\ &= \sum_{n=0}^{N_r} A_n \exp(j\theta_n) \frac{1}{T} \int_0^T \exp[j2\pi(f_{d_n} - f_{d_0} - \Delta f_i)t] c(t - \tau_n) c(t - \tau_k) dt, \\ &\quad + N_{ik}, \end{aligned} \quad (6)$$

with the noise component being

$$N_{ik} = NI_{ik} + jNQ_{ik} = \frac{1}{T} \int_0^T N(t) \exp(-j2\pi\Delta f_i t) c(t - \tau_k) dt. \quad (7)$$

The integrals in (6) are of the form

$$I_c = \frac{1}{T} \int_0^T c(t - \tau_n) c(t - \tau_k) \exp(j2\pi\varphi t) dt. \quad (8)$$

Assuming that the signal $\exp(j2\pi\varphi t)$ is slowly varying in each interval of duration $T_p = T/P$, with $P \gg 1$, we may write

$$I_c \approx \frac{1}{P} \sum_{p=1}^P \exp[j2\pi\varphi(p - 1/2)T_p] \frac{1}{T_p} \int_{(p-1)T_p}^{pT_p} c(t - \tau_n) c(t - \tau_k) dt. \quad (9)$$

Let $R_c(\tau)$ denote the autocorrelation function of code $c(t)$ with period ST_c

$$R_c(\tau) = \frac{1}{ST_c} \int_0^{ST_c} c(t) c(t - \tau) dt \approx \frac{1}{T_p} \int_{(p-1)T_p}^{pT_p} c(t) c(t - \tau) dt, \quad (10)$$

and $T_p < ST_c$, leading to

$$\begin{aligned} I_c &\approx \frac{1}{P} R_c(\tau_k - \tau_n) \sum_{p=1}^P \exp[j2\pi\varphi(p - 1/2)T_p] \\ &= \frac{1}{P} R_c(\tau_k - \tau_n) \exp(j\pi\varphi T) \frac{\sin(\pi\varphi T)}{\sin(\pi\varphi T_p)}. \end{aligned} \quad (11)$$

For $|\varphi T_p| \leq 0.2$, the Dirichlet kernel $\sin(\pi\varphi PT_p)/[P \sin(\pi\varphi T_p)]$, is well-approximated by the sinc function, defined as $\text{sinc}(x) \equiv \sin(\pi x)/(\pi x)$, yielding $I_c \approx R_c(\tau_k - \tau_n) \text{sinc}(\varphi T) \exp(j\pi\varphi T)$. Thus, (6) can be written approximately as

$$\begin{aligned} Z_{ik} &\approx \sum_{n=0}^{N_r} A_n R_c(\tau_k - \tau_n) \text{sinc}[(f_{d_n} - f_{d_0} - \Delta f_i)T] \exp\{j[\pi(f_{d_n} - f_{d_0} - \Delta f_i)T + \theta_n]\} \\ &\quad + N_{ik}. \end{aligned} \quad (12)$$

The cross-correlation of the noise components of two correlators, N_{ik} and N_{lm} , is defined as

$$\begin{aligned} E\{N_{ik}N_{lm}^*\} &= \\ \frac{1}{T^2} \int_0^T \int_0^T E\{N(t)N^*(\lambda)\} \exp[-j2\pi(\Delta f_i t - \Delta f_l \lambda)] c(t - \tau_k) c(\lambda - \tau_m) dt d\lambda. \end{aligned} \quad (13)$$

The determination of the cross-correlation is complicated except when $BT \gg 1$. In this case, $E\{N(t)N^*(\lambda)\} \approx 2N_0\delta(t - \lambda)$ and

$$\begin{aligned} E\{N_{ik}N_{lm}^*\} &\approx \\ \frac{2N_0}{PT} \sum_{p=1}^P \exp[-j2\pi(\Delta f_i - \Delta f_l)(p - 1/2)T_p] \frac{1}{T_p} \int_{(p-1)T_p}^{pT_p} c(\lambda - \tau_k) c(\lambda - \tau_m) d\lambda. \end{aligned} \quad (14)$$

Therefore,

$$E\{N_{ik}N_{lm}^*\} \approx \frac{2N_0}{PT} R_c(\tau_m - \tau_k) \exp[j\pi(\Delta f_l - \Delta f_i)T] \frac{\sin[\pi(\Delta f_l - \Delta f_i)T]}{\sin[\pi(\Delta f_l - \Delta f_i)T_p]} \quad (15)$$

and, assuming that $|(\Delta f_l - \Delta f_i)T_p| \ll 1$, we have

$$E\{N_{ik}N_{lm}^*\} \approx \frac{2N_0}{T} R_c(\tau_m - \tau_k) \exp[j\pi(\Delta f_l - \Delta f_i)T] \text{sinc}[(\Delta f_l - \Delta f_i)T]. \quad (16)$$

Taking into account (7) and (16), we obtain the following expressions for the real-valued cross-correlations

$$\begin{aligned} \begin{bmatrix} E\{NI_{ik}NI_{lm}\} \\ E\{NQ_{ik}NQ_{lm}\} \end{bmatrix} &= \begin{bmatrix} E\{NQ_{ik}NQ_{lm}\} \\ -E\{NI_{ik}NQ_{lm}\} \end{bmatrix} \\ &\approx \frac{N_0}{T} R_c(\tau_m - \tau_k) \text{sinc}[(\Delta f_l - \Delta f_i)T] \begin{bmatrix} \cos[\pi(\Delta f_l - \Delta f_i)T] \\ \sin[\pi(\Delta f_l - \Delta f_i)T] \end{bmatrix}. \end{aligned} \quad (17)$$

Using Monte Carlo simulation, the noise random variables (r.v.) NI_{ik} and NQ_{ik} have to be generated for all the elements of the grid $i = 1, \dots, N_f$, $k = 1, \dots, N_c$. To this end, we concatenate the noise matrices $NI(N_f \times N_c)$ and $NQ(N_f \times N_c)$ into a single vector $U(p)$, $p = 1, \dots, 2N_f N_c$, according to $U((i-1)N_c + k) = NI_{ik}$ and $U((i-1)N_c + k + N_f N_c) = NQ_{ik}$.

The covariance matrix $\mathcal{C} = E\{UU^T\}$ of the resulting vector $U(2N_f N_c \times 1)$ has $4N_f^2 N_c^2$ elements. For $p, q = 1, \dots, 2N_f N_c$ and $p \leq q$, each element of \mathcal{C} is provided by

$$\mathcal{C}_{pq} = E\{U(p)U(q)\} = \begin{cases} E\{NI_{ik}NI_{lm}\}, & q \leq N_f N_c \text{ (case I)} \\ E\{NI_{ik}NQ_{lm}\}, & p \leq N_f N_c, q > N_f N_c \text{ (case II)}, \\ E\{NQ_{ik}NQ_{lm}\}, & p > N_f N_c \text{ (case III)} \end{cases} \quad (18)$$

with $\mathcal{C}_{qp} = \mathcal{C}_{pq}$. For the three cases, the indices of the elements of \mathcal{C} are computed as

Case I: $p = (i-1)N_c + k$, $q = (l-1)N_c + m$.

Case II: $p = (i-1)N_c + k$, $q = (l-1)N_c + m + N_f N_c$.

Case III: $p = (i-1)N_c + k + N_f N_c$, $q = (l-1)N_c + m + N_f N_c$.

After the covariance matrix is determined, U can be easily generated from vector $W(2N_f N_c \times 1)$ of Gaussian, zero-mean, unity power, and independent components using the technique described in Appendix B.

3. Neural Network Training

3.1. Multipath Model

In order to train the neural network (NN), we assume a certain ensemble of models to characterize multipath. This does not imply that the experimental multipath to be detected

obeys those models. In fact, in practice, it is almost impossible to assign a set of models that encompasses all cases encountered of real multipath. Nevertheless, the NN should be trained with the largest possible set of plausible multipath models.

Land mobile satellite (LMS) channels are usually divided into narrowband and wideband models. While the narrowband models describe the channel by a multiplicative operation on the signal, the wideband models take into account the frequency dependency caused by the signal echoes [23]. Echoes with different delays can be resolved when the difference in delay is larger than the inverse of the corresponding receiver (baseband) bandwidth: $\tau_m - \tau_k > 1/B$. For instance, the receiver reference bandwidth of Galileo signal E1 is $2B = 24.552$ MHz [24], which corresponds to a minimum resolvable difference of delays of approximately 81.5 ns (approximately 24.5 m).

We consider, henceforth, for the purpose of NN training, a wideband model of LMS communications affected by multipath fading and signal shadowing. The N_r different reflectors cause echoes with delays $\tau_m(t) = \tau_0(t) + \Delta\tau_m(t)$, where $\tau_0(t)$ is the propagation delay of the direct ray and $\Delta\tau_m(t)$ is the excess delay of each reflected ray. A simple but efficient model for wideband LMS channels is the tapped delay line model, where each tap is described by a narrowband model. Assume that the transmitted signal is represented as $s(t) = \text{Re}\{s_{bb}(t)e^{j2\pi f_c t}\}$, where $s_{bb}(t)$ is the corresponding complex envelope. In the case of multiple propagation paths, the equivalent lowpass channel is described by the time-variant impulse response [25]

$$h(\tau; t) = \sum_{m=0}^{N_r} E_m(t) \delta(\tau - \tau_m(t)), \quad E_m(t) = \alpha_m(t) e^{-j2\pi f_c \tau_m(t)}. \quad (19)$$

Let the satellite-receiver distance traveled by the m th path, in a short time interval, change approximately linearly with time (constant Doppler frequency). The channel impulse response may be written as

$$h(\tau; t) = \sum_{m=0}^{N_r} \alpha_m(t) \exp[j(2\pi f_m t - \phi_m)] \delta(\tau - \tau_m(t)), \quad (20)$$

which means that each echo is characterized by the following quantities: α_m (amplitude), f_m (Doppler frequency), ϕ_m (phase), and τ_m (delay).

Admit that the receiver's phase lock loop is synchronized to the carrier frequency of the direct ray (LOS), which includes the component due to the Doppler effect. Next, we characterize the Doppler frequency deviation $\Delta f_m = f_m - f_0$ for each reflected ray relative to the LOS. We follow the simplified scenario where the propagation between the satellite and the user (receiver) is constituted by an LOS and a reflected ray, as described in Figure 2. The reflector is considered static.

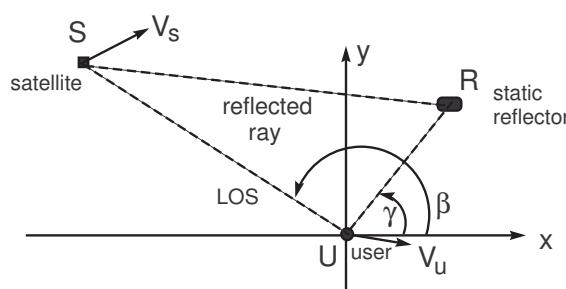


Figure 2. Multipath scenario.

If the coordinates of the satellite, reflector, and receiver are, respectively, $(x_s(t), y_s(t))$, (x_r, y_r) , and $(x_u(t), y_u(t))$, the lengths of the direct ray (LOS) and the reflected ray will be provided by

$$L_0(t) = \sqrt{[x_u(t) - x_s(t)]^2 + [y_u(t) - y_s(t)]^2} \quad (21)$$

and

$$L_1(t) + L_2(t) = \sqrt{[x_r - x_s(t)]^2 + [y_r - y_s(t)]^2} + \sqrt{[x_u(t) - x_r]^2 + [y_u(t) - y_r]^2}. \quad (22)$$

The satellite–user velocity using the LOS is

$$V_0(t) = \frac{dL_0(t)}{dt} = \frac{1}{L_0(t)} \left([x_u(t) - x_s(t)](V_{ux} - V_{sx}) + [y_u(t) - y_s(t)](V_{uy} - V_{sy}) \right), \quad (23)$$

where the satellite and user velocity vectors are, respectively, $V_s = (V_{sx}, V_{sy})$ and $V_u = (V_{ux}, V_{uy})$. The apparent satellite–user velocity using the reflected ray is

$$\begin{aligned} V_{12}(t) &= \frac{dL_1(t)}{dt} + \frac{dL_2(t)}{dt} \\ &= -\frac{1}{L_1(t)} \left([x_r - x_s(t)]V_{sx} + [y_r - y_s(t)]V_{sy} \right) + \\ &\quad \frac{1}{L_2(t)} \left([x_u(t) - x_r]V_{ux} + [y_u(t) - y_r]V_{uy} \right). \end{aligned} \quad (24)$$

Since, in general, $L_1(t) \approx L_0(t)$ and $L_2(t) \ll L_0(t)$, we have $V_{12}(t) - V_0(t) \approx V_{ux}(\cos \beta - \cos \gamma) + V_{uy}(\sin \beta - \sin \gamma)$. Consider now, without loss of generality, that the receiver travels along the X-axis. Then, $V_{uy} = 0$, and we obtain for the Doppler frequency deviation

$$\Delta f_d = -\frac{V_{12} - V_0}{c} f_c = \frac{V_{ux}(\cos \gamma - \cos \beta)}{c} f_c. \quad (25)$$

The maximum absolute value of the Doppler frequency deviations is equal to $2|V_{ux}|f_c/c$, corresponding for instance to $\beta = \pi$ and $\gamma = 0$. Namely, if the receiver is traveling at the speed of 140 km/h (typical car speed on a highway) and $f_c = 1.57$ GHz, the maximum absolute value will be ≈ 400 Hz (Doppler frequency range equal to ± 400 Hz).

The DLR model for the LMS channel, herein adopted, divides the channel impulse response with $N_r + 1$ rays into three parts [19,20]:

1. Direct path. There are two states for the direct ray: shadowing [bad channel state] and LOS (no shadowing) [good channel state]. The probability of each state depends on the type of environment: open, rural, suburban, urban, and highway (see Table A1 in Appendix A). For LOS conditions, a Rice distribution describes the probability density function (pdf) of the signal amplitude

$$p_{\alpha_0}(x) = \frac{x}{\sigma^2} I_0\left(\frac{x}{\sigma^2}\right) \exp\left(-\frac{x^2 + 1}{2\sigma^2}\right), \quad x \geq 0, \quad (26)$$

where $I_0(\cdot)$ is the zeroth-order modified Bessel function of the first kind and the Rice factor $\rho = 1/(2\sigma^2)$ denotes the carrier-to-multipath ratio. The corresponding cumulative distribution function (CDF) is [25]

$$F_{\alpha_0}(x) = 1 - Q\left(\frac{1}{\sigma}, \frac{x}{\sigma}\right), \quad x \geq 0, \quad (27)$$

where $Q(\cdot, \cdot)$ is Marcum's Q function.

In shadowed environments (bad channel state), α_0 is Rayleigh-distributed with lognormal-distributed mean power $P_0 = 2\sigma^2$. The result is the Suzuki distribution [26]. That is,

$$p_{\alpha_0}(x) = \frac{x}{\sigma^2} \exp\left(-\frac{x^2}{2\sigma^2}\right), \quad x \geq 0, \quad (28)$$

$$p_{P_0}(x) = \frac{10}{\sqrt{2\pi}\sigma \ln 10} \frac{1}{x} \exp\left[-\frac{(10 \log_{10} x - \mu)^2}{2\sigma^2}\right], \quad x > 0, \quad (29)$$

where μ is the mean power level decrease in dB and σ^2 is the variance of the power level expressed in dB due to shadowing. The dimension of σ^2 is dB².

The CDF of $p_{\alpha_0}(x)$ is $F_{\alpha_0}(x) = 1 - \exp[-x^2/(2\sigma^2)]$, $x \geq 0$. The CDF of $p_{P_0}(x)$ is

$$F_{P_0}(x) = \frac{1}{2} \left[1 + \operatorname{erf}\left(\frac{10 \log_{10} x - \mu}{\sigma \sqrt{2}}\right) \right], \quad x > 0, \quad (30)$$

with the error function being defined by

$$\operatorname{erf}(z) = \frac{2}{\sqrt{\pi}} \int_0^z \exp(-t^2) dt. \quad (31)$$

2. Near echoes. A number of near echoes appear in the close vicinity of the receiver, with excess delays not exceeding $\tau_e = 600$ ns. Most of the echoes will occur in this delay interval. The mean power of near echoes $S(\tau) = E\{\alpha_m^2\}$ is exponentially decreasing: $S(\tau) = S_0 \exp(-\mu\tau)$. Given a mean echo power $S(\tau)$ for a fixed delay τ , the amplitude $\alpha_m^{(n)}$ of the near echoes will vary around this mean value according to a Rayleigh distribution with $2\sigma^2 = S(\tau)$. The number of near echoes is Poisson-distributed, with mean λ . Recall that the Poisson distribution provides the probability that a certain number of independent events occur in a given interval (of time or space) when, on average, λ events occur in that interval [27]. The corresponding pdf is

$$p(x) = e^{-\lambda} \sum_{n=0}^{\infty} \frac{\lambda^n}{n!} \delta(x - n) \quad (32)$$

and the CDF is

$$F(x) = e^{-\lambda} \sum_{n=0}^{\lfloor x \rfloor} \frac{\lambda^n}{n!}, \quad (33)$$

with $\lfloor x \rfloor$ denoting the largest integer not exceeding x . The delay distribution $\Delta\tau_m^{(m)}$ of the near echoes follows an exponential distribution with pdf $p_{\Delta\tau_n}(x) = b^{-1} \exp(-x/b)$ and corresponding CDF $F_{\Delta\tau_n}(x) = 1 - \exp(-x/b)$, $x \geq 0$.

The mean power of the near echoes $S(\tau) = E\{\alpha_k^2\}$ is exponentially decreasing with the delay $S(\tau) = S_0 \exp(-\nu\tau)$, $\nu > 0$, or in logarithmic scale $S_{dB}(\tau) = S_{0,dB} - d\tau_{\mu s}$, $d > 0$, with d being expressed in dB/ μs .

For the adopted parameters of the near echoes, see Table A2.

3. Far echoes. The number of far echoes is Poisson-distributed. The far echoes appear with delays $\tau_e < \Delta\tau_m \leq \tau_{\max}$. The amplitudes $\alpha_m^{(f)}$ of the far echoes follow a Rayleigh distribution. The delays $\Delta\tau_m^{(f)}$ of the far echoes are uniformly distributed in $[\tau_e, \tau_{\max}]$. The adopted parameters of the far echoes are indicated in Table A3.

The direct ray is affected by obstacles, such as trees, whereas the echoes are affected by the presence of reflectors (buildings, mountains, etc.). The set of parameters are distinguished by the environments: the near echoes are determined by the foreground environment and the far echoes are determined by the background environment.

3.2. NN Characterization

In this work, we assume supervised learning for the NN, which relies on learning from a dataset with labels for each of the examples. There are two types of supervised learning: classification and regression [28]. Classification is used to determine the class that the data belong to and regression extracts a real value from the data. We consider the application of two different NN types to the problem of detecting GNSS signals affected by multipath aspects: a multilayer perceptron (MLP) and a convolutional NN (CNN). CNNs have been extensively applied in different fields, including computer vision, speech processing, face recognition, etc. [18]. Unlike the MLPs, which are conventional fully connected networks, shared weights and local connections are employed in the CNNs to make full use of 2D input-data structures, like image signals. This operation utilizes an extremely small number of parameters, which both simplifies the training process and speeds up the network [29].

For both NNs, we have considered the following alternative types of inputs: (mode 1), the correlators complex outputs Z_{ik} , displayed in Figure 1 [time-domain inputs], or (mode 2), the corresponding Discrete Fourier Transform (DFT) [frequency-domain inputs], which are defined as

$$Y_{rs} = \sum_{i=0}^{N_f-1} \exp\left(-j\frac{2\pi ir}{N_f}\right) \sum_{k=0}^{N_c-1} Z_{ik} \exp\left(-j\frac{2\pi ks}{N_c}\right), \quad (34)$$

with $r = 0, \dots, N_f - 1$ and $s = 0, \dots, N_c - 1$.

The use of the two types of NN inputs was motivated by the fact that, although signal processing concerning multipath detection and mitigation is predominantly carried out in the time domain, some research has also been performed in the frequency domain [30].

For the MLP, we have considered the structure depicted in Figure 3, which contains a single hidden layer with N_h neurons. In contrast, the architecture of the adopted CNN consists of two main parts: feature extractors and a classifier. In the feature extraction layers, each layer of the network receives the output from its immediate previous layer as its input and passes its output to the input of the next layer. CNNs are built by repeatedly concatenating three classes of layers: convolutional, activation, and pooling. This structure is followed by a last stage that contains three fully connected layers and a classification layer [18,31]. The block diagram is shown in Figure 4. In the convolution block, the pair (a, b) indicates the filter size ($a \times a$) and the number of filters (b). In the max-pooling block, the pair (c, d) indicates the pool size ($c \times c$) and the 2D stride ($d \times d$), where d is the common horizontal and vertical step size for traversing input. The NNs were implemented using Matlab, version R2022a.

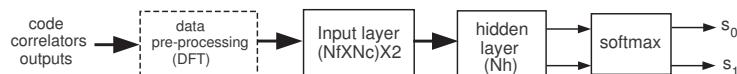


Figure 3. Block diagram of the MLP.

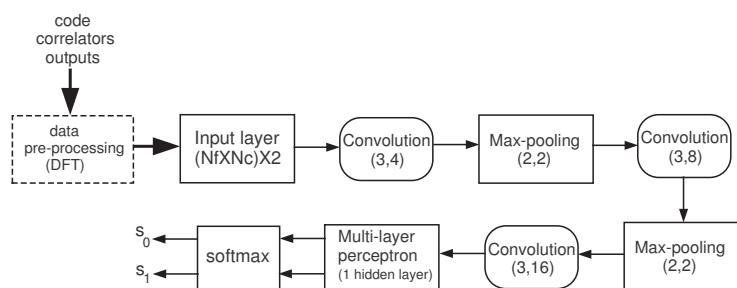


Figure 4. Block diagram of the CNN.

The convolutional layer performs feature extraction by convolving the input with filters (kernels). After each convolution layer, a nonlinear activation layer is applied. We used the ReLU activation function $f(x) = \max\{0, x\}$. The (sub-sampling) pooling layer performs nonlinear downsampling operations, which aims at reducing the spatial size of the representation while simultaneously decreasing the number of parameters, the possibility of overfitting, and the computational complexity of the network. The max-pooling function is used. In the last layer of the fully connected network, the softmax activation function is applied.

The NN is used as a binary classifier with classes H_0 (no multipath) and H_1 (multipath). In binary classification, the information about the success of a model is conveniently described by the confusion matrix, which contains four elements: true negative (TN), true positive (TP), false positive (FP), and false negative (FN) decisions. Accuracy is an informative measure of success, being defined as $1 - \text{Prob}\{\text{decision error}\}$, or

$$\text{accuracy} = 1 - (P_{fa} \cdot \text{Prob}\{H_0\} + P_{md} \cdot \text{Prob}\{H_1\}), \quad (35)$$

with P_{fa} and P_{md} denoting, respectively, the probabilities of false alarm and missed detection (or miss rate) (see, for instance, [32]). Thus,

$$\begin{aligned} P_{fa} &= \text{Prob}\{D_1|H_0\} = \frac{FP}{TN + FP}, \\ P_{md} &= \text{Prob}\{D_0|H_1\} = \frac{FN}{TP + FN}, \end{aligned} \quad (36)$$

where D_0 and D_1 stand for decision in favor of classes H_0 and H_1 , respectively. Additionally,

$$\begin{aligned} \text{Prob}\{H_0\} &= \frac{TN + FP}{TN + FP + TP + FN}, \\ \text{Prob}\{H_1\} &= 1 - \text{Prob}\{H_0\} = \frac{TP + FN}{TN + FP + TP + FN}. \end{aligned} \quad (37)$$

Previous formulas lead to [28,29]

$$\text{accuracy} = \frac{TN + TP}{TP + FN + FP + TN}, \quad (38)$$

which is the evaluation metric we are going to use throughout this work. Other common metrics, different from accuracy, are the precision and the recall [28,29]. Precision is the fraction of multipath detections (D_1) that are correct, while recall is the fraction of multipath events that were detected. That is, precision is $P = TP / (TP + FP)$ and recall is $R = TP / (TP + FN)$, with $P_{md} = 1 - R$.

Equation (35) shows that, when the data are unbalanced, accuracy becomes biased towards the majority class and provides a wrong estimate of the decision success. In fact, when $\text{Prob}\{P_0\} \gg \text{Prob}\{P_1\}$, the accuracy will be almost independent of P_{md} , and, when $\text{Prob}\{P_0\} \ll \text{Prob}\{P_1\}$, the accuracy will be essentially insensitive to P_{fa} . In those cases, metrics such as the F1 score, defined as $F1 = 2R \cdot P / (R + P)$, provide more realistic results [28].

Consider that the output of the NN is the vector $\mathbf{O} = (O_0, O_1)$. The hard tentative decision generated by the NN is $\arg \max_k O_k$. Alternatively, we can normalize the network outputs using the softmax function

$$s_k = \frac{\exp(O_k)}{\exp(O_0) + \exp(O_1)}, \quad k = 0, 1, \quad (39)$$

to compute the probability of each class. The softmax function maps the real-value network output to a probability distribution over a number of classes, where the number of classes equals the number of neurons in the final layer [28]. Notice that $s_0 + s_1 = 1$.

4. Simulation Results

4.1. Single Observation Decisions

Before training the NN, we establish the maximum expected value of the Doppler frequency deviation of the reflected rays due to the multipath effect for the scenario under analysis. This value depends on the dynamics of the user (receiver) and the possible motion of the reflectors. Following the example referred to in Section 3 (typical car speed in a highway), we set the maximum absolute Doppler frequency deviation used by the grid of correlators to $f_{\text{maxtrain}} = 400$ Hz (or Doppler frequency range of ± 400 Hz). The resulting frequency step in the bank of heterodyning units of Figure 1 is then $\Delta f = 2f_{\text{maxtrain}}/N_f = 800/N_f$. Taking into account that multipath delayed signals only affect the receiver's performance approximately in a chip duration interval [1], the delay step used by the bank of correlators is made equal to $\Delta\tau = 2T_c/N_c$.

Both NNs are trained with 2D data formatted as $(N_f \times N_c)$ complex matrices, and 100,000 matrices were generated: 80% of the matrices are used to train the network and the remaining 20% are included in the validation set, whose goal is to determine whether the trained model is overfitting. For training and validation purposes, the data are generated assuming equal probabilities for hypotheses H_0 (multipath absent) and H_1 (multipath present). In case H_1 , the data were generated assuming each of the near echo scenarios, open, rural, suburban, urban, and highway, with equal probabilities. The data are produced with different values of C/N_0 in the interval $30 \leq C/N_0 \leq 50$ dB-Hz, with a uniform distribution in dB-Hz units and correlation interval $T = 10$ ms. In all the following results, the BOCs(1,1) modulation will be utilized. The use of modulation BPSK(1) as an alternative has revealed only minor differences in terms of the achieved accuracy. A learning rate of 5×10^{-4} was used with both types on NNs. To achieve the best performance, the range of Doppler frequency deviations, $\pm f_{\text{maxtest}}$, used by the NN in test mode should verify $f_{\text{maxtest}} \leq f_{\text{maxtrain}}$.

The two NN architectures are tested with correlations outputs generated according to (12), with a maximum number of reflected rays $N_r = 5$, assuming multipath and no-multipath scenarios, with equal probabilities. This value of N_r is considered to be a reasonable amount of reflected rays in most multipath environments. In fact, simulations carried out with larger values of N_r have revealed no significant differences in the NN behavior.

The amplitudes of the different rays are $A_0 = 1$ and A_k ($k = 1, \dots, N_r$) following a uniform distribution with $0 < A_k < 1$. The phases θ_k are independent and uniformly distributed in the interval $0 \leq \theta_k < 2\pi$. The delays τ_k are independent and uniformly distributed in the interval $0 < \tau_k < T_c$. The Doppler frequency shifts $f_{d_k} - f_{d_0}$ are independent and uniformly distributed, such that $|f_{d_k} - f_{d_0}| < f_{\text{maxtest}}$. The characterization of amplitudes, phases, delays, and Doppler frequencies as independent uniform random variables, although simplistic, seems to be the natural distribution choice, meaning that we have no a priori statistical knowledge of those variables. It is expected that, with real signals, the NNs have the capacity to withstand the possible non-uniformity of the variables.

Figure 5 compares the accuracies achieved with the MLP and the CNN architectures when time-domain [mode 1] and frequency-domain [mode 2] data are used. The data are formatted as $(N_f \times N_c) = (32 \times 32)$ matrices. In this simulation, we have considered two matching conditions for the Doppler frequencies: (a) $f_{\text{maxtest}} = f_{\text{maxtrain}} = 400$ Hz (matching condition) and (b) $f_{\text{maxtest}} = 50$ Hz, $f_{\text{maxtrain}} = 400$ Hz (mismatch condition). The tests were performed with 40,000 data matrices. An MLP with one hidden layer constituted by $N_h = 25$ neurons is employed in (a) and a convolutional neural NN (CNN) is applied in (b). The number of neurons indicated for the hidden layer of the MLP was found experimentally and is a trade-off between the accuracy achieved by the classifier and the complexity of the algorithm.

Comparing the two sets of plots, we can see that, overall, the results improve for large values of (C/N_0) , as expected, and there is a performance degradation of both NNs in the case of frequency mismatch. In all the cases, the CNN permits to obtain better results

than the MLP, with the best performance being achieved with mode-2 data (black solid curves). It can be shown that this trend is essentially kept even if the values of f_{maxtrain} and f_{maxtest} change. The advantage of the CNN architecture is due to its double role as a feature extractor and classifier. Thus, in order to simplify the analysis, we will henceforth consider only the CNN architecture and mode-2 input data.

In Figure 5b, the NNs were not conveniently trained to match the Doppler frequency range of the test dataset. So, to avoid performance degradation, it is convenient to determine the approximate limits of the Doppler frequency deviation corresponding to each scenario, taking into account the range of the receiver speeds, and use a trained network as matched as possible to that expected range of Doppler frequency deviations.

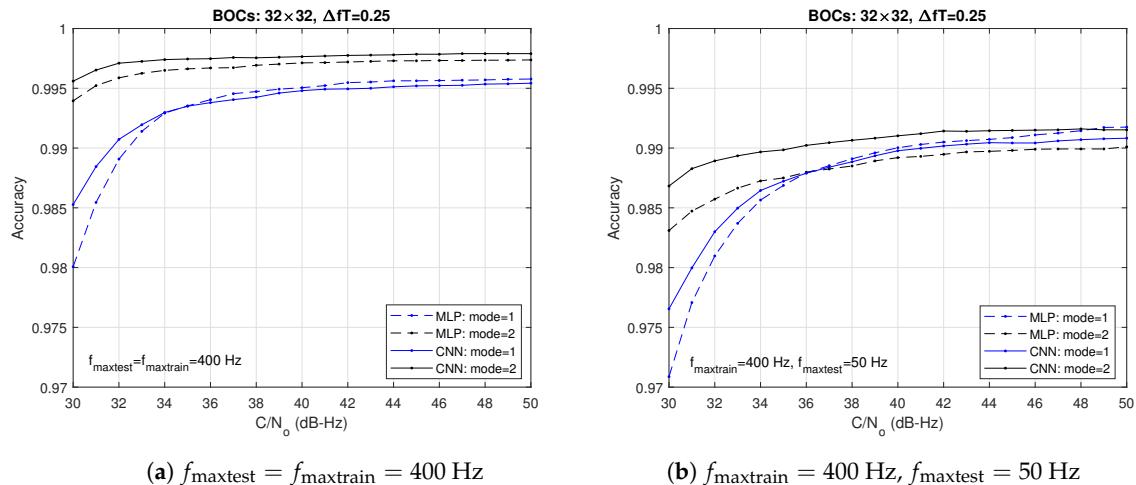


Figure 5. Accuracies obtained when time-domain (blue lines) and frequency-domain inputs (black lines) are used with a multilayer perceptron (MLP) and a convolutional neural network (CNN).

Figure 6 exhibits the accuracies provided by correlators grids of different sizes ($N_f \times N_c$). The NN was trained with $f_{\text{maxtrain}} = 400 \text{ Hz}$. Two cases are considered: (a) matching case with $f_{\text{maxtest}} = f_{\text{maxtrain}}$ and (b) mismatch case with $f_{\text{maxtest}} = f_{\text{maxtrain}}/8$. As expected, the best results are achieved with the largest grid, i.e., (32×32) in the matching case. With the smallest grid, i.e., (16×16) , the degradation is particularly significant for the lowest carrier-to-noise ratios. Moreover, in all the plots, the main factor of degradation is the decrease in the number of frequency steps N_f .

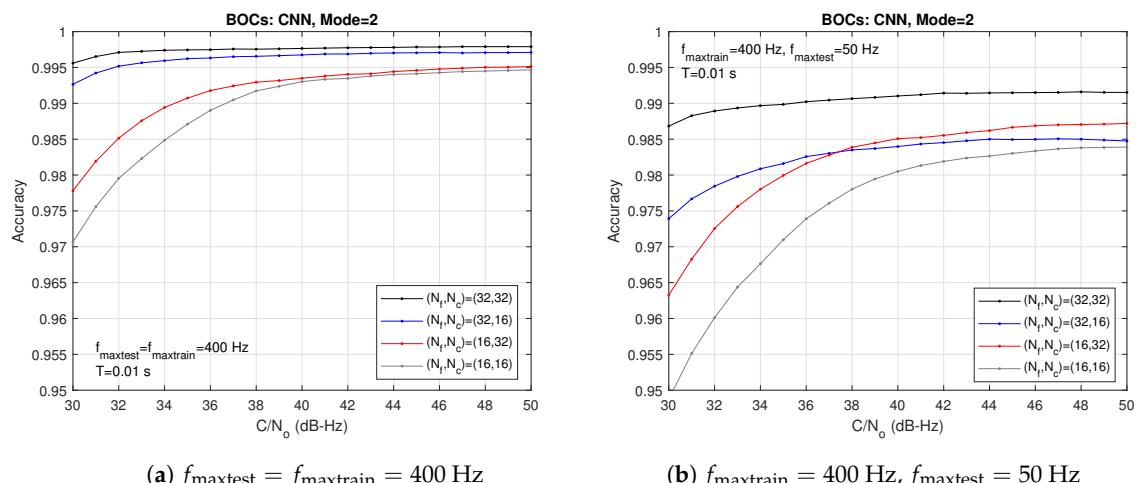


Figure 6. Accuracies obtained with correlators grids of different sizes (N_f, N_c) with $f_{\text{maxtrain}} = 400 \text{ Hz}$.

Figure 7 exhibits the accuracies provided by correlators grids of different sizes ($N_f \times N_c$), but now the NN has been trained with $f_{\text{maxtrain}} = 50$ Hz. Notice the striking contrast between the results of Figures 6 and 7. In the latter case, the performance is almost independent of the number of correlators in the Doppler frequency (N_f), depending mainly on N_c , that is, the number of correlators in the code delay. As a result, Figure 7a,b, generated, respectively, with frequency matching and mismatch, are practically equal.

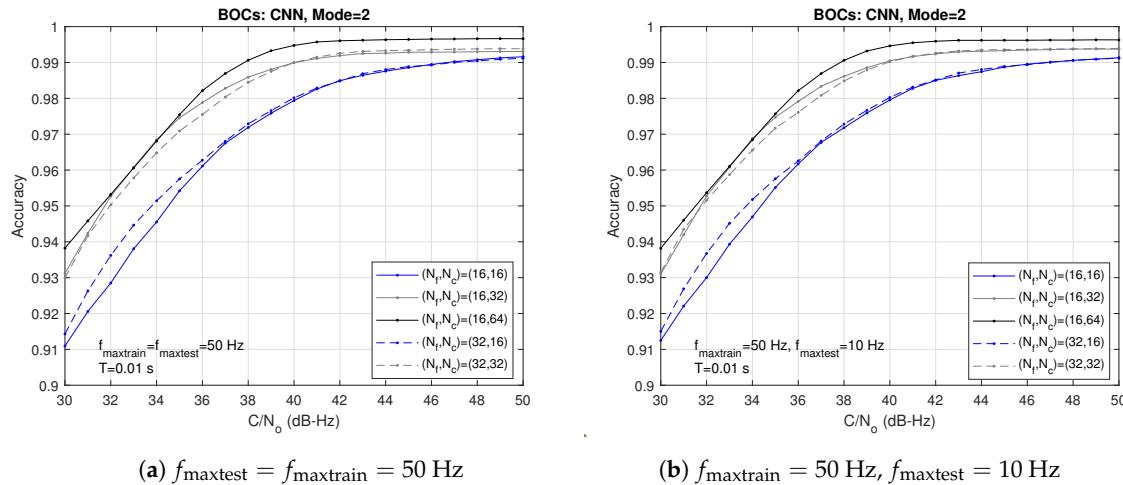


Figure 7. Accuracies obtained with correlators grids of different sizes (N_f, N_c) with $f_{\text{maxtrain}} = 50$ Hz.

The receiver model considered in Figure 1 includes a grid of correlators that use information contained in the time domain (code delay) and frequency domain (Doppler frequency). This architecture was previously used, for instance, in [11] and further referred to in [17]. However, the novelty of our approach is characterized by the following aspects: (i) we developed mathematical tools to analytically define the additive noise components in each correlator, including their cross-correlations, (ii) we implemented models for multipath, during the training of the neural network, based on the DLR model, and (iii) we used DFT pre-processing of the correlators outputs to improve the CNN performance.

4.2. Multi-Observation Decisions

When the product $(C/N_0)T$ is small, the probabilities P_{fa} and/or P_{md} tend to be high, leading to a significant decrease in the accuracy. This drawback is minimized by making decisions based on multiple observations of the grid of correlators in N adjacent intervals of duration T .

With hard tentative multipath/no multipath decisions generated by the NN in N consecutive correlation intervals (observations), we may apply the following criterion: select class H_1 only when there are, at least, M positive tentative decisions, with $1 \leq M \leq N$ (M-of-N selection). We refer to a hard decision regarding this methodology based on the M-of-N selection. Let the probabilities of false alarm and missed detection for each tentative decision be, respectively, P_{fa} and $P_{md} = 1 - P_d$, as defined in (36). The overall probability of false alarm (under hypothesis H_0) is

$$\tilde{P}_{fa} = \sum_{n=M}^N \binom{N}{n} P_{fa}^n (1 - P_{fa})^{N-n} \quad (40)$$

and the overall probability of correct decision (under hypothesis H_1) is

$$\tilde{P}_d = \sum_{n=M}^N \binom{N}{n} P_d^n (1 - P_d)^{N-n}. \quad (41)$$

Hence, the overall probability of missed detection is provided by $\tilde{P}_{md} = 1 - \tilde{P}_d$.

Ideally, for multiple observations ($N > 1$), the overall probabilities of false alarm and missed detection should decrease simultaneously relative to the corresponding probabilities of a single observation ($N = 1$). However, this is achievable only for certain values of P_{fa} , P_{md} and M , as illustrated in Figure 8 for $N = 4$. In general, for a given value of P_{fa} and N being constant, \tilde{P}_{fa} decreases when M grows, and, for a fixed value of P_{md} and N being constant, \tilde{P}_{md} increases when M grows.

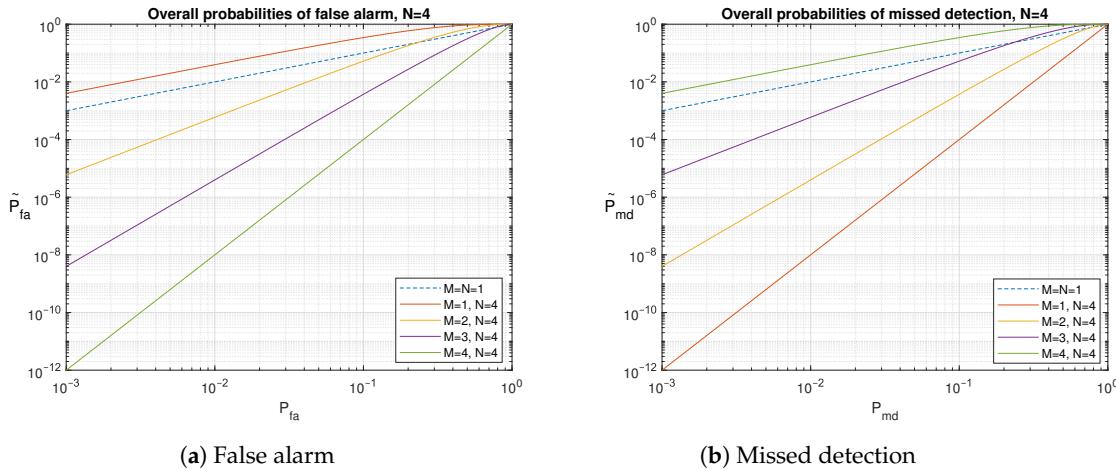


Figure 8. Example of overall probabilities of false alarm and missed detection for $N = 4$.

When N softmax values $s_0^{(n)}$ and $s_1^{(n)}$, $n = 1, \dots, N$, are available in consecutive correlation intervals, we may add them before producing a final decision, according to

$$S_0 = \frac{1}{N} \sum_{n=1}^N s_0^{(n)}, \quad S_1 = \frac{1}{N} \sum_{n=1}^N s_1^{(n)} \quad (42)$$

and deciding in favor of hypothesis H_0 , if $S_0 > S_1$, or H_1 otherwise. We refer to this as the soft decision to this methodology, in contrast to the alternative hard decision.

Figure 9 depicts the accuracies obtained with multiple observations using hard and soft decisions. The correlation interval is $T = 4$ ms, which means that the classifier produces a decision every $NT = 4N$ ms. For the hard decision case, $N = 4$ correlation intervals were used. Note that, for $N = 4$, the best results achieved in both plots are approximately equivalent. However, using soft decisions, the accuracy is practically independent of N , provided that $N \geq 3$. As a consequence, soft decisions are preferable to hard decisions (at least for this example).

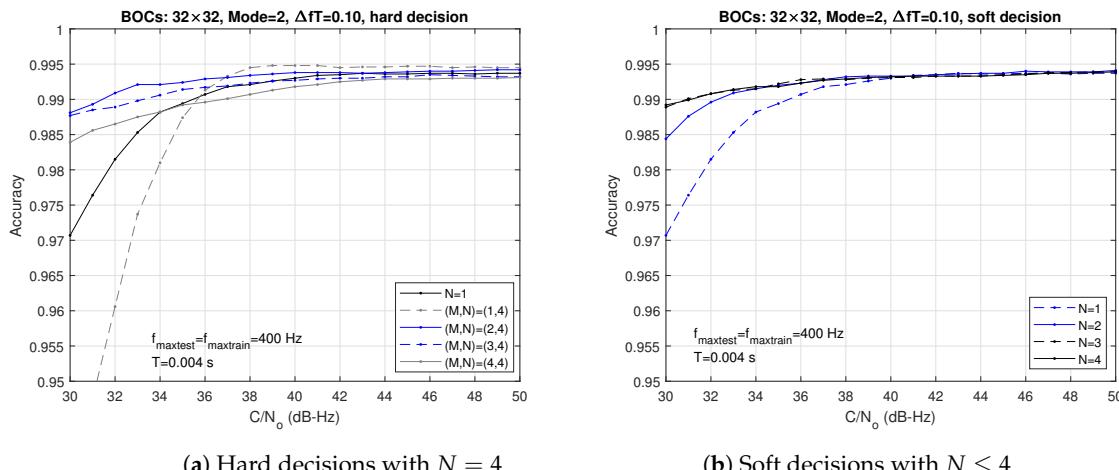


Figure 9. Accuracies obtained with multi-observation decisions and $T = 4$ ms.

4.3. Multipath Mitigation Technique Using Soft Decisions

With the goal of understanding the motivation for having an algorithm that detects the effect of multipath in a given received signal, consider the scenario described in Figure 10. The figure displays the three-dimensional (3D) positioning errors obtained with five GPS satellites in view (SV2, SV18, SV26, SV29, and SV31) when one of the received signals is affected by multipath, provoking an increment of 50 m in the corresponding pseudorange. In the absence of multipath, the rms positioning error is 22.0 m. This value includes the effect of thermal noise (it is assumed that the pseudoranges are disturbed by independent zero-mean Gaussian noises with standard deviation equal to 5 m). The circles indicate the rms positioning errors achieved when the pseudorange associated with multipath is included in the least-squares navigation solution (single-point solution, [33]). The different errors depend on the position of the corresponding satellite through the geometric dilution of precision (GDOP). The squares indicate the resulting errors when the pseudorange measurement of the multipath-disturbed signal is removed from the solution (strategy I).

The figure shows that, in most scenarios, if reliable information is available concerning the presence of multipath in a certain signal, the removal of that signal from the equation of navigation permits to reduce the resulting positioning error (although at the cost of a slight GDOP increase). However, when the SV31 signal is affected by a multipath effect and is removed from the equation of navigation, a substantial increment of the positioning errors is obtained. The explanation for this anomalous behavior is the very large growth in the GDOP that results from discarding SV31. This example enables concluding that strategy I (signal removal) may not be the best one if it leads to the significant growth of the GDOP parameter. We will see next that an alternative strategy (strategy II), based on the outputs of the NN, may provide a better multipath mitigation performance.

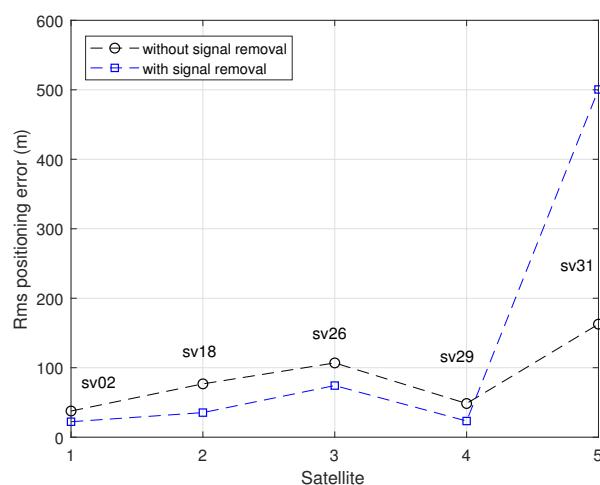


Figure 10. 3D positioning errors obtained in the presence of one multipath-disturbed signal when the corresponding pseudorange measurement is either used (circles) or is removed from the navigation equation (squares).

In fact, the availability of softmax values S_k permits to devise new strategies to mitigate the multipath effect on the position, velocity, and time (PVT) solution as an alternative to removing the affected signal(s) from the equation of navigation. As the degree of confidence in the classification provided by the NN increases with the difference $|S_1 - S_0|$, a weighted least-squares algorithm (or an extended Kalman filter having the observations noise covariance matrix with adjusted inputs) that uses K signals (with $K \geq 5$) may provide a more accurate estimate of the position, velocity, and time (PVT) than the least-squares solution based on $K - 1$ signals. This strategy is particularly useful when a reduced number of visible satellites are available because the removal of one or more signals from

the equation of navigation could significantly increase the GDOP or even prevent the computation of the PVT solution.

Suppose $K \geq 4$ satellites are being tracked. The pseudorange measurements, after correction of the ionospheric/tropospheric delays, are provided by [3]

$$r_i = \sqrt{(X_i - x)^2 + (Y_i - y)^2 + (Z_i - z)^2} + \Delta_r c, \quad i = 1, \dots, K, \quad (43)$$

where (X_i, Y_i, Z_i) is the satellite position, (x, y, z) is the receiver position, and $\Delta_r c$ is the range bias due to the receiver clock offset Δ_r .

A least-squares solution $(\hat{x}, \hat{y}, \hat{z}, \hat{\Delta}_r c)$ is difficult to obtain analytically for the set of pseudorange measurements (43) because the measurements are nonlinear, but the equations may be linearized by performing a Taylor expansion regarding the predicted user position and range bias $(\tilde{x}, \tilde{y}, \tilde{z}, \tilde{\Delta}_r c)$. The single-point navigation solution using the weighted least-squares estimate is [34]

$$\begin{bmatrix} \hat{x} \\ \hat{y} \\ \hat{z} \\ \hat{\Delta}_r c \end{bmatrix} = \begin{bmatrix} \tilde{x} \\ \tilde{y} \\ \tilde{z} \\ \tilde{\Delta}_r c \end{bmatrix} + (G^T W G)^{-1} G^T W \begin{bmatrix} r_1 - \tilde{r}_1 \\ r_2 - \tilde{r}_2 \\ \vdots \\ r_K - \tilde{r}_K \end{bmatrix}, \quad (44)$$

where G is the geometry matrix, W is the weighting matrix, assumed to be positive definite and symmetric, and \tilde{r}_i is the predicted pseudorange. Let $\Lambda_i = S_{1,i} - S_{0,i} = 1 - 2S_{0,i}$, $i = 1, \dots, K$, denote the difference of the two softmax outputs for satellite i , with $0 \leq S_{0,i} \leq 1$, where $\Lambda_i = -1$ indicates a no-multipath scenario with probability one and $\Lambda_i = 1$ a multipath scenario with probability one. In contrast, $\Lambda_i = 0$ corresponds to a minimum of confidence in the NN decision. Therefore, we propose the following $(K \times K)$ weighting matrix

$$W = \begin{bmatrix} (1 - \Lambda_1)^\gamma & 0 & \dots & 0 \\ 0 & (1 - \Lambda_2)^\gamma & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & (1 - \Lambda_K)^\gamma \end{bmatrix} = 2^\gamma \begin{bmatrix} S_{0,1}^\gamma & 0 & \dots & 0 \\ 0 & S_{0,2}^\gamma & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \dots & S_{0,K}^\gamma \end{bmatrix} \quad (45)$$

where γ (with $\gamma \geq 0$) is a selectable parameter that enables adjusting the range of values of the weights in W . For instance, conducting $\gamma = 0$, the estimated position corresponds to the conventional least-squares solution obtained with K satellites because W is the identity matrix. The $\gamma \rightarrow \infty$ estimated solution is tantamount to the least-squares solution obtained with $K - 1$ satellites (the signal affected by multipath aspects is discarded) as $(1 - \Lambda_i)^\gamma \rightarrow \infty$, if $S_{1,i} < S_{0,i}$, and $(1 - \Lambda_i)^\gamma \rightarrow 0$, if $S_{1,i} > S_{0,i}$. For the remaining values of γ , a weighted least-squares solution with K satellites is computed.

Figure 11 displays the rms 3D positioning errors versus γ for the same scenario described in Figure 10, provided by (44) and (45). In this simulation, we assumed that $S_{0,i} = 0.9$ for satellite signals free of multipath and $S_{0,i} = 0.1$ for satellite signals affected by multipath. Figure 11 shows that, when the signals of satellites SV02, SV18, SV26, or SV29 are affected by multipath aspects, the best solution consists of removing that signal from the equation of navigation (strategy I), but this strategy leads to poor results when the disturbed signal belongs to SV31. Overall, strategy II, which consists of using (44) with $\gamma \approx 2$, provides the best results in terms of multipath mitigation.

If, instead, an extended Kalman filter was employed, the weighting matrix W could be replaced by the covariance matrix, $R_n + R_m$, in the measurement (observations) model [35], with R_n depending on the noise pseudorange terms and $R_m \propto W^{-1}$ depending on the multipath effect.

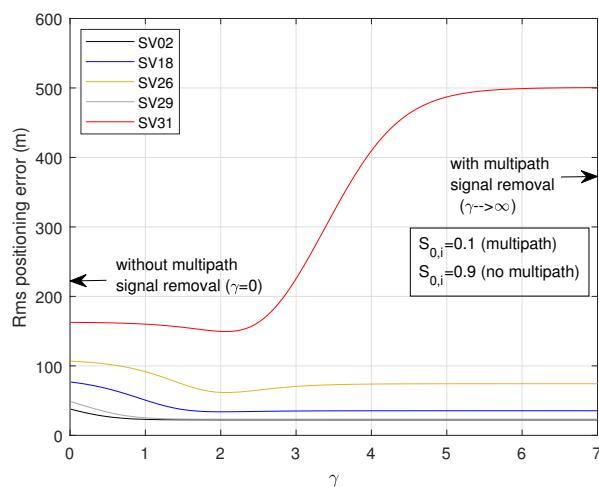


Figure 11. 3D positioning errors obtained with one multipath-disturbed signal using softmax values and weighted least-squares solutions with different values of γ .

5. Conclusions

In this work, we proposed algorithms for the detection of GNSS signals affected by multipath aspects based on two neural network architectures: the multilayer perceptron (MLP) and the convolutional neural network (CNN). Extensive simulations have shown that the CNN is, in general, superior to the MLP, thus becoming the adopted solution. To obtain a classifier that is robust to various types of multipath aspects, the network was trained with synthetic noisy signals generated from a blend of different multipath scenarios that characterize the DLR model. The testing of the neural networks was performed with a set of multipath scenarios affected or not by the multipath effect with equal probabilities. No specific model was adopted for the testing task in order to make it as general as possible.

In the GNSS receiver, the neural network inputs were produced by a grid of $N_f N_c$ correlators that covered the Doppler frequencies and the code delays of the different received replicas. The number of correlators has a major impact on the computational effort and should be kept as small as possible. However, decreasing the number of correlators, in particular those in the frequency domain (N_f), tends to negatively affect the performance of the classifier. Good results were obtained for the Doppler range of ± 400 Hz by conducting $N_f = 32$ and $N_c = 16$.

Since the neural network was trained using Monte Carlo techniques, increased emphasis was placed on the analytic characterization of the cross-correlations between the noise components in the different correlators. It was also found that pre-processing the CNN inputs with a 2D Discrete Fourier Transform enabled significantly improving the detection performance in certain scenarios. The performance achieved with the CNN was compared with the MLP, which was used as a benchmark. In general, the CNN provided better results than the MLP, especially in low-signal-to-noise-ratio conditions. Thus, the CNN was used as the main multipath detection algorithm in the current work. The explanation for the better performance of the CNN compared to the MLP architecture is that the former includes an initial part (convolutional and max-pooling layers) to perform automatic feature extraction before carrying out the classification task. In fact, the CNN may be viewed as a feature extractor followed by a classifier.

Depending on the type of the CNN outputs, hard or soft decisions could be utilized. When hard decisions are used, the GNSS signal affected by the multipath aspects is removed from the equation of navigation. The adoption of soft decisions enables implementing an alternative strategy for multipath mitigation, which consists of solving the equation of navigation using a weighted least-squares algorithm (or an extended Kalman filter), with the processing of each GNSS signal being affected by a different weight (or probability) provided by the neural network. The result is a decrease in the receiver's position, velocity,

and timing errors while keeping the number of processed signals unchanged, which may be a significant advantage when the presence of visible satellites has already been reduced.

Author Contributions: Conceptualization, F.N. and F.S.; Writing—original draft, F.N. and F.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by FCT—Fundação para a Ciência e Tecnologia, I.P. by project reference UIDB/50008/2020, and DOI identifier <https://doi.org/10.54499/UIDB/50008/2020>.

Data Availability Statement: Data are contained within the article.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A. Tables of DLR Model Parameters

The following tables were obtained using data from [20]. The second column of Table A1 indicates the shadowing probabilities.

Table A1. Parameters for the direct path.

Environment	Shad. Prob.	ρ (dB) [No Shad]	μ (dB) [Shad]	σ (dB) [Shad]
open, 25°	0	6.0	-	-
45°	0	10.4	-	-
rural, 25°	0.96	10.8	-9.9	3.3
45°	0.79	4.7	-5.4	2.3
suburban, 25°	0.59	4.7	-6.0	3.5
45°	0.43	4.0	-7.2	3.2
urban, 25°	0.79	3.2	-12.1	6.3
45°	0.56	8.5	-3.0	2.7
highway, 25°	0.19	8.4	-5.8	1.7
45°	0	7.8	-	-

Table A2. Parameters for near echoes.

Environment	λ	τ_e (ns)	b (μ s)	S_0 (dB)	d (dB/ μ s)
open, 25°	1.2	400	0.03	-28.6	1.0
45°	0.5	400	0.027	-29.0	1.1
rural, 25°	1.5	400	0.055	-24.9	19.2
45°	1.8	400	0.051	-24.5	13.4
suburban, 25°	1.4	400	0.038	-23.8	23.7
45°	1.5	400	0.027	-24.4	23.0
urban, 25°	4.0	600	0.063	-17.0	26.2
45°	3.6	600	0.081	-23.5	8.5
highway, 25°	2.2	600	0.077	-25.8	7.3
45°	1.8	600	0.043	-27.1	29.5

Table A3. Parameters for far echoes.

Environment	λ	$2\sigma^2$ (dB)	τ_{max} (μ s)
flat terrain, 25°	0.3	-26.4	15
45°	-	-	15
rural, 25°	0.8	-28.2	5
45°	-	-	5

Table A3. Cont.

Environment	λ	$2\sigma^2$ (dB)	τ_{max} (μ s)
hilly, 25°	1.2	-29.0	10
	45°	-	10
mountainous, 25°	1.8	-28.5	15
	45°	-21.7	15

Appendix B. Generation of a Gaussian Vector with a Given Covariance Matrix

Consider the generation of a generic Gaussian noise vector $U = [u_1, \dots, u_M]^T$, with zero mean and covariance matrix \mathcal{C} , from the noise vector $W = [w_1, \dots, w_M]^T$ of zero-mean independent Gaussian components with unity variance. Let $U = GW$, where $G(M \times M)$ is a square matrix. The problem consists of determining G from $\mathcal{C} = GE\{WW^T\}G^T = GG^T$. According to the spectral theorem for real symmetric matrices [36], the covariance matrix can be written in terms of its eigenvalues and eigenvectors as $\mathcal{C} = V\Lambda V^T$, where V is an orthogonal matrix whose columns are the orthonormal eigenvectors of \mathcal{C} and $\Lambda = \text{diag}\{\lambda_1, \lambda_2, \dots, \lambda_M\}$ is a diagonal matrix with the real nonnegative eigenvalues of \mathcal{C} . This leads to $G = V\Lambda^{1/2} = V\text{diag}\{\sqrt{\lambda_1}, \sqrt{\lambda_2}, \dots, \sqrt{\lambda_M}\}$.

References

- Misra, P.; Enge, P. *Global Positioning System: Signals, Measurements, and Performance*, 2nd ed.; Ganga-Jamuna Press: Lincoln, MA, USA, 2006.
- Kaplan, E.D.; Hegarty, C.J. *Understanding GPS. Principles and Applications*, 2nd ed.; Artech House: Boston, MA, USA, 2006.
- Betz, J.W. *Engineering Satellite-Based Navigation and Timing*; IEEE Press Hoboken, NJ, USA; Wiley: Hoboken, NJ, USA, 2016.
- Morrisey, T.; Shallberg, K.; Townsend, B. Code Tracking Errors for Double Delta Discriminators with Narrow Correlator Spacings and Bandlimited Receivers. In Proceedings of the ION NTM 2006, Monterey, CA, USA, 18–20 January 2006; pp. 914–926.
- Townsend, R.; Fenton, P.; van Dierendonck, K.; van Nee, D. Performance Evaluation of the Multipath Estimating Delay Lock Loop. *Navig. J. Inst. Navig.* **1995**, *42*, 503–514. [[CrossRef](#)]
- Vigneau, W.; Nouvel, O.; Manzano-Jurado, M.; Sanz, C.; Abdulkader, H.; Roviras, D.; Juan, J.; Macabiau, C.; Holsters, P. Neural Networks Algorithms Prototyping to Mitigate GNSS Multipath for LEO Positioning Applications. In Proceedings of the ION GNSS 2006, Fort Worth, TX, USA, 26–29 September 2006; pp. 1752–1762.
- Quan, Y.; Lau, L.; Roberts, G.; Meng, X.; Zhang, C. Convolutional Neural Network Based Multipath Detection Method for Static and Kinematic GPS High Precision Positioning. *Remote Sens.* **2018**, *10*, 2052. [[CrossRef](#)]
- Savas, C.; Dovis, F. Multipath Detection based on K-means Clustering. In Proceedings of the ION GNSS+2019, Miami, FL, USA, 16–20 September 2019; pp. 3801–3811. [[CrossRef](#)]
- Munin, E.; Blais, A.; Couellan, N. Convolutional Neural Network for Multipath Detection in GNSS Receivers. In Proceedings of the AIDA-AT, Singapore, 3–4 February 2020; pp. 1–10. [[CrossRef](#)]
- Suzuki, T.; Amano, Y. NLOS Multipath Classification of GNSS Signal Correlation Output Using Machine Learning. *Sensors* **2021**, *21*, 2503. [[CrossRef](#)] [[PubMed](#)]
- Blais, A.; Couellan, N.; Munin, E. A Novel Image Representation of GNSS Correlation for Deep Learning Multipath Detection. *Array* **2022**, *14*, 100167. [[CrossRef](#)]
- Guillard, A.; Thevenon, P.; Milner, C. Using convolutional neural networks to detect GNSS multipath. *Front. Robot. AI* **2023**, *10*, 1106439. [[CrossRef](#)] [[PubMed](#)]
- Guillard, A.; Thevenon, P.; Milner, C.; Macabiau, C. Benefits of CNN-Based Multipath Detection for Robust GNSS Positioning. In Proceedings of the 36th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2023), Denver, CO, USA, 11–15 September 2023; pp. 283–297. [[CrossRef](#)]
- Li, H.; Borhani-Darian, P.; Wu, P.; Closas, P. Deep Neural Network Correlators for GNSS Multipath Mitigation. *IEEE Trans. Aerosp. Electron. Syst.* **2023**, *59*, 1249–1259. [[CrossRef](#)]
- Qi, X.; Xu, B.; Wang, Z.; Hsu, L.T. Rain forest-based multipath parameter estimation. *GPS Solut.* **2024**, *2024*, 126. [[CrossRef](#)]
- Mohanty, A.; Gao, G. A survey of machine learning techniques for improving Global Navigation Satellite Systems. *EURASIP J. Adv. Signal Process.* **2024**, *2024*, 73. [[CrossRef](#)]
- Xu, P.; Zhang, G.; Yang, B.; Hsu, L.T. Machine Learning in GNSS Multipath/NLOS Mitigation: Review and Benchmark. In *IEEE Aerospace and Electronic Systems Magazine*; IEEE: New York, NY, USA, 2024. [[CrossRef](#)]
- Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016.
- Jahn, A.; Buonomo, S.; Sforza, M.; Lutz, E. A Wideband Channel Model for Land Mobile Satellite Systems. In Proceedings of the IMSC'95, Ottawa, ON, Canada, 6–8 June 1995; pp. 122–127.

20. Jahn, A.; Bischl, H.; Heiss, G. Channel Characterisation for Spread Spectrum Satellite Communications. In Proceedings of the ISSSTA'95, Mainz, Germany, 25 September 1996; pp. 1221–1226. [[CrossRef](#)]
21. ITU-R. Radiocommunication Sector of ITU. *Propagation Data Required for the Design Systems in the Land Mobile-Satellite Service, Recommendation ITU-R P.681-11*; International Telecommunication Union: Geneva, Switzerland, 2019.
22. Borre, K.; Fernández-Hernández, I.; López-Salcedo, J.A.; Buhiyan, M.Z.H. (Eds.) *GNSS Software Receivers*; Cambridge University Press: Cambridge, UK, 2023.
23. Lutz, E. Modelling of the satellite communications channel—with emphasis on the land mobile satellite channel. In Proceedings of the ESTEL, Rome, Italy, 2–5 October 2012. [[CrossRef](#)]
24. Galileo. *Signal-in-Space Interface Control Document, Open Service*, ver. 2.0; European GNSS (Galileo): Madrid, Spain, 2021.
25. Proakis, J.G. *Digital Communications*, 4th ed.; McGraw-Hill: New York, NY, USA, 2001.
26. Suzuki, H. A Statistical Model for Urban Radio Propagation. *IEEE Trans. Commun.* **1977**, *25*, 673–680. [[CrossRef](#)]
27. Papoulis, A. *Probability, Random Variables, and Stochastic Processes*, 3rd ed.; McGraw-Hill: New York, NY, USA, 1991.
28. Kamath, U.; Liu, J.; Whitaker, J. *Deep Learning for NLP and Speech Recognition*; Springer: Berlin/Heidelberg, Germany, 2019.
29. Alzubaidi, L.; Zhang, J.; Huaidi, A.; Al-Dujaili, A.; Al-Shamma, O.; Santamaría, J.; Fadhel, M.; Al-Amidie, M.; Farhan, L. Review of deep learning: Concepts, CNN, architectures, challenges, applications, future directions. *J. Big Data* **2021**, *8*, 53. [[CrossRef](#)] [[PubMed](#)]
30. Zhang, Y.; Bartone, C. Multipath Mitigation in the Frequency Domain. In Proceedings of the IEEE PLANS 2004, Monterey, CA, USA, 26–29 April 2004; pp. 486–495. [[CrossRef](#)]
31. Zhang, A.; Lipton, Z.C.; Li, M.; Smola, A.J. *Dive into Deep Learning*; Cambridge University Press: Cambridge, UK, 2024.
32. Kay, S.M. *Fundamentals of Statistical Signal Processing. Detection Theory*; Prentice Hall: Upper Saddle River, NJ, USA, 1998.
33. Axelrad, P.; Brown, R.G. GPS Navigation Algorithms. In *GPS Positioning System: Theory and Applications*; AIAA: Washington, DC, USA, 1996; Volume 1, Chapter 9, pp. 409–433.
34. Groves, P.D. *GNSS, Inertial, and Multisensor Integrated Navigation Systems*; Artech House: Boston, MA, USA, 2008.
35. Brown, R.G.; Hwang, P.Y.C. *Introduction to Random Signals and Applied Kalman Filtering*, 3rd ed.; Wiley: New York, NY, USA, 1997.
36. Banerjee, S.; Roy, A. *Linear Algebra and Matrix Analysis for Statistics*; CRC Press: Boca Raton, FL, USA, 2014.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.