Distributional loss for convolutional neural network regression and application to parameter estimation in satellite navigation signals

Thomas Gonzàlez, Antoine Blais, Nicolas Couëllan, Christian Ruiz

Distributional loss for convolutional neural network regression and application to parameter estimation in satellite navigation signals

Thomas Gonzàlez[a,b], Antoine Blais[b], Nicolas Couëllan[b,c,*], Christian Ruiz[a]

[a] *OKTAL-SE, 31320 Vigoulet-Auzil, France*
[b] *ENAC, Université de Toulouse, 7 Avenue Édouard Belin, BP 54005, 31055 Toulouse Cedex 4, France*
[c] *Institut de Mathématiques de Toulouse, Université de Toulouse, UPS IMT, F-31062 Toulouse Cedex 9, France*

Abstract

Convolutional Neural Network (CNN) have been widely used in image classification. Over the years, they have also benefited from various enhancements and they are now considered state-of-the-art techniques for image-like data. However, when they are used for regression to estimate some function value from images, few recommendations are available to construct robust CNN regressor models. In this study, a robustness enforcing mechanism is proposed for CNN regression models. It combines convolutional neural layers to extract high level features representations from images with a soft labelling technique that helps generalization performance. More specifically, as the deep regression task is challenging, the idea is to account for some uncertainty in the targets that are seen as distributions around their mean. Building from earlier work (Imani & White, 2018), a specific histogram loss function based on the Kullback-Leibler (KL) divergence is applied during training. The prior distributions are selected according to the physical characteristics of the parameters to estimate. To assess and illustrate the technique, the model is applied to Global Navigation Satellite System (GNSS) multipath estimation where multipath signal parameters have to be estimated from correlator output images from the I and Q channels. The multipath signal delay, magnitude, Doppler shift frequency and phase parameters are estimated from synthetically generated datasets of satellite signals. Experiments are conducted under various receiving conditions and various input images resolutions to test the estimation performances quality and robustness. The results show that the proposed soft labelling CNN technique using distributional loss outperforms classical CNN regression under all conditions. Furthermore, the extra learning performance achieved by the model

*Corresponding author
Email addresses: thomas.gonzalez@oktal-se.fr (Thomas Gonzàlez), antoine.blais@recherche.enac.fr (Antoine Blais), nicolas.couellan@recherche.enac.fr (Nicolas Couëllan), christian.ruiz@oktal-se.fr (Christian Ruiz)

allows the reduction of input image resolution from 80x80 down to 40x40 or sometimes 20x20.

Keywords: Convolutional neural network, distributional loss, GNSS, image regression, histogram loss, Kullback-Leibler divergence, multipath estimation, soft labelling.

---

1. Introduction

There has been a growing interest for CNN (Goodfellow et al., 2016) in the machine learning community in order to construct high level features from images. Mostly used in images classification, CNN have also been employed to estimate various information from images. Building regression models from image data is a difficult task since complex and high level feature representation is needed. This is why deep architectures are usually considered. Among the traditional examples from the literature, one can refer to (Toshev & Szegedy, 2014), (Mahendran et al., 2017), (Li & Chan, 2014) and (Girshick et al., 2011) where holistic reasoning on Human pose estimation is based on CNN. Age estimation according to face images (Yi et al., 2014) or magnetic resonance images (Ueda et al., 2019) are other successful examples of regression with CNN. They were also applied on X-ray tensor images in (Miao et al., 2016) and recently, in (Lathuilière et al., 2020), the authors have proposed an extensive review of CNN for regressions. To increase the regression model generalization performance, it has been shown that the use of soft labelling techniques can greatly help (Imani & White, 2018; Gao et al., 2017). Among soft labelling techniques, one idea is to consider that labels used during training are uncertain and drawn from a given prior distribution. Training can therefore be carried out at the distribution level rather than single observations. Especially suited when labels are ambiguous or subject to noise, this procedure can also be seen, in the general regression task, as a robustness enforcing alternative to other strategies such as batch normalization (Ioffe & Szegedy, 2015), dropout (Hinton et al., 2012), early stopping (Zhang & Yu, 2005) or regularization (Goodfellow et al., 2016; Couellan, 2021b,a) most often used in classification problems.

In this study, we propose a dedicated CNN regression model that implements distributional loss on GNSS multipath data. The objective is to estimate multipath parameters from two dimensional GNSS correlation images.

A multipath is a parasitic reflection of the signal of interest which contaminates it at the very beginning of the receiving chain, the antenna. In the specific case of GNSS receivers, multipaths remain one of the most difficult disturbance to mitigate. Indeed, as the multipath is of the same nature as the signal of interest it could be barely discernible from it. This similarity between the original signal and its disruptive replica can induce a large positioning error (Kos et al., 2010). This problem explains the large number of research activities which have been led on multipath detection, estimation and mitigation. Conventional signal processing methods have been extensively studied. In the statistical approach,

the narrow correlator technique (Van Dierendonck et al., 1992), the early-late-slope technique (Townsend & Fenton, 1994), the strobe correlator (Garin & Rousseau, 1997), the double-delta correlator (McGraw & Braasch, 1999) and the multipath insensitive delay lock loop (Jardak et al., 2011) methods have raised the interest of the GNSS community, mainly for their simplicity despite their mixed efficiency. The Bayesian strategy was also explored, the Multipath Estimating Delay Lock Loop (MEDLL) remaining for years the reference implementation of the maximum likelihood principle (Townsend et al., 1995), but with multiple variants proposed subsequently like in (Sahmoudi & Amin, 2008) or (Blanco-Delgado & Nunes, 2012). The recent application of particle filtering to multipath mitigation presented in (Qin et al., 2019) makes up for error accumulation of the MEDLL algorithm, at the expense of a significantly higher computational complexity. It is worth noting that if the aforementioned methods are predominantly time-based, some research works have also investigated the frequency domain, through the Fourier transform (Zhang & Bartone, 2004a) or the wavelet decomposition (Zhang & Bartone, 2004b). Nevertheless, these methods may damage the signal of interest, particularly in cases where the spectrum of the multipath is too close from the one of the direct path. The use of Machine Learning (ML) techniques to mitigate the errors in GNSS signals has gained some interest in the early 2000s. A Multi-Layer Perceptron (MLP) architecture designed to mitigate multipath error for Low Earth Orbit (LEO) satellites has been detailed in (Vigneau et al., 2006) for example. More recently, taking advantage of the progress in kernel methods, (Phan et al., 2013) proposed a support vector regressor using signal geometrical features to mitigate multipath on ground fixed Global Positioning System (GPS) stations. Still with Support Vector Machines (SVM), (Hsu, 2017) has conducted multipath detection using high-level products of the GNSS receiver positioning unit. A comparison of the performances of SVM and Neural Networks (NN) algorithms to detect Non Line-Of-Sight (NLOS) multipath is exposed in (Suzuki & Amano, 2021), using native GNSS signal processing outputs as features. Unsupervised ML algorithms, like K-means clustering, have also been used with some success, for instance in (Savas & Dovis, 2019). However, the latest and significant advances in Artificial Intelligence (AI), and notably in Deep Learning (DL), have opened up new perspectives. In (Quan et al., 2018), using a CNN, a carrier-phase multipath detection model is developed. The authors propose to extract feature maps from multi-variable time series at the output of the signal processing stage using 1-Dimensional (1-D) convolutional layers. DL spoofing attack detection in GNSS systems was addressed in the research literature (Schmidt et al., 2020) as well. Lately, (Tao et al., 2021) has proposed a combined CNN-Long Short-Term Memory (LSTM) real-time approach, also based on 1-D convolution, and (Kong et al., 2022) and (Blais et al., 2022) have introduced the use of 2-Dimensional (2-D) signal-as-image representation. (Kong et al., 2022) makes use of MLP input layers for automatic features construction whereas (Blais et al., 2022) processes the images by 2-D convolutional filters. A review of the recent applications of ML in GNSS can also be found in (Siemuri et al., 2021), focusing on use cases relevant to the GNSS community.

However, the multipath difficulty is still a challenge in GNSS signal processing, especially concerning the most ambitious task, the multipath removal. This ultimate solution requires an accurate and precise estimation of the multipath characteristics beforehand. To be more specific, the multipath can be completely modeled by mean of four parameters, its delay, attenuation, frequency and phase. Therefore, this research work focuses on the estimation of these multipath parameters through an original CNN regression method.

Proposed methodology:

Even though our method is illustrated on GNSS data, it applies to a wide range of applications with grid-like data where robustness needs to be enforced due to input or label uncertainties for example. The methodology we propose can be decomposed into several steps. First, for the quantity of interest to be estimated using deep regression, it is necessary to predefine a distribution of the uncertainty of its labels. In many cases, a Gaussian distribution will work but it is important to take into account the physical nature of the quantity of interest as it is discussed in Section 2.2. Next, distributional labels are constructed using the predefined distributions and converted into histograms by choosing the bin size hyperparameter. Using the distributional labels and by minimizing a Histogram Loss function defined also in Section 2.2, the neural network regressor with distributional outputs is trained. The distributional outputs represent the predicted histograms where each softmax output estimates the probability of a bin. To retrieve the scalar prediction of the quantity of interest, the expectation of the whole predicted histogram is computed. The technique enforces robustness by taking into account all possible realizations of the targets instead of using a single realization. The method is then illustrated on the GNSS multipath data. For each multipath parameter, we train such distributional neural network regressor using predefined label uncertainty distribution. The truncated Gaussian distribution is selected for all multipath parameters except for the phase parameter for which it is shown that the Von Mises distribution is a better choice due to its periodic nature.

Contributions of this study:

- To the best of our knowledge, this is the first model combining CNN regression and distributional loss optimization that has been proposed.
- Among the growing literature on multipath detection using ML techniques, this study is one of the few research investigations that has been studying the application of modern DL methods to GNSS multipath parameter estimation. In particular, the investigation of distributional loss on image representations of GNSS correlation signals had not been carried out before.
- The proposed technique naturally applies to cross channels data. Indeed, the regression model investigated here works across input channels, using

information from several distinct channels to carry out estimation. In that sense, the model is also different from models that learn from Red Green Blue (RGB) images that use replicas of the same input image under various color channels.

- To the best of our knowledge, this study is also the first end-to-end real life application employing the technique of distributional loss as a soft labelling mechanism for regression on images.
- Finally, numerical experiments show clear evidence of robustness and boosted performance with the distributional regression loss model when compared to basic CNN regression on the GNSS multipath application.

This article is organized as follows. Section 2 details the concept of CNN regression and distributional loss. Section 3 describes the GNSS multipath parameter estimation problem, details the experiments that are conducted and provides numerical results. Section 4 concludes the article.

2. CNN Regression

In this section, the principles of the classical deep regression are first recalled. The classical deep regression will be considered as the baseline for further comparisons. In the first sub-section, the theoretical developments of the soft labelling technique and the distributional loss mechanism are presented. The second sub-section describes the general specification of the neural network architecture used in this study.

2.1. Baseline CNN-Regression

Regression methods (Hastie et al., 2001) have been extensively studied and applied to many kinds of engineering problems. If data are not too complex, linear regression methods may be used. Alternatively, when direct linear models are not applicable, basis expansion may also be used in order to carry out linear regression on features instead of raw data. Mathematically, considering input vectors $x \in \mathcal{X}$, one would like to express a response $y \in \mathcal{Y}$ as a linear function of features $\phi_\lambda(x) \in \mathcal{H}$ as $y = w^\top \phi_\lambda(x)$ where w are the regression weights and λ represents the parameters needed to represent x by its features in $\mathcal{H}$. There are several techniques to construct the features $\phi_\lambda(x)$ such as basis function expansion to carry out polynomial regression, kernel methods for SVM (Murphy, 2012) - in theses two cases λ may represent degrees of polynomials or parameters of kernel functions respectively - or also NN representations.

In this study, we are interested in the specific case where $\mathcal{X}$ is a subspace of images that can be represented in $\mathbb{R}^{r1 \times r2 \times n_C}$ where $r1 \times r2$ is the resolution of the images and n_C is the number of channels (ex: $n_C = 3$ for RGB images or $n_C = 2$ for GNSS correlation images on I and Q channels in our study). For such input objects, it has been shown empirically that the construction of features $\phi_\lambda(x)$ using convolutional filters achieves the best feature representation for

image-like inputs (Lecun et al., 1998). This is the reason why several authors have proposed convolutional network regression techniques to estimate various information from images (see (Lathuilière et al., 2020) for a review of such techniques).

CNN are composed of several layers. Multiple convolutional blocks are used in order to extract local features in the images using a variety of parameterized convolutional filters (see (Goodfellow et al., 2016) for a detailed explanation of convolutional mechanisms). Successive convolutional layers combined with activation layers (usually rectified unit activation layers) capture multiscale features in the images. Stacking several convolutional blocks that are separated by max-pooling layers to reduce dimension will allow the extraction of highly complex features $\phi_\lambda(x)$ where $\lambda = (W^l, b^l)_{l=1,\dots,L}$, W^l are the filter weights tensors and b^l are the bias vectors for the L convolutional layers. These features are then fed to conventional dense layers that are fully connected with connection weights and compute the regression model $y = w^\top \phi_\lambda(x)$. There are various architectures that can be designed for such purposes. Among these, Visual Geometry Group (VGG)-like networks (Simonyan & Zisserman, 2015) are often used since they have proven to be very effective in practice. The idea is to increase the number of convolutional filters when going deeper in the architecture. See Figure 1 for an example of a basic VGG architecture. Other architectures have been proposed such as ResNet (He et al., 2016), InceptionV3 (Szegedy et al., 2016) which in some contexts may outperform the VGG model. However, our approach in this study is to emphasize the methodological aspects of the proposed technique rather than detail a fully optimized neural architecture framework. For this reason, we have chosen to use a baseline neural network architecture (the VGG architecture) that is commonly accepted as efficient in a wide range of applications on images. Using alternative architectures, the proposed method would be exactly identical as the one detailed in this article. The model performance could even achieve better results as an optimal neural architecture could be found for the application at hand. However, this would not emphasize the role of the distributional loss regression approach but rather the optimization of the architecture. The choice of architecture is therefore secondary and this is why in the sequel of the article, we will only consider VGG-like neural networks.

The CNN regression task on complex input data such as images is intrinsically difficult. The training of deep representations of features combined with the training of a dense network on a large flatten feature representation are required. Sometimes, the image information is also spread across several input channels as we will see later when dealing with I and Q GNSS correlation images (see section 3). Therefore, the regression model has to construct complex features representations from the various channels, adding even more complexity to the overall regression process. In order to ease the training of such multi-channel regression, in the next section, we will propose to use a soft label procedure. The main idea is to learn soft labels rather than sharp target continuous values.

Note that in the following, the response of the complete neural network regression model will be written as $y = N_\theta(x)$ where $\theta = (\lambda, W)$ and λ rep-

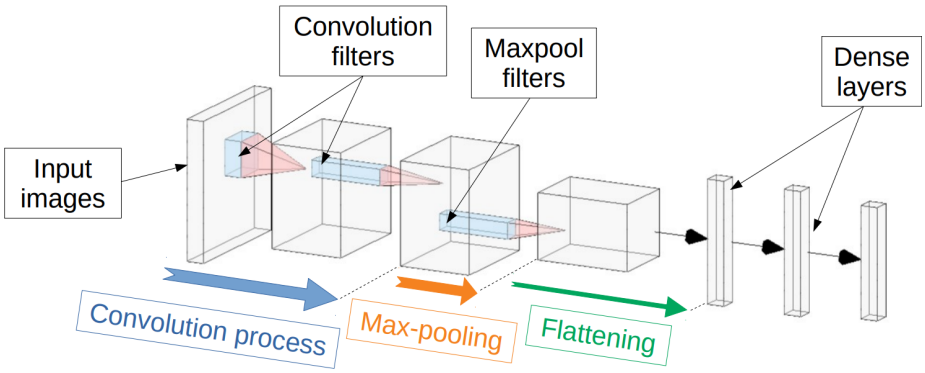

Figure 1: Image convolutional processing inside a CNN. Image generated with (LeNail, 2019).

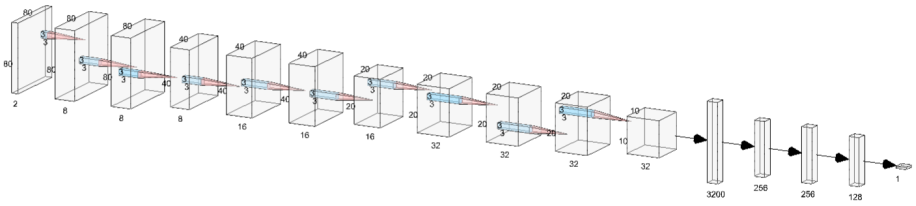

Figure 2: CNN architecture used for regression task on 80x80 images. Image generated with (LeNail, 2019).

resent filter and connection weights from the CNN network architecture up to the penultimate dense layer and W is the weight matrix storing all connection weights from the last dense layers.

2.2. Soft labelling using distributional loss

In this section, we introduce the concept of distributional loss as proposed in (Imani & White, 2018). Traditionally, given an input x, the regression task consists in computing the parameter θ of a regression model N_θ that would assign predicted values $N_\theta(x)$ for given inputs x and targets y. The target values y can be seen as expected values of an underlying distribution of $Y|x$ assumed to be Gaussian. Therefore, in this setting, it is natural to calibrate the parameter θ by minimizing a square loss function $(N_\theta(x) - y)^2$.
The target value y is usually considered as the ground truth and obtained by experiments or measurements. However, it may be subject to uncertainty or ambiguity (Gao et al., 2017) and may impair the generalization performance of the regression model. The main idea of soft labelling is to consider that the target value is an observation of an underlying ground truth distribution and one may benefit from learning the distribution rather than the individual targets.

When dealing with label uncertainties in deep learning, the simplest and most common approaches are using techniques that improve the generalization performance of the model. This can be achieved using loss regularization or dropout (Goodfellow et al., 2016). While they are easy to implement and usually already available in deep learning software packages, these methods do not explicitly address the underlying uncertainty that may be present in the labels. They rather control the statistical bias-variance trade-off of the data. Investigations have also been conducted using partial labelling (Cour et al., 2011) or co-teaching strategies (Han et al., 2018). These last two methods are more focused on uncertainty but require the availability of "clean" labels (labels not subject to uncertainty) for some of the data which is often difficult to ensure in real life applications. Deep quantile regression (Shen et al., 2021) is an alternative to these techniques. Quantile regression has been well studied in the case of linear models. The concept has been extended to nonlinear models and deep regression models. The idea is to estimate a quantile rather than a specific value. In a way, it is similar to the method we propose. However, our model extracts the complete distribution while quantile regression models only estimate a specific quantile. Additionally, the distribution loss as we construct using the KL divergence has nice mathematical properties (its Lipschitz constant is low when compare to classical regression) and greatly helps during the training as shown in (Imani & White, 2018). The loss also embeds the prior distribution of the target and allows the use of distributions that model specific physical behaviors from the application such as the cyclic nature of the phase parameter in the GNSS application. The models constructed by the other methods discussed above would not integrate this specific behavior and would require that the dataset contains many representations of the phenomenon so that the model is able to learn it. In practice, the distribution of input samples does not contain sufficient representation of data to observe and train on these specific physical behaviors. These are the main reasons why the distribution loss approach was chosen over other methods. Next, we detail its mathematical model.

Consider now the task of learning the distribution $Y|x$ instead of predicting directly $\mathbb{E}[Y|x]$. For each target value, a target distribution $Y|x$ is chosen. This choice is of course problem dependent. However in (Imani & White, 2018), it has been shown experimentally, when no prior knowledge on the target distribution is known, that the use of a truncated Gaussian is the best choice among a variety of other distributions. In that case, we will assume that $Y|x \sim \mathcal{N}_{[a,b]}(\mu, \sigma^2)$ where $\mathcal{N}_{[a,b]}(\mu, \sigma^2)$ is the truncated Gaussian distribution of mean μ and variance σ^2 on the interval $[a, b]$ with density

$$f_{\mu,\sigma,[a,b]}(y) = \frac{1}{\sigma} \frac{\varphi(\frac{y-\mu}{\sigma})}{\phi(\frac{b-\mu}{\sigma}) - \phi(\frac{a-\mu}{\sigma})} \mathbb{1}_{[a,b]}(y)$$

where φ and ϕ are the density and the cumulative distribution functions of the standard normal distribution $\mathcal{N}(0, 1)$.

In some specific cases, the use of a Gaussian distribution as a prior may not be adequate. For example, when data are defined modulo some period

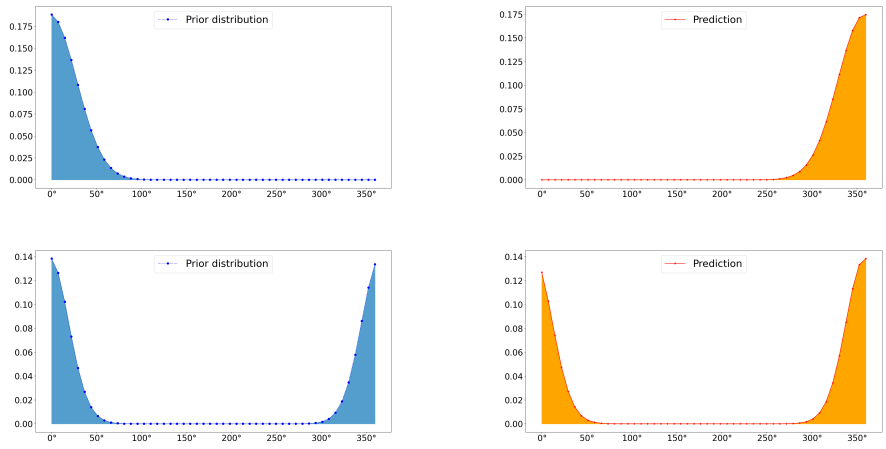

Figure 3: The truncated Gaussian distribution (top) and the Von Mises circular distribution (bottom) for angles of 2°and 355°. The Von Mises circular distribution displays similar values near 0 and 2π whereas the truncated Gaussian distribution shows very different probabilities around these two values.

when dealing with angles, a circular distribution is more adapted. Later, in the application to GNSS data, the phase of the signal is one example of such circular information defined on the interval $[0, 2\pi]$. Small values of the phase should be considered very similar to values around 2π. The use of a Von Mises distribution rather than a truncated Gaussian distribution will conserve the symmetry around zero when the Gaussian distribution would consider phases values around 2π large when compared to small phases. See Figure 3 for an illustration of the two distributions.

The Von Mises density function is defined as follows: for an angle $\theta \in [0, 2\pi]$,

$$f_{\kappa,\mu}(\theta) = \frac{1}{2\pi I_0(\kappa)} e^{\kappa \cos(\theta - \mu)}$$

where I_0 is the modified Bessel function of the first kind of order 0. The parameter μ and $1/\kappa$ play the same role as the mean and variance in the case of the Gaussian distribution.

In order to learn the target distributions, a discrete version of $Y|x$ is considered in the form of an histogram with K bins. To represent a discrete distribution at the output of a neural network, we will assume that is has a compact support. When this assumption is not met, the distribution can be truncated, implicitly considering that the target value does not follow an heavy tail distribution around its mean value. For some distributions, the truncation is rather simple (ex: Gaussian). However, when the distribution does not have a cumulative probability function with a close analytical form (ex: the Von Mises distribution), numerical integration is required to integrate the proba-

bility distribution function and calculate the probability mass of each bin of the histogram. Such computation may be expensive when the number of bins is high. During training, as we are now comparing the estimated distribution represented by the softmax output of the network $N_\theta(x)$ and the target distribution $f_{\mu,\sigma,[a,b]}$, we can use the KL divergence as a loss function. Considering distributions with density functions p and q, it is defined as follows

$$\mathcal{D}_{KL}(p||q) = \int_{-\infty}^{+\infty} p(y) \log \frac{p(y)}{q(y)} dy.$$

It measures how different q is from p. If p is the target and q the estimated density, using $\mathcal{D}_{KL}$ as a loss function will require to minimize simply the following term also known as the cross-entropy

$$h(p, q) = -\int_{-\infty}^{+\infty} p(y) \log q(y) dy.$$

When p is the density of the truncated Gaussian distribution $f_{\mu,\sigma,[a,b]}$, the cross-entropy can be written as

$$h(f_{\mu,\sigma,[a,b]}, q) = -\int_{a}^{b} f_{\mu,\sigma,[a,b]}(y) \log q(y) dy.$$

For each input x, the estimated discrete distribution q is a K-dimensional vector where each component i returns a probability $q_i = (N_\theta(x))_i$ that the target value is in the bin i. The number of bins K will be an hyper-parameter of the method. The discrete target distribution p corresponding to $f_{\mu,\sigma,[a,b]}$ can also be seen as a K-dimensional vector where each component i returns the probability mass $p_i = F(a_i+t_i)-F(a_i)$ contained in the i-th bin, where a_i and t_i are the left bound and the width of the bin respectively and F is the cumulative distribution of $f_{\mu,\sigma,[a,b]}$. Therefore, for a given input x the distributional Histogram Loss (HL) can be written as follows

$$HL(x) = -\sum_{i=1}^{K} p_i \log(N_\theta(x))_i$$

$$HL(x) = -\sum_{i=1}^{K} [F(a_i + t_i) - F(a_i)] \log(N_\theta(x))_i$$

After training, for a given input x, $\mathbb{E}[N_\theta(x)]$ will provide the estimated value of $\mathbb{E}[Y|x]$. Note that this method is distinct from transforming the regression task into a classification task by discretizing uniformly the support set of target values. Here, a target distribution is assumed and learned during the process. Additionally, by tuning the variance parameter σ and the number of bins K used to transform the target distribution into an histogram, it is possible to adjust the bias-variance trade-off achieved by the model. A large σ would enforce robustness while a small value would tend to reduce the method to a classical regression. However, too large values of σ could affect negatively the estimation accuracy. A trade-off has then to be found.

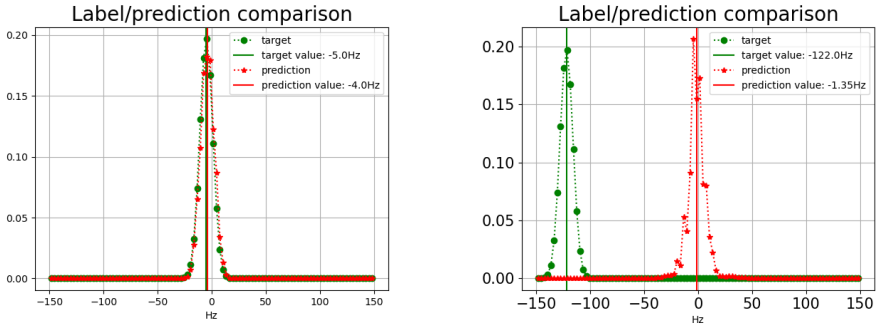

Figure 4: Comparison between the target and the predicted distributions for a correct (left) and wrong (right) prediction.

In Figure 4 (left) and Figure 4 (right), a comparison between the target and the predicted distributions are shown. The two figures correspond to a correct and a wrong estimation respectively. Each point of the *target* curves represents the target probability mass p_i while each point of the *prediction* curves is the probability estimate q_i. It can be noticed that in the left figure, both curves are nearly superposed and both form a near Gaussian distribution. In the opposite, in the right figure, the target and predicted probability distribution do not coincide. This situation corresponds to a wrong estimation. In this case, the Gaussian behaviour is not so well reconstructed.

Algorithm 1 summarizes the complete training procedure with distributional loss.

Algorithm 1: Robust CNN regression training with distributional loss

input : $D = (X, T)$ dataset of inputs X and targets T
input : prior distribution of $Y(t)|x$ for all pairs $(x, t) \in D$
input : K number of bins
output: θ^* the optimal neural parameters

1. *Construct* $\mathcal{H}_t^K = \{p_i \in [0,1] \ , \ i = 1, \ldots, K\}$ *for all* $t \in T$ *where* p_i *is the probability mass in bin* i *for* $Y(t)|x$;
2. *Construct the neural function* $N_\theta : \dim(X) \to K$;
3. *Train* N_θ *by solving:*

$$\theta^* = \underset{\theta}{\operatorname{argmin}} \left\{ - \sum_{(x,t) \in D} \left(\sum_{p_i \in \mathcal{H}_t^K} p_i \log(N_\theta(x))_i \right) \right\}$$

2.3. Dedicated CNN architectures

In this study, in order to compare the HL strategy with the classical CNN regression , two VGG-like architectures (Simonyan & Zisserman, 2015) will be used. Sharing the same backbone as shown in Figure 2, they only differ in their outputs. The extraction part is composed by 3 convolution blocks, each formed by consecutive convolutional layers and ended by a max-pooling layer. The number of filters is increasing with the depth of the convolutional layers. At the end of the convolutional process, the data are flattened to feed the 3 hidden dense layers. The output layer consists in a single neuron equipped with a linear activation function for the classical baseline CNN and a softmax layer with K outputs in the soft labelling case.

To differentiate the two algorithms, we will refer to the network with a single output as CNN-Regression (CNN-Reg) and the one using the softmax output as CNN-Histogram Loss (CNN-HL). Furthermore, the CNN-HL implementing the truncated Gaussian distribution will be noted the CNN-Histogram Loss Truncated Gaussian (CNN-HL TG) while the one using the Von Mises distribution will be written CNN-Histogram Loss Von Mises (CNN-HL VM).

3. Application to GNSS multipath parameter estimation

This section presents the GNSS multipath parameter estimation application considered as a use case for the deep regression with distributional loss. First, the problem of estimating multipath parameters in GNSS signals is introduced. The setup of our experiments is then detailed. Next, we present and analyze the experimental results. Finally, we discuss our findings.

3.1. Problem statement

The GNSS positioning principle is based on the distance measurement between satellites of known positions and the receiver to locate. Using the propagation time of a dedicated signal emitted by the satellite, the receiver estimates its relative distance to the satellite (Tsui, 2005). Using several distances, the receiver is able to calculate its position by trilateration.

More precisely, the calculation process of the Position Time Velocity (PVT) solution relies on the synchronisation between the received signal and a receiver replica of the wanted signal. The alignment between both signals requires the estimation of the three unknown parameters of the incoming signal which are:

- The propagation delay τ,
- The Doppler shift frequency f_D,
- The carrier phase ϕ.

In a classical receiver, this estimation process is typically conducted by mean of the maximum likelihood principle. It is implemented through the maximization of the cross-correlation between the received signal and a local replica signal

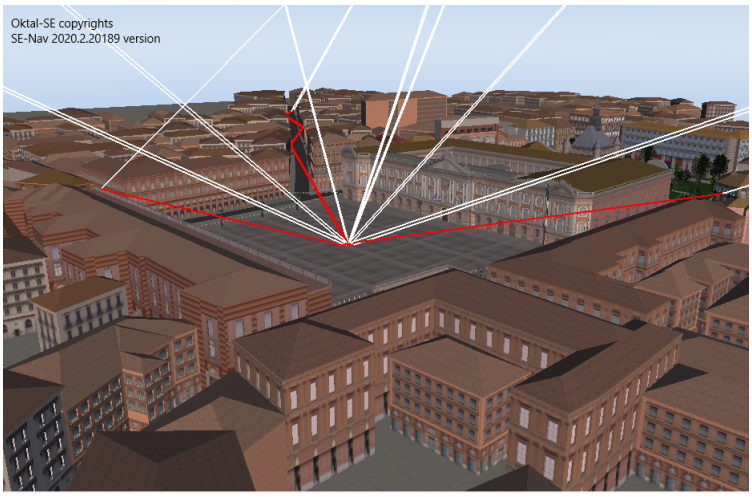

Figure 5: GNSS receiver collecting LOS and NLOS signals in Toulouse city center (France).

parameterized by three test values $\tilde{\tau}$, $\tilde{f_D}$ and $\tilde{\phi}$. In practice, the correlation operation is accomplished through a product followed by an integrate and dump stage. To be complete, it should also be pointed out that correlation is split into two orthogonal channels, named by convention In phase (I) and in Quadrature (Q). The maximization task is generally carried out by loop systems, namely the Delay-Locked Loop (DLL) and the Phase-Locked Loop (PLL). In addition, the loops present the intrinsic advantage to track the target parameters which are time-varying due to the continuously changing receiver-satellites geometry. No more than three correlation points per I and Q channels are usually required for normal loop operation.

However, various kinds of interference can deteriorate the positioning process. multipath is one of the most common and the most harmful interference. A multipath is a reflection on a surrounding obstacle of the useful signal picked up by the receiving antenna concurrently to the LOS signal. An illustration of the phenomenon is shown in Figure 5, generated by the SE-Nav software (Oktal-SE, 2020), a signal propagation simulator. In general, a receiver is impacted by multiple multipaths, especially in urban environments where reflectors are numerous. Sometimes, the direct path may even be absent due to an obstruction, for example when high buildings are surrounding the receiver (Ziedan, 2018). However, in this study the assumption is made that the direct path is always present and a single multipath will be considered. Being the replica of the signal of interest, the multipath contains the same information but with shifted parameters:

- The code delay in excess compared with the useful signal $\Delta\tau_{MP}$,
- The difference in Doppler shift frequency with the useful signal Δf_{MP},

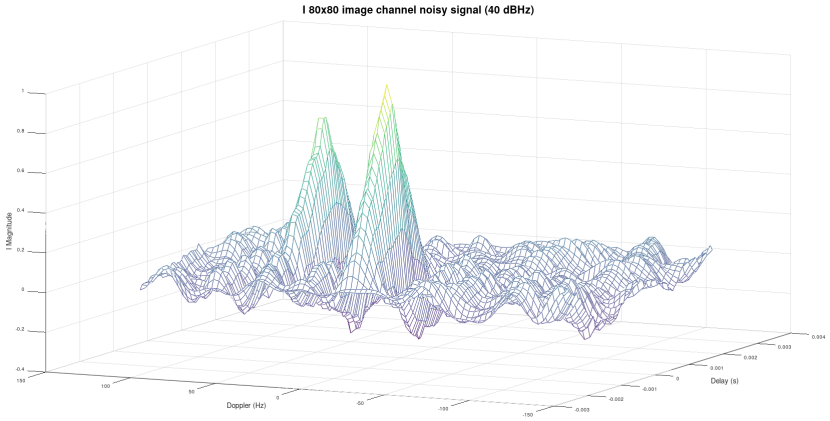

Figure 6: I channel for a signal undergoing a multipath with $\tau_{MP} = 0.75$ Tc[1], $f_{MP} = 100$ Hz, $\alpha_{MP} = 0.8$, $\phi_{MP} = 45°$.

- The phase difference with the useful signal $\Delta\phi_{MP}$,
- The attenuation of the multipath with respect to the magnitude of the useful signal α_{MP}.

The addition of the multipath to the direct signal biases the result of the correlation operation. As a consequence, the estimation of the incoming signal parameters is altered and the accuracy of the position delivered to the user may be degraded. With as few as three correlation points per channel, the information available to detect the multipath contamination and possibly mitigate its effects is poor.

This work proposes to use a larger number of correlation points in order to overcome this lack of information. Indeed, $\tilde{\tau}$ and $\tilde{f_D}$ each sample a specific range, forming a 2-D grid. The correlation outputs in turn compose a 2-D matrix, in other words an image. The signal having 2 channels, I and Q, there will be two 2D-images at the output of the correlators. The general process is depicted in Figure 8. Figures 6 and 7 are given as detailed examples of the resulting images. Concerning the estimation of ϕ, the necessary information is available through the orthogonality property of the I and Q channels, the first one granting access to $\cos(\phi)$ and the second to $\sin(\phi)$. The image construction is detailed in (Blais et al., 2022).

3.2. Experimental setup

[1]The results presented in this paper where established using the GPS L1 C/A legacy signal. However, the authors are confident that they could be generalized to other navigation signals, with the same overall structure, as no specific assumption has been made. Tc is the chip period, a basic defining parameter of this type of signal. Tc = 1/1023 ms for the GPS L1 C/A signal.

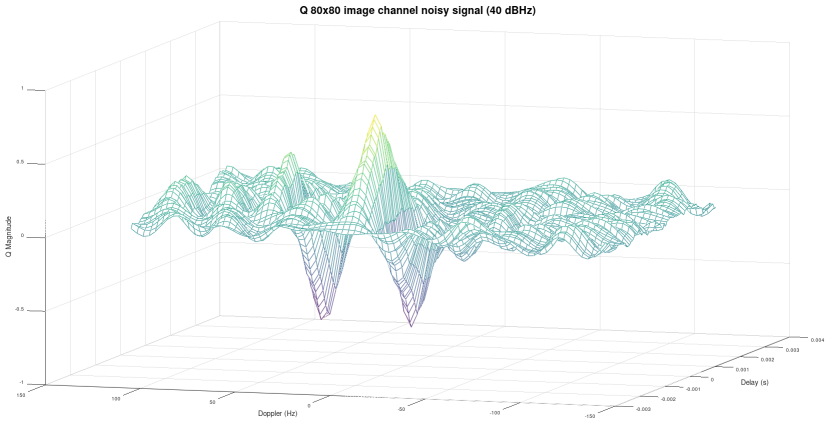

Figure 7: Q channel for a signal undergoing a multipath with $\tau_{MP} = 0.75$ Tc, $f_{MP} = 100$ Hz, $\alpha_{MP} = 0.8$, $\phi_{MP} = 45°$.

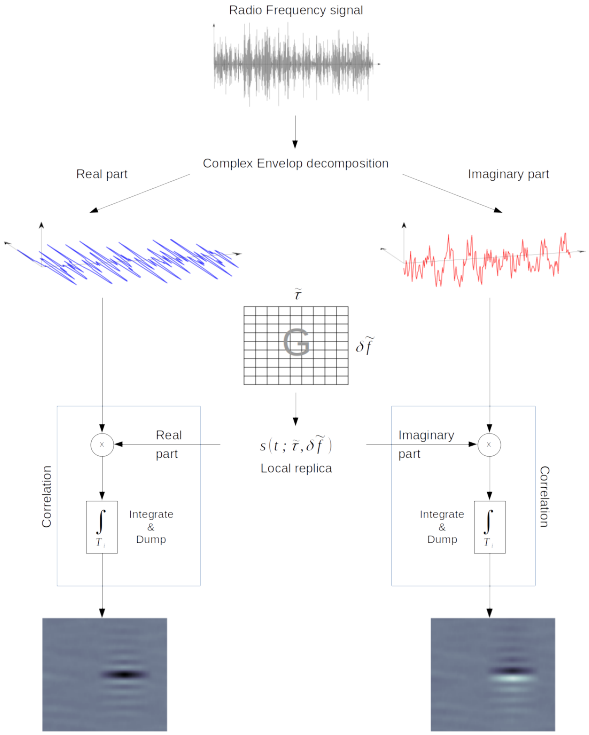

Figure 8: Illustration of the image elaboration process. The received signal is split in two orthogonal components. Next, each component is correlated with a local replica signal whose parameters span a grid G. The result forms a 2D-image. The pair of images then supplies the CNN implementing the regression task. The tilde notation indicates the local parameter by opposition to the received signal unknown parameter.

This sub-section describes the datasets we have used and the hyper-parameters that we have selected to conduct the experiments.

3.2.1. Dataset definition

Experiments were made on several datasets which are divided into 2 groups. The first group is characterized by a fixed Carrier to Noise *C/N0* ratio while the second has distinct *C/N0* ratio levels. The *C/N0* figure represents the ratio between the power of the signal of interest and the receiver intrinsic noise level. In the GNSS community it is considered as one of the most important figure of merit for the quality of the received signal. When the ratio is high, more precise downstream parameter estimation is expected.

In the first group, *C/N0* is equal to 40 dBHz to simulate a typical urban environment receiving condition. The multipath attenuation α_{MP} is categorized to assess the performance of both CNN algorithms as a function of this parameter. In this way, we have:

- Strong multipaths (SMP) where $\alpha_{MP} \sim \mathcal{U}([0.6, 0.9])$,
- Moderate multipaths (MMP) where $\alpha_{MP} \sim \mathcal{U}([0.4, 0.6])$,
- Weak multipaths (WMP) where $\alpha_{MP} \sim \mathcal{U}([0.1, 0.4])$,
- Random multipaths (RMP) where $\alpha_{MP} \in [0.1, 0.9]$ are selected randomly and uniformly in SMP, MMP and WMP.

The second group is composed of datasets with different *C/N0* ratio while $\alpha_{MP} \sim \mathcal{U}([0.1, 0.9])$. The goal is to evaluate the algorithms with respect to *C/N0*, which takes the following values: 43 dBHz, 40 dBHz, 37 dBHz and 34 dBHz. Moreover, all datasets share the same following setup:

- $\Delta\tau_{MP}$ is uniformly generated in $[0, 1.5] \times$ Tc,
- Δf_{MP} follows a truncated centered Gaussian distribution with standard deviation $\sigma = 125/3$ Hz. Δf_{MP} is restricted to $[-3\sigma, +3\sigma]$ Hz,
- $\Delta\phi_{MP}$ is uniformly generated in $[0, 2\pi]$ rad.

Each dataset contains 10000 samples. Each sample is composed of a I and Q image pair. Within a dataset, the image size does not change. However, to observe the CNN parameter estimation performance under various image resolutions, multiple datasets were generated: 80x80, 40x40, 20x20 and 10x10 pixels image resolutions.

A synthetic correlator output generator (Blais et al., 2021) has been used to populate the datasets used in this study. The underlying signal model and the generation process are completely described in (Blais et al., 2022). The generator is fully configurable with respect to the multipath parameters *C/N0*, α_{MP}, $\Delta\tau_{MP}$, Δf_{MP} and $\Delta\phi_{MP}$. The resolution of the I & Q output images can also be set on demand.

3.2.2. Hyper-parameter selection

Performance results provided in the next section are results averaged over 10 runs. For each run, the training set is formed by randomly selecting 8000 samples and the validation set by selecting randomly 1000 samples among the 2000 remaining samples. The last 1000 samples constitute the test set.

CNN-Reg and CNN-HL were trained for 100 epochs and have used a 1000 samples batch size. Learning rates of both CNN were empirically fixed to 10^{-3} and 3x3 filters were used for convolutional layers as shown in Figure 2.

The prior distribution for CNN-HL is taken as the truncated Gaussian distribution. In addition, for the phase estimation,experiences are conducted for both truncated Gaussian and Von Mises prior distributions. Comparison between the two will be discussed.

The choice of the specific values for the hyper-parameters defining the prior distributions is as follows. Regarding the support of the distribution $[a, b]$ it is fixed in accordance with the range of variation of the parameter as defined in Section 3.2.1 and further detailed in (Blais et al., 2022). The number K of histogram bins is fixed in order to obtain a finely sampled representation of the prior distribution. A bin size $w = (b-a)/K$ of two order of magnitude lower than the support of the distribution has been selected leading to $K = 100$. In practice for our application this value is a good trade-off between a rich representation of the prior distribution and the number of neural network outputs. On one hand, a higher value for K would lead to a large neural network architecture complexifying the training task. On the other hand, a too small value for K would generate a very biased estimation due to large bins. Finally, the standard deviation of the truncated Gaussian distribution σ has been determined as a function of w via grid-search. The ratio $\sigma = 2w = 2(b-a)/K$ was retained as the best trade-off over all parameters and all test sets. It corresponds to a narrow Gaussian shape with respect to the support $[a, b]$. The parameter κ plays a similar role as $1/\sigma$ and its value has been deduced from σ.

3.3. Results

Performance assessment is based on Mean Absolute Error (MAE) results averaged over 10 runs. For the CNN-Reg, the MAE metric is as follows:

$$\frac{1}{L}\sum_{l=1}^{L}|N_\theta(x_l) - y_l|$$

where L is the number of samples in the test dataset, x_l is the l-th test sample and y_l is the target value for the l-th sample.

For CNN-HL, the MAE metric is calculated as follows:

$$\frac{1}{L}\sum_{l=1}^{L}\left|\tilde{\mathbb{E}}[N_\theta(x)] - y_l\right|$$

with

$$\tilde{\mathbb{E}}[N_\theta(x)] = \sum_{i=1}^{K} q_{i,x_l}.c_i$$

where q_{i,x_l} is the probability of the i-th softmax output neuron associated to x_l and c_i is the center of the i-th bin for $i \in [1, K]$ (when the interval $[a, b]$ has been partitioned into K equal subdivisions).

Tables 1, 2, 3 and 4 gather the averaged MAE performance results as defined above for the two algorithms CNN-Reg and CNN-HL on the first group of datasets.

Figures 9, 10, 11 and 12, provide results for the second group of datasets and illustrate the CNN-Reg and CNN-Reg MAE behaviour when *C/N0* varies. The curves are plotted according to a decreasing *C/N0* because a high *C/N0* corresponds to better receiving conditions. The scale on the x-axis starts from 43 dBHz (good receiving conditions) and goes down to 34 dBHz (deteriorated receiving conditions).

In Table 1, it can be observed that in all conditions and datasets, CNN-HL always shows better results than CNN-Reg. The delay τ_{MP} estimation performance is higher with CNN-HL on 20x20 images than the performance of CNN-Reg on 80x80 images. Additionally, as expected the average MAE and its standard deviation increase as the image resolution diminishes.

In Table 2, similar results are observed for strong multipaths. For moderate and weak multipaths, the Doppler frequency estimation performance remains better on 40x40 images with CNN-HL than the performance of CNN-Reg on 80x80 images.

In Tables 3 and 4, similar behaviour as τ_{MP} estimation are found. It can be noticed that in the particular case of α_{MP} estimation, the average MAE decreases as the strength of the multipath decreases. This could be explained by the fact that in strong, moderate and weak multipath datasets, the α_{MP} parameter estimation is biased by the dataset design that only contains ranges of α_{MP} parameters.

In Table 4, ϕ_{MP} estimation with both CNN-Reg and CNN-HL TG shows difficulties. For example, in the case of weak multipaths on 20x20 images the average MAE remains at a level of 16°, which might not be sufficient for practical multipath mitigation use. In contrast, the results of the CNN-HL VM algorithm are always at least 30% higher than those of the CNN-HL TG, for all resolutions down to 20x20. In particular, for an image size of 20x20, the maximum average MAE is less than 10°. It corresponds to a projection error of 2% and 18% on the I and Q channels respectively. Experiences on the RMP dataset emphasize some robustness when the attenuation α_{MP} is drawn from a large range of values, which shall arise in many real life situations.

Figures 9, 10, 11 and 12 show a moderate monotonic degradation of the estimation performances with decreasing *C/N0* for the four parameters of interest τ_{MP}, f_{MP}, α_{MP} and ϕ_{MP}. This observation stands for all resolutions from 80x80 to 20x20 when the CNN-Reg is in use. The trend is the same when CNN-Reg operates at the lowest studied resolution of 10x10, but with a much more significant slope. Indeed, at this resolution MAE values are at least doubled when the *C/N0* decreases from 43 to 34 dBHz. The behaviour of CNN-HL with respect to *C/N0* is the same, although a clear offset in accuracy and precision is always visible between the curves for CNN-Reg and the ones for CNN-HL,

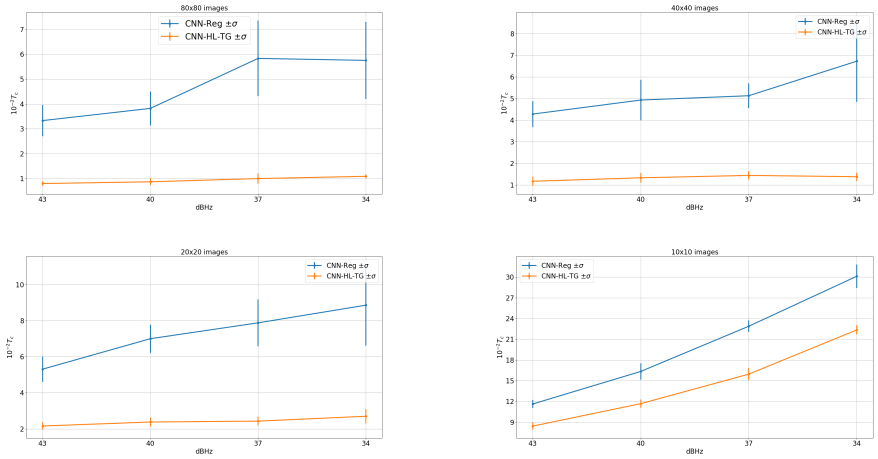

Figure 9: MAE behaviour according to *C/N0* ratio for τ_{MP} estimation.

in favour of the later. This is consistent with the comments made previously exploiting Tables 1, 2, 3 and 4. Regarding the specific case of ϕ_{MP} the improvement brought by CNN-HL VM over CNN-HL TG can be distinctly noticed for all values of *C/N0* with a phase estimation accuracy always increased by at least 2° for all resolutions.

Table 1: MAE in 10^{-2} Tc for τ_{MP} estimation with respect to various multipath attenuations.

Dataset	Image size	Algorithm	
		CNN-Reg	CNN-HL
Strong mulipaths	80×80	2.50 ± 0.22	**0.64** ± 0.18
	40×40	2.91 ± 0.32	**0.76** ± 0.11
	20×20	4.30 ± 0.42	**1.28** ± 0.20
	10×10	6.70 ± 0.40	**4.85** ± 0.17
Moderate multipaths	80×80	3.03 ± 0.54	**0.66** ± 0.04
	40×40	3.44 ± 0.65	**0.96** ± 0.26
	20×20	4.44 ± 0.45	**1.63** ± 0.15
	10×10	9.90 ± 0.72	**7.35** ± 0.38
Weak multipaths	80×80	4.64 ± 1.44	**1.01** ± 0.13
	40×40	4.33 ± 0.42	**1.55** ± 0.20
	20×20	7.20 ± 1.55	**2.86** ± 0.40
	10×10	26.13 ± 1.96	**21.17** ± 0.58
Random multipaths	80×80	3.82 ± 0.68	**0.81** ± 0.11
	40×40	4.92 ± 0.94	**1.20** ± 0.21
	20×20	7.00 ± 0.78	**2.21** ± 0.29
	10×10	16.34 ± 1.19	**12.1** ± 0.59

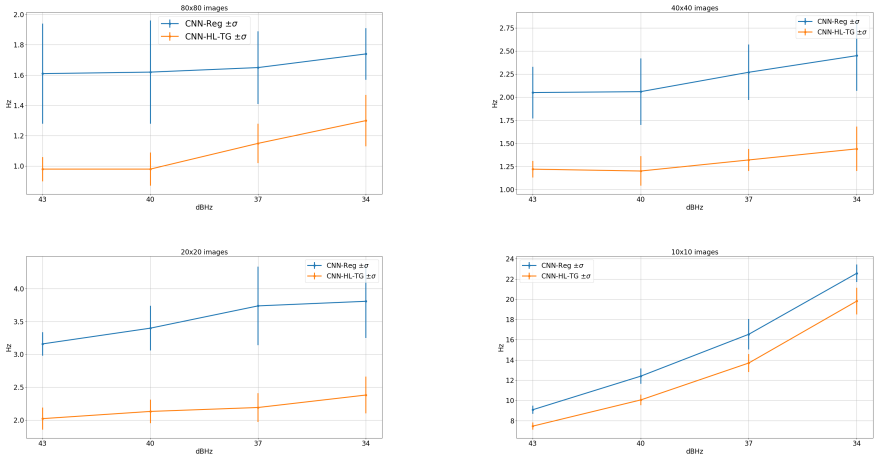

Figure 10: MAE behaviour according to *C/N0* ratio for f_{MP} estimation.

Table 2: MAE in Hz for f_{MP} estimation with respect to different multipath attenuations.

Dataset	Image size	Algorithm CNN-Reg	Algorithm CNN-HL
Strong multipaths	80×80	1.22 ± 0.20	**0.60** ± 0.04
	40×40	1.60 ± 0.27	**0.75** ± 0.07
	20×20	2.36 ± 0.41	**1.17** ± 0.12
	10×10	5.16 ± 0.26	**3.60** ± 0.18
Moderate mutipaths	80×80	1.33 ± 0.29	**0.72** ± 0.06
	40×40	1.70 ± 0.23	**0.87** ± 0.06
	20×20	2.61 ± 0.45	**1.52** ± 0.11
	10×10	7.08 ± 0.36	**5.52** ± 0.27
Weak multipaths	80×80	1.75 ± 0.31	**1.26** ± 0.15
	40×40	2.41 ± 0.38	**1.42** ± 0.13
	20×20	4.30 ± 0.41	**2.84** ± 0.45
	10×10	21.48 ± .68	**19.04** ± 1.07
Random multipaths	80×80	1.62 ± 0.34	**0.95** ± 0.08
	40×40	2.06 ± 0.36	**1.10** ± 0.13
	20×20	3.40 ± 0.34	**2.10** ± 0.28
	10×10	12.40 ± 0.76	**9.71** ± 0.22

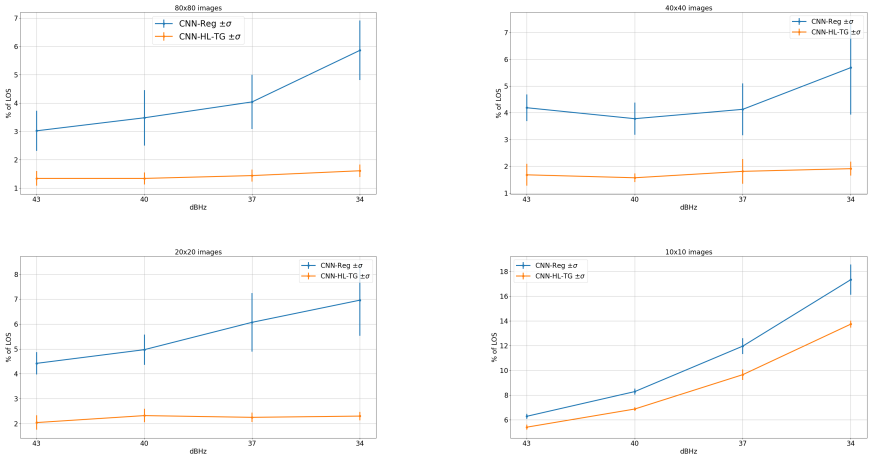

Figure 11: MAE behaviour according to *C/N0* ratio for α_{MP} estimation.

Table 3: MAE in % for α_{MP} estimation with respect to different multipath attenuations.

Dataset	Image size	Algorithm CNN-Reg	Algorithm CNN-HL
Strong multipaths	80×80	3.95 ± 0.93	**1.30** ± 0.26
	40×40	4.45 ± 0.61	**1.48** ± 0.14
	20×20	5.60 ± 0.70	**2.22** ± 0.20
	10×10	7.32 ± 0.40	**5.99** ± 0.25
Moderate multipaths	80×80	3.71 ± 0.64	**1.08** ± 0.05
	40×40	3.76 ± 0.53	**1.22** ± 0.15
	20×20	4.79 ± 0.55	**1.73** ± 0.10
	10×10	5.97 ± 0.16	**5.00** ± 0.07
Weak multipaths	80×80	3.53 ± 0.88	**1.91** ± 1.16
	40×40	4.06 ± 1.28	**1.38** ± 0.57
	20×20	4.09 ± 1.10	**1.66** ± 0.22
	10×10	7.87 ± 0.33	**6.08** ± 0.12
Random multipaths	80×80	3.48 ± 0.98	**1.45** ± 0.19
	40×40	3.78 ± 0.60	**1.60** ± 0.14
	20×20	4.97 ± 0.61	**2.21** ± 0.28
	10×10	8.28 ± 0.24	**6.84** ± 0.12

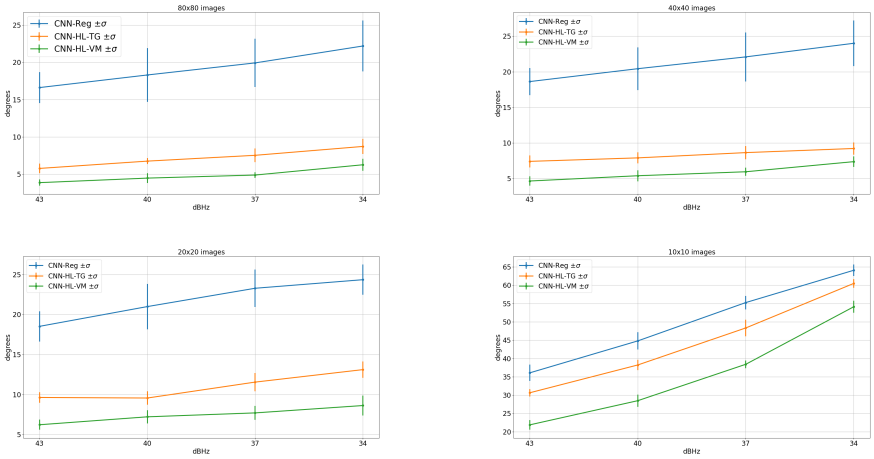

Figure 12: MAE behaviour according to *C/N0* ratio for ϕ_{MP} estimation.

Table 4: MAE in degrees for ϕ_{MP} estimation with respect to different multipath attenuations.

Dataset	Image size	Algorithm CNN-Reg	CNN-HL TG	CNN-HL VM
Strong multipaths	80×80	12.30 ± 1.46	4.26 ± 0.46	**2.21** ± 0.16
	40×40	15.41 ± 3.46	5.06 ± 0.61	**2.62** ± 0.18
	20×20	15.59 ± 2.44	5.68 ± 0.62	**3.84** ± 0.23
	10×10	29.68 ± 2.06	19.88 ± 0.88	**13.16** ± 0.57
Moderate multipaths	80×80	15.71 ± 3.07	5.13 ± 0.72	**3.08** ± 0.35
	40×40	15.72 ± 1.74	5.61 ± 0.32	**3.70** ± 0.41
	20×20	17.90 ± 1.99	7.28 ± 0.64	**4.48** ± 0.48
	10×10	42.55 ± 2.30	29.78 ± 1.44	**19.90** ± 0.85
Weak multipaths	80×80	26.45 ± 6.25	9.29 ± 1.40	**6.46** ± 1.24
	40×40	30.06 ± 3.35	12.26 ± 1.20	**7.26** ± 0.54
	20×20	29.57 ± 3.17	16.27 ± 1.86	**9.58** ± 1.08
	10×10	70.23 ± 1.56	60.53 ± 2.31	**51.53** ± 2.41
Random multipaths	80×80	19.90 ± 3.92	7.02 ± 0.67	**4.47** ± 0.66
	40×40	21.90 ± 3.51	8.26 ± 0.57	**5.39** ± 0.78
	20×20	22.73 ± 3.15	10.00 ± 1.06	**7.21** ± 0.82
	10×10	50.63 ± 2.56	37.85 ± 1.26	**28.48** ± 1.66

3.4. Interpretation and discussion of results

The experiments made on the various datasets show the boosting effect of the CNN-HL algorithm. The model CNN-HL always performs better than CNN-Reg. It always achieves lower average MAE but also a lower standard deviation. This shows that the technique is also more robust across runs. Therefore estimations with the distributional technique are more accurate. The results also

show that better estimation can be achieved on smaller input images. The input size could be reduced from a resolution of 80x80 to 40x40 and sometimes 20x20. The nature of the information we are interested in the correlator output images is characterized by the multipath secondary peak of the correlation signal. The results show that when the images are sub-sampled, the information remains sufficiently present to be extracted by the neural network model. The main reason comes from the fact that the 80x80 images are raw images that contain redundant information. Sub-sampling the images affects the estimation but the estimation performance remains acceptable as long as the resolution remains above a certain level. This is also due to the fact that the information is spread across the two channels I and Q. Combining low resolution patterns from the two I and Q channels is very informative for the model to retrieve multipath parameter estimates. Therefore, to a certain limit, the sub-sampling of images preserves sufficient information. However, when the resolution reaches low levels such as 10x10, the estimation performance is poorer and not sufficient for practical use anymore. Additionally, in the case of the phase parameter, the results also show that the use of the Von Mises distribution as a prior label distribution is adequate and significantly improves the accuracy of the estimation when compared to the use of a truncated Gaussian distribution. Overall, the numerical results clearly demonstrate that the use of label distributions during training enforces robustness in the estimation and confirms the relevance of the proposed distributional loss training model for CNN.

4. Conclusions

Estimating information from images is a useful but difficult task. In this work, we have addressed deep learning regression from multiple images channels. The proposed model makes use of convolutional neural layers in order to extract high level features from images. Instead of carrying out classical regression on these extracted features, a soft labelling approach is used to learn underlying target distributions. The idea is to allow the neural network model to account for some uncertainty in the target and therefore increase its generalization performance. The target distributions are modeled using histograms and a specific histogram loss function based on the KL divergence is applied during training. The resulting neural architecture incorporates a softmax output in order to reconstruct the histogram discrete target probability. The complete process could be applied to any applications that requires inference of function values from multiple images channels. The model is applied to GNSS multipath estimation where multipath signal parameters have to be estimated from correlator output images from the I and Q channels. The multipath signal delay, attenuation, Doppler frequency and phase parameters are estimated from synthetically generated datasets of satellite signals. Experiments are conducted under various receiving conditions and various input images resolutions. For all receiving conditions that have been tested, the proposed soft labelling CNN technique using distributional loss outperforms classical CNN regression. In addition, the gain in accuracy obtained by the model allows downsizing of input

image resolution from 80x80 down to 40x40 or sometimes 20x20. This reduction of image resolution is a first step towards the implementation of such models in physical receivers that are limited in the number of correlator outputs that can be designed. Despite real datasets are difficult to construct and label, further research using the CNN-HL model on GNSS data should focus on data that incorporates real receiving conditions. Additionally, from a model perspective, future investigation should propose adaptive histogram loss techniques that adjust the number of bins and target distribution parameters to the data.

Acknowledgements

This project has been funded by Oktal-Synthetic Environment under the ANRT PhD program.

References

Blais, A., Couellan, N., & Munin, E. (2022). A novel image representation of gnss correlation for deep learning multipath detection. *Array*, *14*, 100–167.

Blais, A., Munin, E., & Couellan, N. (2021). A synthetic GNSS correlator output generator. URL: `https://github.com/AntoineBlaisENAC/Synthetic_GNSS_Correlator_Output_Generator.git`.

Blanco-Delgado, N., & Nunes, F. D. (2012). Multipath estimation in multicorrelator gnss receivers using the maximum likelihood principle. *IEEE Transactions on Aerospace and Electronic Systems*, *48*, 3222–3233.

Couellan, N. P. (2021a). The coupling effect of lipschitz regularization in neural networks. *SN Comput. Sci.*, *2*, 113.

Couellan, N. P. (2021b). Probabilistic robustness estimates for feed-forward neural networks. *Neural networks*, *142*, 138–147.

Cour, T., Sapp, B., & Taskar, B. (2011). Learning from partial labels. *Journal of Machine Learning Research*, *12*, 1501–1536.

Gao, B.-B., Xing, C., Xie, C.-W., Wu, J., & Geng, X. (2017). Deep label distribution learning with label ambiguity. *IEEE Transactions on Image Processing*, *26*, 2825–2838.

Garin, L., & Rousseau, J.-M. (1997). Enhanced strobe correlator multipath rejection for code & carrier. In *Proceedings of the 10th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS 1997)* (pp. 559–568).

Girshick, R. B., Shotton, J., Kohli, P., Criminisi, A., & Fitzgibbon, A. W. (2011). Efficient regression of general-activity human poses from depth images. In *2011 International Conference on Computer Vision* (pp. 415–422).

Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press.

Han, B., Yao, Q., Yu, X., Niu, G., Xu, M., Hu, W., Tsang, I., & Sugiyama, M. (2018). Co-teaching: Robust training of deep neural networks with extremely noisy labels. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, & R. Garnett (Eds.), *Advances in Neural Information Processing Systems*. Curran Associates, Inc. volume 31.

Hastie, T., Tibshirani, R., & Friedman, J. (2001). *The Elements of Statistical Learning*. Springer Series in Statistics. New York, NY, USA: Springer New York Inc.

He, K., Zhang, X., Ren, S., & Sun, J. (2016). Deep residual learning for image recognition. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 770–778).

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. R. (2012). Improving neural networks by preventing co-adaptation of feature detectors. *CoRR*, *abs/1207.0580*.

Hsu, L. (2017). GNSS multipath detection using a machine learning approach. In *Proceedings of the 2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)* (pp. 1–6).

Imani, E., & White, M. (2018). Improving regression performance with distributional losses. In J. Dy, & A. Krause (Eds.), *Proceedings of the 35th International Conference on Machine Learning* (pp. 2157–2166). PMLR volume 80 of *Proceedings of Machine Learning Research*.

Ioffe, S., & Szegedy, C. (2015). Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *Proceedings of the 32nd International Conference on International Conference on Machine Learning - Volume 37* ICML'15 (p. 448–456). JMLR.org.

Jardak, N., Vervisch-Picois, A., & Samama, N. (2011). Multipath insensitive delay lock loop in GNSS receivers. *IEEE Transactions on Aerospace and Electronic Systems*, *47*, 2590–2609.

Kong, S.-H., Cho, S., & Kim, E. (2022). Gps first path detection network based on mlp-mixers. *IEEE Transactions on Wireless Communications*, .

Kos, T., Markezic, I., & Pokrajcic, J. (2010). Effects of multipath reception on gps positioning performance. In *Proceedings ELMAR-2010* (pp. 399–402).

Lathuilière, S., Mesejo, P., Alameda-Pineda, X., & Horaud, R. (2020). A comprehensive analysis of deep regression. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *42*, 2065–2081.

Lecun, Y., Bottou, L., Bengio, Y., & Haffner, P. (1998). Gradient-based learning applied to document recognition. In *Proceedings of the IEEE* (pp. 2278–2324). volume 86.

LeNail, A. (2019). Nn-svg: Publication-ready neural network architecture schematics. *Journal of Open Source Software*, *4*, 747. URL: `https://doi.org/10.21105/joss.00747`. doi:`10.21105/joss.00747`.

Li, S., & Chan, A. B. (2014). 3d human pose estimation from monocular images with deep convolutional neural network. In *12th Asian Conference on Computer Vision (ACCV)*.

Mahendran, S., Ali, H., & Vidal, R. (2017). 3d pose regression using convolutional neural networks. In *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)* (pp. 2174–2182).

McGraw, G. A., & Braasch, M. S. (1999). GNSS multipath mitigation using gated and high resolution correlator concepts. In *Proceedings of the 1999 National Technical Meeting of The Institute of Navigation* (pp. 333–342).

Miao, S., Wang, Z. J., & Liao, R. (2016). A cnn regression approach for real-time 2d/3d registration. In *IEEE Transactions on Medical Imaging* (pp. 1352–1363). volume 35.

Murphy, K. P. (2012). *Machine Learning: A Probabilistic Perspective*. The MIT Press.

Oktal-SE (2020). SE-Nav software. URL: `https://www.oktal-se.fr/`.

Phan, H., Tan, S.-L., Mcloughlin, I., & Vu, L. (2013). A unified framework for GPS code and carrier-phase multipath mitigation using support vector regression. *Advances in Artificial Neural Systems*, .

Qin, H., Xue, X., & Yang, Q. (2019). Gnss multipath estimation and mitigation based on particle filter. *IET Radar, Sonar & Navigation*, *13*, 1588–1596.

Quan, Y., Lau, L., Roberts, G., Meng, X., & Zhang, C. (2018). Convolutional neural network based multipath detection method for static and kinematic GPS high precision positioning. *Remote Sensing*, *10*.

Sahmoudi, M., & Amin, M. G. (2008). Fast iterative maximum-likelihood algorithm (fimla) for multipath mitigation in the next generation of gnss receivers. *IEEE Transactions on Wireless Communications*, *7*, 4362–4374.

Savas, C., & Dovis, F. (2019). Multipath detection based on k-means clustering. In *Proceedings of the 32nd International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS+ 2019)* (pp. 3801–3811).

Schmidt, E., Gatsis, N., & Akopian, D. (2020). A GPS spoofing detection and classification correlator-based technique using the LASSO. *IEEE Transactions on Aerospace and Electronic Systems*, *56*, 4224–4237.

Shen, G., Jiao, Y., Lin, Y., Horowitz, J. L., & Huang, J. (2021). Deep quantile regression: Mitigating the curse of dimensionality through composition. `arXiv:2107.04907`.

Siemuri, A., Kuusniemi, H., Elmusrati, M. S., Välisuo, P., & Shamsuzzoha, A. (2021). Machine learning utilization in gnss—use cases, challenges and future applications. In *2021 International Conference on Localization and GNSS (ICL-GNSS)* (pp. 1–6).

Simonyan, K., & Zisserman, A. (2015). Very deep convolutional networks for large-scale image recognition. In Y. Bengio, & Y. LeCun (Eds.), *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Suzuki, T., & Amano, Y. (2021). NLOS multipath classification of GNSS signal correlation output using machine learning. *Sensors*, *21*.

Szegedy, C., Vanhoucke, V., Ioffe, S., Shlens, J., & Wojna, Z. (2016). Rethinking the inception architecture for computer vision. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (pp. 2818–2826).

Tao, Y., Liu, C., Chen, T., Zhao, X., Liu, C., Hu, H., Zhou, T., & Xin, H. (2021). Real-time multipath mitigation in multi-gnss short baseline positioning via cnn-lstm method. *Mathematical Problems in Engineering*, *2021*, 6573230.

Toshev, A., & Szegedy, C. (2014). Deeppose: Human pose estimation via deep neural networks. In *2014 IEEE Conference on Computer Vision and Pattern Recognition* (pp. 1653–1660).

Townsend, B., & Fenton, P. (1994). A practical approach to the reduction of pseudorange multipath errors in a Ll GPS receiver. In *Proceedings of the 7th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GPS 1994)* (pp. 143–148).

Townsend, B., Van Nee, R., Fenton, P., & Van Dierendonck, K. (1995). Performance evaluation of the multipath estimating delay lock loop. In *Proceedings of the 1995 National Technical Meeting of The Institute of Navigation* (pp. 277–283).

Tsui, J. B.-Y. (2005). *Fundamentals of Global Positioning System Receivers: A Software Approach*. Wiley Series in Microwave and Optical Engineering (2nd ed.). Wiley-Blackwell.

Ueda, M., Ito, K., Wu, K., Sato, K., Taki, Y., Fukuda, H., & Aoki, T. (2019). An age estimation method using 3d-cnn from brain mri images. In *2019 IEEE 16th International Symposium on Biomedical Imaging (ISBI 2019)* (pp. 380–383).

Van Dierendonck, A. J., Fenton, P., & Ford, T. (1992). Theory and performance of narrow correlator spacing in a GPS receiver. *Navigation*, *39*, 265–283.

Vigneau, W., Nouvel, O., Manzano-Jurado, M., Sanz, C., Carrascosa, A. H., Roviras, D., Juan, J. M., & Holsters, P. (2006). Neural networks algorithms prototyping to mitigate GNSS multipath for LEO positioning applications.

In *Proceedings of the 19th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2006)* (pp. 1752–1762).

Yi, D., Lei, Z., & Li, S. (2014). Age estimation by multi-scale convolutional network. In *Asian Conference on Computer Vision (ACCV)*.

Zhang, T., & Yu, B. (2005). Boosting with early stopping: Convergence and consistency. *The Annals of Statistics*, *33*, 1538–1579.

Zhang, Y., & Bartone, C. (2004a). Multipath mitigation in the frequency domain. In *PLANS 2004. Position Location and Navigation Symposium (IEEE Cat. No.04CH37556)* (pp. 486–495).

Zhang, Y., & Bartone, C. (2004b). Real-time multipath mitigation with wavesmoothtm technique using wavelets. In *Proceedings of the 17th International Technical Meeting of the Satellite Division of The Institute of Navigation (ION GNSS 2004)* (pp. 1181–1194).

Ziedan, N. I. (2018). Improved multipath and NLOS signals identification in urban environments. *NAVIGATION*, *65*, 449–462.