Deep learning methods for flood mapping

A review of existing applications and future research directions

Bentivoglio, Roberto; Isufi, Elvin; Jonkman, Sebastiaan N.; Taormina, Riccardo

Important note
To cite this publication, please use the final published version (if applicable).
Please check the document version above.

Hydrology and
Earth System
Sciences

Deep learning methods for flood mapping: a review of existing applications and future research directions

Roberto Bentivoglio[1], Elvin Isufi[2], Sebastian Nicolaas Jonkman[3], and Riccardo Taormina[1]

[1]Department of Water Management, Faculty of Civil Engineering and Geosciences,
Delft University of Technology, Delft, the Netherlands
[2]Department of Intelligent Systems, Faculty of Electrical Engineering, Mathematics and Computer Science,
Delft University of Technology, Delft, the Netherlands
[3]Department of Hydraulic Engineering, Faculty of Civil Engineering and Geosciences,
Delft University of Technology, Delft, the Netherlands

Correspondence: Roberto Bentivoglio (r.bentivoglio@tudelft.nl)

Abstract. Deep learning techniques have been increasingly used in flood management to overcome the limitations of accurate, yet slow, numerical models and to improve the results of traditional methods for flood mapping. In this paper, we review 58 recent publications to outline the state of the art of the field, identify knowledge gaps, and propose future research directions. The review focuses on the type of deep learning models used for various flood mapping applications, the flood types considered, the spatial scale of the studied events, and the data used for model development. The results show that models based on convolutional layers are usually more accurate, as they leverage inductive biases to better process the spatial characteristics of the flooding events. Models based on fully connected layers, instead, provide accurate results when coupled with other statistical models. Deep learning models showed increased accuracy when compared to traditional approaches and increased speed when compared to numerical methods. While there exist several applications in flood susceptibility, inundation, and hazard mapping, more work is needed to understand how deep learning can assist in real-time flood warning during an emergency and how it can be employed to estimate flood risk. A major challenge lies in developing deep learning models that can generalize to unseen case studies. Furthermore, all reviewed models and their outputs are deterministic, with limited considerations for uncertainties in outcomes and probabilistic predictions. The authors argue that these identified gaps can be addressed by exploiting recent fundamental advancements in deep learning or by taking inspiration from developments in other applied areas. Models based on graph neural networks and neural operators can work with arbitrarily structured data and thus should be capable of generalizing across different case studies and could account for complex interactions with the natural and built environment. Physics-based deep learning can be used to preserve the underlying physical equations resulting in more reliable speed-up alternatives for numerical models. Similarly, probabilistic models can be built by resorting to deep Gaussian processes or Bayesian neural networks.

1 Introduction

Flooding is one of the most dangerous and frequent natural hazards, accounting for significant human and economic losses every year (Jonkman and Vrijling, 2008). Because of climate change effects, more frequent and intense extreme precipitation is expected to further increase the severity of this hazard (Tabari, 2020). To mitigate the impact of floods on human lives and property, both preventive and emergency measures are required (European Union, 2007). Emergency measures are operations carried out just before, during, or after a flooding event. In those cases, real-time knowledge of the extent of the flood and the areas in danger is needed to execute countermeasures (Lendering et al., 2016). Instead, preventive measures are operations aiming at reducing the possibility of a certain area being flooded. Those can be de-

termined by maps that indicate the hazard of floods, i.e., the potential flood characteristics for an event.

There are the following three main flood maps used for dealing with such measures: (i) flood extent or inundation maps determine the observed inundation extent, during or after the event, and are used for emergency measures, (ii) susceptibility maps provide a qualitative categorization of the flood hazard in an area, given its physical characteristics, and are used for preventive measures, and (iii) flood hazard maps indicate the spatial distribution of variables that characterize the flood hazard of a specific event, such as flood depth and water extent, and are used for both emergency and preventive measures. Traditionally, inundation maps are obtained via remote sensing analysis (e.g., Lin et al., 2016), susceptibility maps with multi-criteria decision analysis (MCDA; e.g., Abdullah et al., 2021), and hazard maps with numerical methods (e.g., Dottori et al., 2022). Despite their wide usability, each method has its limitations. Remote sensing analysis for flood inundation requires manual or semi-automated procedures to improve the results and additional data such as land cover distribution (e.g., Manavalan, 2017). In addition, traditional models for flood inundation are not scalable to large amounts of data in the way that the ones currently produced by worldwide satellite missions are. MCDA for flood susceptibility is simple and interpretable, but its results are not accurate for complex phenomena (Khosravi et al., 2020). Moreover, the weights assigned to each criterion are subjective and thus biased by the external choices. Numerical methods for flood hazard modeling are robust and effective, but fast and accurate flood simulations remain a challenge (Costabile et al., 2017). There exist several ways to improve the speed of the simulations, for example, through parallel computing (e.g., Zhang et al., 2014; Ming et al., 2020; Glenis et al., 2013) or simplified models (e.g., Zhao et al., 2021b; Sridharan et al., 2021). However, parallel computing has high computational costs, and simplified models are unable to correctly reproduce rapidly evolving flows such as in urban floods (Costabile et al., 2017) and dam breaks (Prestininzi, 2008). Moreover, numerical models have intrinsic limitations which depend on the discretization of the governing physical equations and physical domain.

To overcome these limitations, practitioners and developers have used data-driven models based on machine learning. Machine learning (ML) is a branch of artificial intelligence in which a model improves its performance, with respect to some class of tasks, as the available data increases (Mitchell, 1997). Conventional ML techniques require the specific feature engineering of raw data before its processing. Deep learning (DL) can, instead, automatically discover the representations needed for detection or classification in raw data (LeCun et al., 2015). Nonetheless, data must be carefully selected according to the task at hand. DL methods are representation learning methods with multiple levels of representation obtained by composing simple but nonlinear modules that each transform the representation at one level (starting with the raw input) into a representation at a higher and more abstract level (LeCun et al., 2015). The model can then learn hidden patterns in the data and, consequently, improve its performance. Both ML and DL models have been applied in the fields of hydraulics and flood analysis. Mosavi et al. (2018) examined ML models for the prediction of floods in the short and long term. Sit et al. (2020) reviewed deep learning models for hydrology and water resources, focusing also on the hydrological modeling of floods. Zounemat-Kermani et al. (2020) reviewed neurocomputing for surface water hydrology and hydraulics, including some applications concerning floods.

The existing reviews mainly focused on the temporal variability in floods, especially concerning rainfall–runoff modeling, covering only a few instances of flood mapping applications. But the spatial evolution of flood events is extremely important to determine affected areas, plan mitigation measures, and inform response strategies. Yet, there are no comprehensive overviews and analyses of DL in flood mapping to facilitate flood researchers and practitioners. The aim of this review is thus to advance the emerging field of DL-based flood mapping by surveying the state of the art, identifying outstanding research gaps, and proposing fruitful research directions.

A total of 58 papers are analyzed considering two main parallel yet intertwined directions. On the one hand, we focused on the flood management application, spatial scale of study, and type of flood. On the other hand, we examined the deep learning model, type of training data, and performance with respect to alternative methods. This strategy provides insights from a flood management perspective and concurrently facilitates reflection on how to successfully apply DL models. The main insights from this paper can be summarized as follows:

1. We identify common patterns and deduce general considerations based on the presented results, while highlighting individual innovative approaches.

2. We compare against traditional methods to further validate the benefits of employing DL models.

3. We identify a series of current knowledge gaps and propose possible solutions to them, drawing from recent advancements in DL.

The remainder of this review is organized as follows. In Sect. 2, we present the background theory on both floods and deep learning. Then, in Sect. 3, we present the search methodology and discuss the results based on the reviewed papers. In Sects. 4 and 5, we present the knowledge gaps and propose possible future research directions. Finally, conclusions are provided in Sect. 6.

2 Background

This section is divided in two parts, namely flood management and deep learning. In the first part, we present the categories in which we classify flood management, while in the latter we describe the main deep learning models used for flood mapping.

2.1 Flood management

Floods can be defined as an overflow of water in otherwise dry land. Hence, flood management is a very broad field of interest; wherever there is water, there is a certain probability of being affected by it. While there exist several categorizations of flood management, we focus on types of floods, applications, and spatial scales.

2.1.1 Types of floods

We can distinguish flooding depending on how, why, and when it occurs.

- River floods are caused by extensive precipitation over long periods, causing the river to overflow its banks, ultimately inundating the neighboring areas. This process is slow and can last for several days (Serinaldi et al., 2018).

- Flash floods are caused by short but intense rainfall or sudden melting of snow (Sikorska et al., 2015). They are rapid and intense floods, typical of mountain and steep catchments. Flash floods are usually coupled with other hazards such as debris flows (Destro et al., 2018) and landslides (Ávila et al., 2016).

- Coastal floods are caused by extreme meteorological conditions which increase the water level in large bodies of water, due to a combination of low atmospheric pressure and strong winds. They occur near oceans, seas, or large lakes, and we also include tsunamis in this category, although they are generated by geological phenomena such as earthquakes.

- Urban floods are caused by the failure of drainage from a sewer system, due to extreme precipitation, resulting in the overflow of those pipes. Depending on the city position and topography, these floods can also be affected by all the other types of floods.

- Dam break and dike breach floods are caused by the failure of flood protection structures, due to extreme flood events or management issues. The uncertainty of if, where, and how a defense will fail further increases the unexpectedness of these phenomena.

To simplify the categorization, we excluded pluvial flooding, i.e., floods caused by the failure of a drainage system due to intensive precipitation. The underlying hypothesis is that pluvial floods can be addressed as urban floods in urban environments or river floods if they also feature rainfall-driven river overflows.

2.1.2 Flood mapping applications

Since we focus on the spatial variability in floods, we distinguish among three types of mapping, i.e., flood susceptibility, flood inundation, and flood hazard.

- Flood inundation maps determine the extent of a flood, during or after it has occurred (see Fig. 1a). Flood inundation maps represent flooded and non-flooded areas. This application is used for post-flood evacuation, protection planning, and for damage assessment. These maps can then also be used as calibration data for other applications such as flood susceptibility or flood hazard mapping. Flood images are obtained through remote sensing techniques and processed by histogram-based models (e.g., Martinis et al., 2009; Manjusree et al., 2012), threshold models (e.g., Cian et al., 2018), and machine learning models (e.g., Hess et al., 1995; Ireland et al., 2015).

- Flood susceptibility maps determine the tendency to flooding of a study area based on its physical characteristics (see Fig. 1b). This measure is only qualitative and does not evaluate any flood variable. However, it can provide reliable information when no quantitative data are available and can be used to easily assess areas at risk at large scales. Flood susceptibility mapping is performed by considering topographical, geographical, and meteorological factors (such as altitude, slope, lithology, land use, and rainfall) and comparing their spatial distribution with past flood events. This is done with multivariate analysis (e.g., Tehrany et al., 2014; Youssef et al., 2016) and multi-criteria decision analysis (e.g., Kazakis et al., 2015; Mahmoud and Gan, 2018).

- Flood hazard maps measure the water depth and extent across a flooded area (see Fig. 1c). Hazard maps also consider different return periods of the floods and, thus, the probability of a certain event. The latter is determined through a statistical analysis based on the frequency and intensity of floods (Bobée and Rasmussen, 1995). We will also refer to flood hazard when the water depths are estimated independently of the return periods. Flood hazard can also provide a measure of the flow velocities. Flood hazard maps are carried out by numerical models, which simulate flood events by discretizing the governing equations and the computational domain. We distinguish between one-dimensional (1D), two-dimensional (2D), and three-dimensional (3D) models with increasing complexity and, generally, accuracy (e.g., Horritt and Bates, 2002; Teng et al., 2017).
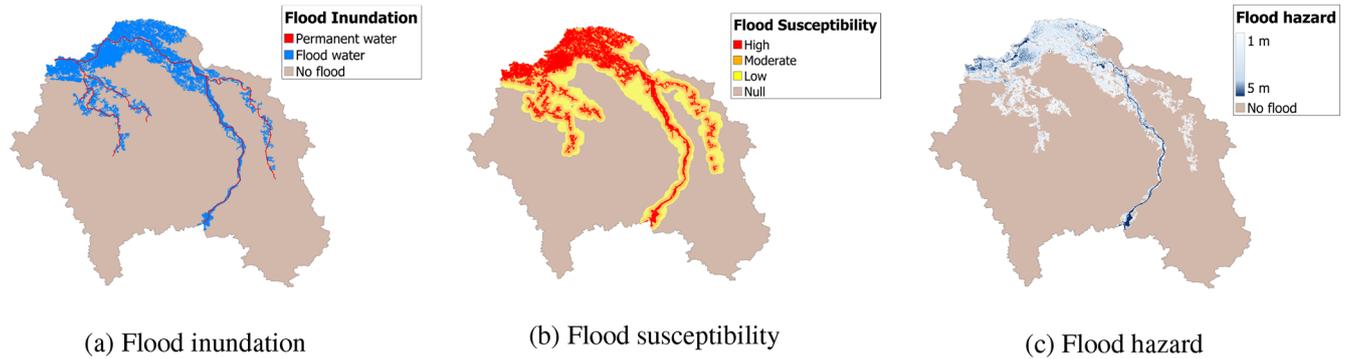
(a) Flood inundation   (b) Flood susceptibility   (c) Flood hazard

Figure 1. Examples of the types of flood maps analyzed for a representative area. Panel **(a)** shows a possible flood inundation map, panel **(b)** a flood susceptibility map, and panel **(c)** a flood hazard map, as defined in this paper.

Flood damage and flood risk maps (de Moel et al., 2009) are other examples of mapping applications. However, they are not described in more details here as no related DL-based paper was found in the literature. Similarly, the review also excludes applications which do not result in maps, such as water level forecasts.

2.1.3 Spatial scale

The importance of flood processes and the resolution of the flood maps varies with their spatial scale. Following de Moel et al. (2015), we also distinguish between local, regional, national, and supra-national scales. The choice between scales is often subjective, but here follows a rational categorization:

- Local scale refers to small study areas, such as towns or a specific river stretch. If a measure of the study area is given, we consider it in this category if the area is smaller than $100 \, \text{km}^2$.

- Regional scale considers a specific province, watershed, or large city. Study areas smaller than $100\,000 \, \text{km}^2$ belong to this scale.

- National scale refers to assessments of entire countries for which consistent (national) data are present. To exclude small countries, the study area must be greater than $100\,000 \, \text{km}^2$.

- Supra-national scales concern assessments of an entire continent or the globe.

2.2 Deep learning methods

Deep learning studies how neural networks learn representations from data through multiple levels of abstraction (LeCun et al., 2015). A neural network is a non-linear compositional model formed by a hierarchical layering of parametric functions that take an input variable x and produce an estimate $\hat{y}$ of a target representation y as $\hat{y} = f(x; \theta)$, where θ are the function's parameters. The purpose of DL is then to calibrate those parameters to have the best fit between predicted output and real output. The raw data x are input to the neural network and the output of each layer serves as input for the following layer, until the final layer, which coincides with the estimate $\hat{y}$. A neural network with L layers can be expressed as follows:

$$
\begin{aligned}
\hat{y} &= f_L(\cdot; \boldsymbol{\theta}_L) \circ f_{L-1}(\cdot; \boldsymbol{\theta}_{L-1}) \circ \ldots \circ f_1(\boldsymbol{x}; \boldsymbol{\theta}_1), \\
\boldsymbol{x}_\ell &= f_\ell(\boldsymbol{x}_{\ell-1}; \boldsymbol{\theta}_\ell), \quad \text{for } \ell = 1, \ldots, L, \\
\hat{y} &\equiv x_L,
\end{aligned} \tag{1}
$$

where $f_\ell(\cdot; \boldsymbol{\theta}_\ell)$ is the function at layer ℓ, $\circ$ represents the composition of functions, $\boldsymbol{\theta}_\ell$ are the trainable parameters, and $\hat{y}_\ell$ is the output layer ℓ. In a network architecture, the layers between the input and the output layer are called hidden layers since their output is not shown. Estimating parameters $\boldsymbol{\theta}_\ell$ is typically referred to as "learning", and it is performed by minimizing a loss function, through back-propagation (Rumelhart et al., 1986). Depending on the task, neural networks can be trained via supervised and unsupervised learning. Since, in flooding analysis, DL has been mainly approached via supervised learning, we focus on that learning process.

Supervised deep learning models identify a mapping from input to output, given a training set of input–output pairs. For example, a training set for flood hazard mapping may comprise a flood's rainfall hyetograph as input x and the corresponding maximum flooded area as output y. Thus, the loss function $l(y, \hat{y})$ compares the real output y with the predicted one $\hat{y}$. The loss function is typically the quadratic loss for regression problems, where the data are continuous (e.g., water depth), or the cross-entropy loss for classification problems, where the data are categorical (e.g., flooded and non-flooded areas). As training data, we can have observations or simulations. Observational data are derived from remote sensing, flood inventory maps, and measuring stations, while simulation data are derived from numerical solvers. Once a model is trained, its goodness of fit is analyzed with a test set composed of data that the model has not seen. If the model

performs well for the test set, it is said to generalize or extrapolate well. The ability to generalize is one of the most important properties of DL and becomes even more important in high-dimensional inputs (Balestriero et al., 2021).

2.2.1 Multi-layer perceptron

Among the possible neural network layers, fully connected ones are the most simple. In a fully connected layer, the layer propagation rule is given by the following:

$$x_\ell = f_\ell(x_{\ell-1}, \theta_\ell) = \sigma(W_\ell x_{\ell-1}), \qquad (2)$$

where x_ℓ is the output of the layer ℓ, $\sigma(\cdot)$ is a pointwise non-linearity (e.g., ReLU, $\sigma(x) = \max\{0, x\}$, or sigmoid, $\sigma(x) = \frac{1}{1+e^{-x}}$), $x_{\ell-1}$ is the input of the layer ℓ, and the training parameter W_ℓ is a weight matrix. Multi-layer perceptrons (MLPs) are composed by sequences of fully connected layers (Fig. 2a). The expressivity of the network increases with the dimensions of the hidden layers, as shown in Fig. 2a. When the dimension of the hidden layers decreases and then increases, as shown in Fig. 2b, the architecture is called encoder–decoder (ED). The idea behind this architecture is that only certain latent representations of the input are useful to represent the output (e.g., Taormina and Galelli, 2018).

In fully connected layers, the values of the parameters in W are independent between them, and there is no reuse of any of them. Thus, the number of learnable parameters is of the order of the input size, making fully connected layers inappropriate for inputs of large dimensions. This issue is referred to as the "curse of dimensionality" and implies that, as the dimension of the input increases, the amount of training data needed to learn representations increases exponentially (LeCun et al., 2015).

To overcome the curse of dimensionality, we need to exploit the structure in data. In flood analysis, data are usually structured; for example, neighboring pixels in raster data represent spatial proximity of nearby close elements, while discharge values in a hydrograph represent temporal proximities. Neural network layers can thus be defined in a way to exploit these data structures. These assumptions create what is known as an inductive bias, which imposes constraints on relationships and interactions among inputs in the learning process, thus prioritizing some solutions over others (Battaglia et al., 2018), as shown in Table 1. Inductive biases derive from the fundamental geometric principle of symmetry (Bronstein et al., 2021). The symmetry of a system is a transformation that leaves a certain property of said system unchanged. Symmetry results in invariance and equivariance properties. Invariance implies that transformations on the input features do not change the output (i.e., $f(g(x)) = f(x)$, $g(\cdot)$ being a generic transformation), while equivariance entails that transformations on the input features change the output via an equivalent transformation (i.e., $f(g(x)) = g'(f(x))$, $g'(\cdot)$ being a transformation equivalent to $g(\cdot)$). We explain the concept of invariance and equiv-

Table 1. Inductive biases and preferred types of data for different neural network layers (adapted from Battaglia et al., 2018).

Layer	Data type	Inductive bias
Fully connected	Unstructured data	–
Convolutional	Grid elements	Spatial equivariance
Recurrent	Sequences	Temporal equivariance

ariance with an example. Consider a picture with a flooded area in its top-left corner and one with the same flooded area shifted in the bottom-right corner. An invariant model would predict that there is a flooded area in both images, while an equivariant model would also reflect the change in position of the flood, i.e., identify that the flood is in the top-left corner in one case and in the bottom-right corner in the other. In this case, invariance and equivariance are associated to a spatial translation, but the same principle applies to other transformations, such as temporal translation. Inductive biases thus lead to the reuse of parameters in different parts of the input of each layer. For instance, convolutional kernels can be used on images of different dimensions, and recurrent layers can consider time series of variable length. Fully connected layers, instead, cannot have such inductive bias capabilities. The main characteristics for each considered layer are synthesized in Table 1. The input data type and the inductive biases are described for each studied layer.

2.2.2 Convolutional neural network

Convolution is an operation for which every entry of an input matrix is replaced by a spatially weighted average of its neighboring entries, as shown in Fig. 2d. The weights are defined by a matrix, called kernel, and are point-wise multiplied with the neighboring entries. This procedure is then repeated, using the same kernel, for every entry in the input. Convolutional layers are a neural network layer that apply convolution on a input using trainable kernels, i.e., the kernels' weights are learned during optimization (LeCun and Bengio, 1995). The propagation rule of layer ℓ of a convolutional layer is as follows:

$$x_{\ell+1} = \sigma(K_\ell * x_\ell), \qquad (3)$$

where K_ℓ is the kernel function for the ℓth layer, and $*$ is the convolution operator. Convolutional layers are mostly applied to images, i.e., two-dimensional spatial grids. For such inputs, the kernel is a 2D matrix. Convolutional layers have an inductive bias of translational equivariance, which reflects the idea that spatially close grid elements influence each other. This results in the reuse of the same kernel across the different input parts, and it implies that it matters where a pattern or object is in an image and that the model should be able to recognize it. Convolutional layers thus perform feature extraction, identifying relevant characteristics in the input.
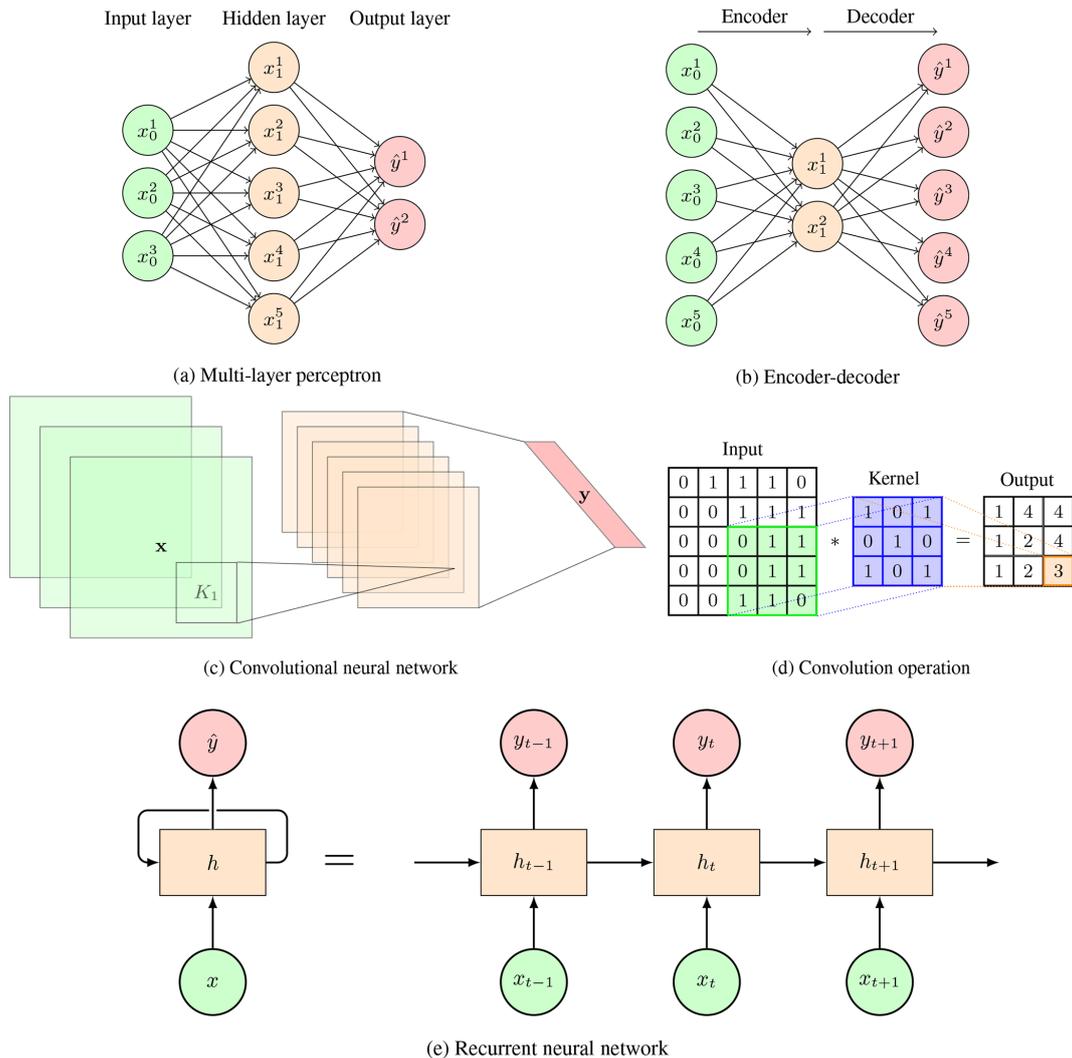
Figure 2. Deep learning architectures, with **(a)** a multi-layer perceptron (MLP) composed of a sequence of three fully connected layers. Every layer is connected to the following one by weights, represented by directed arrows. The values of the input, hidden, and output layers are represented, respectively, by vectors x_0, x_1, and $\hat{y}$. **(b)** An MLP encoder–decoder. The input data x_0 are encoded into a lower dimensional layer x_1 and then decoded into the output $\hat{y}$. This structure is also applicable to convolutional and recurrent layers. **(c)** A convolutional neural network (CNN) composed of a convolutional layer and a fully connected layer. The green squares represent an input tensor, the orange squares represent hidden layers, and the red parallelogram on the right represents the output layer. The small box $\mathbf{K_1}$ represents the convolutional kernel described in Eq. (3). The final layer depends on the task. **(d)** Visual explanation of how convolutional kernels work. Each element of the kernel is multiplied by its matching input value. Then, all values are summed to obtain the convolved output. This process is repeated across the whole input as the kernel shifts along it. **(e)** A recurrent neural network (RNN) in compact form (left) and in the unfolded form (right). The iterative structure of the RNN (left) can be unfolded in time to show how hidden states influence the solution at each time step (right). The coloring scheme indicates, for each architecture, the input (green), the state (orange), and the output (red).

Moreover, the reuse of parameters allows inductive learning over images of different sizes or resolutions. Different from fully connected layers, the number of parameters in a convolutional layer depends only on the kernel size because of this parameter-sharing property (see Fig. 2c). Depending on the input dimensions, we distinguish 1D convolutional layer for vector inputs, such as a rainfall hyetograph, 2D convolutional layers for matrix inputs, such as a digital elevation model (DEM), and 3D convolutional layers for tensor inputs,

such as stacked satellite images. Since 1D convolution considers translation equivariance on vectors, the inductive bias is equivalent to temporal equivariance if the vector is a time series.

Convolutional neural networks (CNNs) are composed of layers alternating convolution and pooling. Pooling operation replaces the output at a certain location with a summary statistic of the nearby features, thus reflecting translational invariance (Bronstein et al., 2021). They extract a single fea-

ture, such as the average or maximum value in a certain neighborhood of a point. Furthermore, pooling reduces the dimension of the input, speeding up computation. The final layers of a CNN are typically fully connected when dealing with classification or regression tasks. This layer allows us to map the convolved embedding to the number of classes or to the regressed value, respectively. Instead, if the task is to perform image segmentation, i.e., classify specific parts of an image, the final layers are composed of deconvolutional layers, which perform the inverse operation of convolutional layers, in an encoder–decoder structure. For details on convolutional layers and CNNs, refer to Goodfellow et al. (2016).

2.2.3 Recurrent neural network

Recurrent layers are used for processing sequential data, such as time series (Rumelhart et al., 1986). A recurrent layer can be seen as a non-linear state space model expressing the output at time t, y_t, as a function of a former hidden state h_t and input x_t. The basic formulation for a recurrent layer is as follows:

$$h_t = \sigma(\mathbf{W}h_{t-1} + \mathbf{U}x_t),$$
$$y_t = \sigma(\mathbf{V}h_t), \tag{4}$$

where $\mathbf{U}$, $\mathbf{V}$, and $\mathbf{W}$ are trainable weight matrices. As it follows from (Eq. 4), the hidden state encodes the temporal memory of previous time instances while the output mapping is instantaneous. These matrices are shared across time, allowing the recurrent layer to exploit the temporal proximities of sequential data, irrespective of their position. This is, for instance, the case for discharge hydrographs (e.g., Zhou et al., 2021). Because there is an inductive bias in temporal sequences, they allow us to reuse parameters without affecting the performance.

Recurrent neural networks (RNNs) are neural networks composed of recurrent layers. The iterative structure of the RNNs can be unfolded in time to show how hidden states influence the output at each time step (Fig. 2e). However, the basic recurrent layer in Eq. (4) suffers from the problem of vanishing and exploding gradients (Hochreiter and Schmidhuber, 1997). This occurs due to the iterative use of the same layer which causes the weights to multiply several times when back-propagating the error, ultimately leading to vanishing gradients if the weights are small and exploding gradients if the weights are large. This then constrains the temporal memory of these networks and limits their capability to extract long-term dependencies between the past inputs and the current output.

This problem is typically solved via the use of long short-term memory (LSTM) layers (Hochreiter and Schmidhuber, 1997). This variation in recurrent layers also improves the hidden state mechanism, even allowing it to remember information which is temporally distant well. Another common variation is the gated recurrent unit (GRU; Cho et al., 2014), which achieves comparable results with the LSTM architec-

ture while using a simpler formulation. Similar to fully connected and convolutional layers, recurrent layers can be used in encoder–decoder architectures. This structure can be composed of an RNN which generates a latent representation, followed by another RNN that decodes it (e.g., Cho et al., 2014).

The most successful applications of RNNs for flood management regard tasks related to sequences and time series analysis, such as rainfall–runoff modeling (e.g., Kratzert et al., 2019a). While RNNs are preferred over 1D CNNs, recently the latter started gaining momentum for some time series learning tasks (e.g., Oord et al., 2016).

3 Review

3.1 Methodology

Papers were retrieved from the Scopus database by combining the keywords "deep learning" or "neural network" with "flood" or "flooding". The 3338 publications obtained were then filtered to include only journal papers from January 2010 until December 2021, in the areas of engineering, environmental science, and Earth and planetary sciences. From this reduced list of 1308 papers, we considered the following two major refining criteria: (i) the papers should be based on the deep learning models presented in Sect. 2.2, and (ii) the applications must address the spatial variability of floods (i.e., not focusing only on the temporal aspects of flood analysis). This procedure resulted in 46 reviewable papers. This list was finally extended via a snowball search that considered cited and citing works, ultimately leading to 58 eligible documents (Fig. 3). We find that the described methodology selected a representative subset for producing a thorough review of recent advances and developments in this field.

The selected papers are listed in Table 2 which reports the major details, including the flood mapping application, the type of flood, the DL model, and the spatial scale. General findings related to these four criteria are first presented in Sect. 3.2. Specific findings for each application are then presented in Sects. 3.3 (flood inundation), 3.4 (flood susceptibility), and 3.5 (flood hazard). These specific sections provide a more in-depth discussion on the deep learning models employed, with a focus on the architecture, the input and output data, and the performance assessment.

3.2 General findings

3.2.1 Flood mapping applications

Figure 4 shows the distribution of papers for each of the applications considered, i.e., flood inundation, flood susceptibility, and flood hazard. The research community has dedicated efforts to investigate each type of application, although flood inundation and susceptibility have received the most
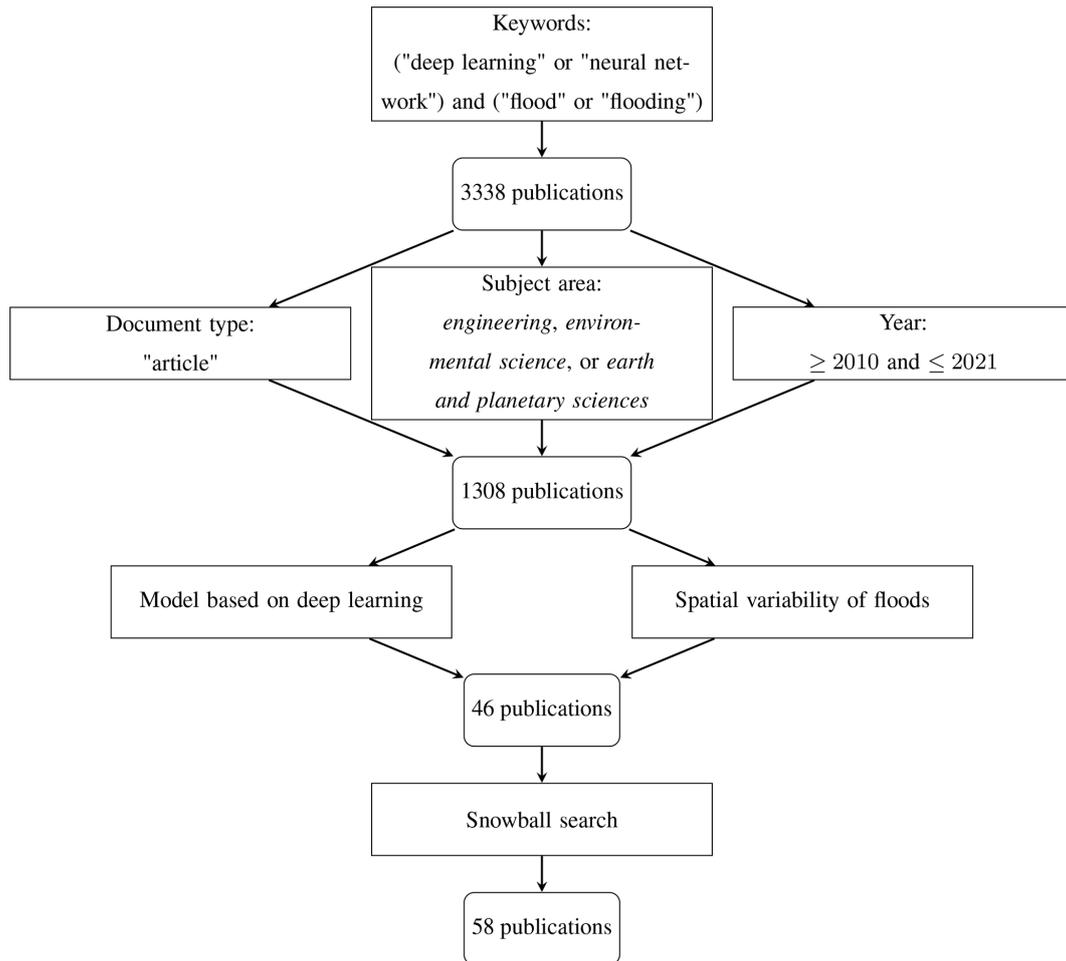
Figure 3. Flowchart of the methodology applied for the paper selection.

attention. While papers on flood inundation are more evenly distributed across years, applications for flood susceptibility and, especially, flood hazard have increased in the last few years. Similar to what was observed in related fields such as hydrology (e.g., Sit et al., 2020), a strong surge in DL publications for spatial flood analysis is witnessed between 2018 and 2019. These years identify a turning point for AI in Earth system sciences driven by the adoption of CNN (striped patterns in Fig. 4) and RNN (dotted patterns) in lieu of traditional MLP models. The late use of convolutional and recurrent models is motivated by their recent popularization and development, along with a rise in awareness of the ML advancements, contrary to fully connected layers, that have a longer application history.

3.2.2   Flood types

Figure 5 shows the types of flood analyzed with respect to each application. River floods are the most common, with many applications in inundation and hazard mapping. This is probably because, for historical reasons, most cities in the world are built close to rivers (Kummu et al., 2011). The scientific community has dedicated significant effort to exploring the potential of DL for urban flooding. This is difficult to model because of the complex topography and the presence of a drainage system whose dynamics need to be coupled with the overland flood (Löwe et al., 2021). Almost all papers analyzing flash floods described flood susceptibility mapping applications. This is expected due to the short duration and the contingent nature of these phenomena, which limit remote sensing imaging and numerical simulations used in flood inundation and flood hazard mapping, respectively. Despite the importance of coastal flooding (Neumann et al., 2015), only a few papers report the use of DL for coastal flooding. While other works are available in the literature (Lütjens et al., 2020, 2021; Bowes et al., 2021), they were not considered since the employed DL models were not trained via supervised learning. Some of these works will be discussed in Sect. 5. Dam break floods are the least analyzed type, possibly because of their relatively rare occurrence and complexity.

Table 2. Deep learning applications for flood mapping. References are classified in terms of flood mapping application, type of flood, deep learning (DL) model, training data, and spatial scale.

Application	Flood type	DL model	Training data	Spatial scale	Reference(s)
Inundation	River	MLP	Observations	Regional	Li et al. (2016a, 2015)
		CNN	Observations	Local	Gebrehiwot et al. (2019); Nogueira et al. (2017); Hou et al. (2021); Ichim and Popescu (2020); Hashemi-Beni and Gebrehiwot (2021); Wieland and Martinis (2019)
				Regional	Sarker et al. (2019); Kang et al. (2018); Nemni et al. (2020); Isikdogan et al. (2017)
	Urban	MLP	Observations	Local	Amini (2010)
				Regional	Li et al. (2016b)
		CNN	Observations	Local	Peng et al. (2019)
		RNN, CNN	Observations	Local	Dong et al. (2021)
	Coastal	CNN	Observations	Regional	Liu et al. (2019); Isikdogan et al. (2017)
			Simulations	Regional	Muñoz et al. (2021)
	Dam break	MLP	Observations	Regional	Syifa et al. (2019)
Susceptibility	River	MLP	Observations	Regional	Jahangir et al. (2019); Khoirunisa et al. (2021); Ahmadlou et al. (2021); Popa et al. (2019); Kia et al. (2012); Ahmed et al. (2021); Chakrabortty et al. (2021b); Saeed et al. (2021)
		CNN	Observations	Regional	Y. Wang et al. (2020)
				National	Khosravi et al. (2020)
		RNN	Observations	Regional	Fang et al. (2020a)
	Flash	MLP	Observations	Regional	Tien et al. (2020); Ngo et al. (2018); Popa et al. (2019); Costache et al. (2020); Chakrabortty et al. (2021a)
				National	Kourgialas and Karatzas (2017)
		CNN	Observations	Regional	Panahi et al. (2021); Liu et al. (2021)
	Urban	MLP	Observations	Local	Darabi et al. (2021)
				Regional	Kalantar et al. (2021)
		CNN	Observations	Local	Zhao et al. (2021c, 2020)
				Regional	Lei et al. (2021)
Hazard	River	MLP	Simulations	Local	Chu et al. (2020); Huang et al. (2021a); Xie et al. (2021); Lin et al. (2020b, a); Jacquier et al. (2021)
		CNN	Simulations	Local	Kabir et al. (2020); Hosseiny (2021)
		RNN	Simulations	Regional	Zhou et al. (2021); Kao et al. (2021)
	Flash	CNN	Simulations	Local	Yokoya et al. (2020)
	Urban	MLP	Simulations	Local	Berkhahn et al. (2019); Chang et al. (2010)
		CNN	Simulations	Regional	Guo et al. (2021); Löwe et al. (2021)
	Coastal	RNN	Simulations	Local	Hu et al. (2019)
	Dam break	MLP	Simulations	Local	Jacquier et al. (2021)

MLP is the multi-layer perceptron, CNN is the convolutional neural network, and RNN is the recurrent neural network.

3.2.3 Spatial scale

As shown in Fig. 6, most applications consider local and regional scales. Local scale refers to towns (e.g., Darabi et al., 2021; Berkhahn et al., 2019), small catchments (e.g., Lin et al., 2020a; Kabir et al., 2020), or river reaches (e.g., Chu et al., 2020; Gebrehiwot et al., 2019). As such, they are mostly referred to as urban and river floods. The cases sizes vary from very small ones, 165 m^2 (Hou et al., 2021), to small towns up to 100 km^2 (Lin et al., 2020a). Regional-
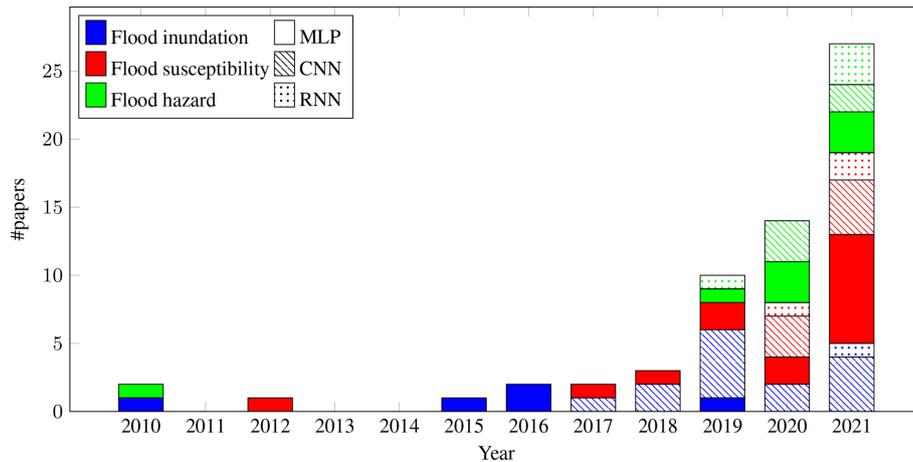
Figure 4. Publications by year, type of application, and type of DL model. The increasing trend of the last 5 years has been mostly driven by the applications in flood susceptibility and flood hazard.
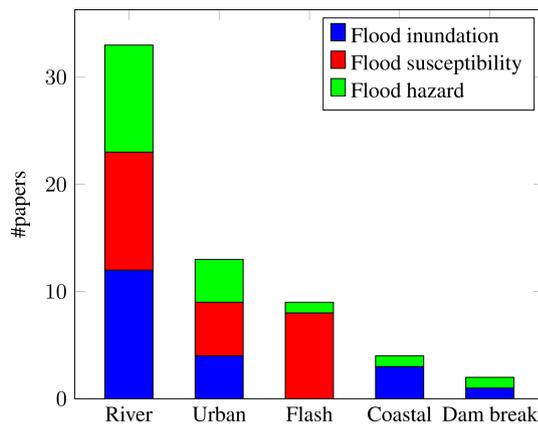

Figure 5. Distribution of the types of floods per flood application in the reviewed papers. River and urban floods are the most common, while flash and coastal floods have fewer occurrences.

scale models consider a catchment (e.g., Popa et al., 2019), a province (e.g., Y. Wang et al., 2020), or large cities (e.g., Löwe et al., 2021; Kalantar et al., 2021). Most works focus on river floods, while some study flash, urban, and coastal floods. National-scale models refer to the assessments of entire countries, with only two papers concerning such scales, respectively, for Iran and Greece (Khosravi et al., 2020; Kourgialas and Karatzas, 2017). Nemni et al. (2020) and Sarker et al. (2019) consider several study areas across Africa and Asia and Australia, respectively, but since the size of each area was smaller than $100\,000\,\text{km}^2$, they were marked as regional-scale models. They also do not fit within the national-scale classification since they do not encompass whole nations. Supra-national-scale models assessing the entire globe or a continent have not been studied yet with deep learning models. This seems unexpected, since ML techniques have already been employed at global

scales, outperforming traditional techniques, for example, in the estimation of design floods along river networks (e.g., Zhao et al., 2021a). Since DL models have been shown to outperform ML models, as later outlined in this review, more models should be used at those scales in future studies.

3.2.4 DL architecture

Figure 4 reports the architecture used for each application, showing that DL models are mainly based on fully connected and convolutional layers.

MLP networks are widely used due to their flexibility and ease of implementation. However, they are usually coupled with other techniques to reach satisfactory performances. Stochastic optimization techniques, such as the genetic algorithm, firefly algorithm, and particle swarm optimization were combined with MLPs to search the optimal model's parameters (e.g., Li et al., 2015; Ngo et al., 2018; Kalantar et al., 2021). Multi-criteria decision analysis models, such as frequency ratio and analytical hierarchy process, were also coupled with MLPs to adjust the weights of each input in flood susceptibility (e.g., Kourgialas and Karatzas, 2017; Costache et al., 2020; Popa et al., 2019). Furthermore, k-means clustering was used to categorize the dataset in classes, to account for different topographical conditions; then, for each class, an MLP was trained (e.g., Chang et al., 2010; Huang et al., 2021a). Combining MLPs with such methods partly compensates the lack of inductive biases; however, this lack blocks the model from employing existing structures in the data, ultimately limiting their usability. Since flooding phenomena have spatial and temporal structures, we expect MLPs to become progressively less used in this field, as hinted by the trend in Fig. 4.

CNNs are best suited for processing raster files and images, thanks to their spatial inductive bias. Since most data for flood analysis (e.g., elevation data, rainfall distribution
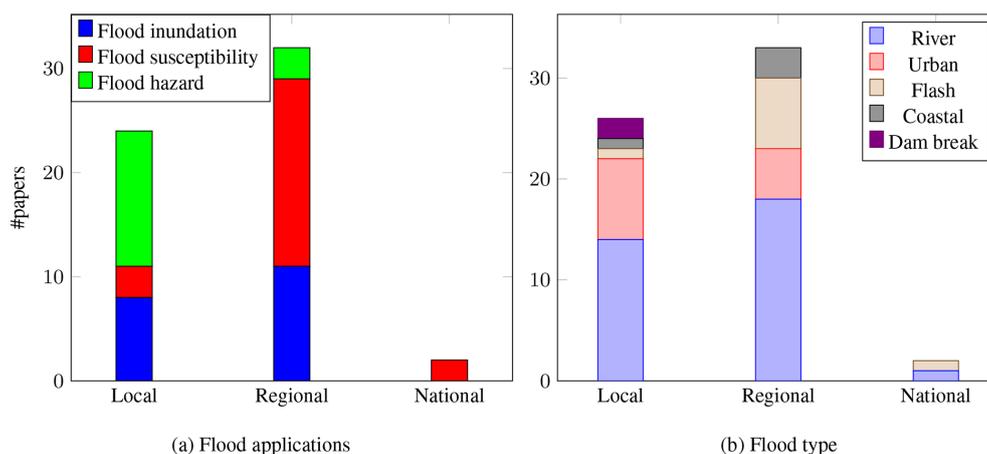
Figure 6. Distribution of the spatial scale per **(a)** flood application and **(b)** type of flood in the reviewed papers. Local and regional scales are the most used.

fields, and remote sensing image) come in this format, CNNs have been increasingly employed by the research community in the recent years. While most papers consider standard CNNs, there are a few which employ 1D CNNs (e.g., Dong et al., 2021; Guo et al., 2021; Liu et al., 2021) and 3D CNNs (e.g., Y. Wang et al., 2020; Fang et al., 2020a). 1D CNNs consider as input a hyetograph or a hydrograph of a certain event, while 3D CNNs consider raster files stacked upon each other. Regarding the architecture, different papers for flood inundation consider an encoder–decoder structure for image segmentation and classification (e.g., Nemni et al., 2020; Hashemi-Beni and Gebrehiwot, 2021; Liu et al., 2019). For such papers, the input is a satellite image of a flood, and the output is its classification in flooded and non-flooded areas. This architecture allows the models to increase their performance since they can retain high-frequency details in the segmented images (Badrinarayanan et al., 2017).

Guo et al. (2021) and Löwe et al. (2021) use a convolutional encoder–decoder structure for flood hazard mapping to embed a rainfall hyetograph in the latent space. In this way, they can consider both spatial and temporal data within the same framework.

RNNs have been mostly employed to model temporally varying floods, where they can exploit best their sequential inductive bias. However, they remain the least common choice of DL architecture for spatial flood analysis. Most papers apply RNNs on a time series, such as a hyetograph or a hydrograph (e.g., Kao et al., 2021; Zhou et al., 2021). Some papers, instead, consider spatial sequentiality by reshaping the original raster data into vectors (e.g., Fang et al., 2020a; Panahi et al., 2021; Lei et al., 2021). For example, Fang et al. (2020a) extract, for each pixel, its neighboring pixels in a 3×3 window and then convert them into a vector based on spatial contiguity. However, this operation introduces arbitrariness in the sequential order chosen for arranging the input pixels, since it is independent of the underlying topography. In fact, Panahi et al. (2021) and Lei et al. (2021) show that these models underperform when compared with CNNs. Among the different RNN layers, most works consider LSTM units (Kao et al., 2021; Zhou et al., 2021; Fang et al., 2020a), but simple recurrent units (Panahi et al., 2021; Huang et al., 2021a) and GRUs (Dong et al., 2021) have also been employed. Some papers analyzed the potential of RNNs in combination with other techniques. Kao et al. (2021) use an encoder–decoder architecture to forecast flood features based on rainfall patterns. The encoder and the decoder steps are composed of fully connected layers, while an LSTM is present in the latent space to process rainfall data. Zhou et al. (2021) identify representative spatial locations in the study area. Then, an LSTM is trained to simulate the water levels' evolution in time at each location. A water surface is ultimately determined by interpolating the water depth at those points. Dong et al. (2021) combine 1D CNNs and RNNs on an urban channel network. The model takes as input the channels' properties, such as their cross sections, and rainfall and water level measures, which are taken from sensors in the network. This input is then given in parallel to a 1D CNN and to a GRU, whose output is then combined to predict the temporal evolution of the flood. Hu et al. (2019) deploy the LSTM model in a lower-dimensional space, obtained via proper orthogonal decomposition and singular value decomposition. The model then requires fewer data to be trained.

3.2.5 Performance assessment

This section discusses different approaches for assessing the performance of the DL models, i.e., how well they match the outcomes of traditional and machine learning models. Flood susceptibility and inundation models are compared with techniques such as frequency ratio (Popa et al., 2019), a type of MCDA model, the soil conservation service runoff model (Jahangir et al., 2019), a hydrologic model, and automatic threshold model (Nemni et al., 2020), a histogram-

based model. They are also compared with machine learning techniques, such as support vector machines (e.g., Sarker et al., 2019; Gebrehiwot et al., 2019; Zhao et al., 2020), random forest (e.g., Darabi et al., 2021; Zhao et al., 2020), adaptive neuro-fuzzy inference system (Panahi et al., 2021), deep boost (e.g., Chakrabortty et al., 2021a; Ahmed et al., 2021), and radial basis function (Nogueira et al., 2017). DL models are shown to outperform both traditional and ML models in terms of the accuracy of the results. Flood hazard models, instead, are compared against numerical models, since they act as surrogate models. Thus, their main purpose is to increase computational speed while maintaining low prediction errors.

There are also a few papers that compared different DL models. Huang et al. (2021b) compared MLPs with RNNs, while Fang et al. (2020a) showed that MLPs were outperformed by the more inductive-biased approaches such as RNNs, 1D CNNs, and 3D CNNs. Wieland and Martinis (2019) showed that CNNs widely outperform MLPs, as expected, because of their inductive bias capabilities. Besides accuracy, the number of parameters and the data requirements are important factors when comparing DL models. A higher number of parameters results in better performances but may also lead to overfitting, which is a condition where the model decreases its performance on the testing data. Hence, when deployed in similar settings, such a model would perform drastically worse. Moreover, data are not always available, leading to possibly unfair comparisons between models with different data budgets. As such, the same model may give different outcomes, depending on the considered case.

In supervised learning, we distinguish between regression and classification problems, depending on whether the target values to predict are continuous (e.g., water depth) or discrete (e.g., flooded vs. non-flooded area), respectively. Depending on the task, we employ a different set of metrics to evaluate model performances.

Regression metrics are a function of the differences, or residuals, between target and predicted values. The most common metrics include the root mean squared error (RMSE), the coefficient of determination (R^2), and the mean average error (MAE). RMSE and MAE improve as they approach zero, while R^2 improves as it approaches one. In general, MAE may be preferred to RMSE since the latter is heavily influenced by the presence of extreme outliers. However, since both metrics are averaged on a domain, their comparison across different works requires careful attention.

Classification tasks can be either binary (e.g., predict flooded and non-flooded locations) or multi-categorical (e.g., classifying between permanent water bodies, buildings, and vegetated areas), according to the output number of classes. In the following discussion, we focus on the former, with concepts extending to the second case. When computing binary classification metrics, flooded areas are generally represented as a positive class, while non-flooded areas are gener-

ally represented as a negative class. The most common metrics for flood modeling are accuracy, recall, and precision, followed by other indices such as the area under the receiver operator characteristic curve. Accuracy represents the number of correct predictions over the total. While popular and easy to implement, this metric is inappropriate for imbalanced datasets, where some categories are more represented than others. For example, if test samples feature an average of 90 % in a non-flooded area, a naïve model constantly predicting no flooding would reach 90 % accuracy, despite having the wrong assumptions. Furthermore, since it may be better to overestimate a flooded area than to underestimate it, one could resort to metrics such as recall that account for false negatives and thus penalize models that cannot recognize a flooded area correctly. However, when used alone, recall can lead to similar issues to those described for accuracy, e.g., yielding a perfect score for a model always predicting the entire domain as flooded. Thus, for an exhaustive understanding of the model's performance, one should also consider metrics accounting for false positives, i.e., where the model misclassifies non-flooded areas as flooded. There are several possible metrics, such as the F1 score, the Kappa score, or the Matthews correlation coefficient, each with their drawbacks and benefits (e.g., Wardhani et al., 2019; Delgado and Tibau, 2019; Chicco and Jurman, 2020). A reasonable choice is the F1 score, which is the geometric mean of recall and precision, and it thus equally considers both false negatives and false positives. Another good example is the ROC (receiver operating characteristic) curve that describes how much a model can differentiate between positive and negative classes for different discrimination thresholds (Bradley, 1997). The area under the ROC curve (AUC) is often used to synthesize the ROC as a single value. However, the AUC loses information on which parts of the dataset the model performs best. For this reason, one should always interpret these results carefully, especially when comparing different studies. Our purpose here is to show that, for the same case study, DL tends to outperform traditional models.

For surrogate models, the comparison is also performed in terms of their speed-up, which is determined as the ratio between the simulation time of the numerical model and the simulation time of the DL model. For a correct comparison, the training time of the DL model must be considered as well in this analysis. However, this was done only by a few papers (e.g., Guo et al., 2021; Kabir et al., 2020; Jacquier et al., 2021).

3.3 Deep learning for flood inundation

Flood inundation maps determine the extent of a flood during or after its occurrence. We remind the reader that, in this paper, we refer to flood inundation as the process of mapping flooded and non-flooded areas from a picture of a flood. This classification is usually binary (e.g., Peng et al., 2019; Nemni et al., 2020), but it can also be extended to include
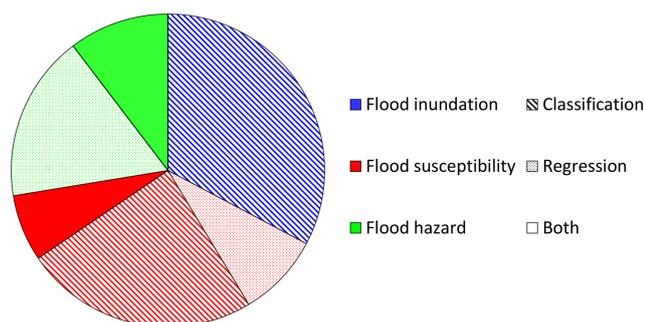
Figure 7. Distribution of the comparison metrics per type of application. The colors represent the different types of applications, while the patterns represent the considered metrics.

permanent water bodies (e.g., Sarker et al., 2019), vegetation (e.g., Ichim and Popescu, 2020), buildings (e.g., Hashemi-Beni and Gebrehiwot, 2021), and more (e.g., Muñoz et al., 2021). All types of floods were well represented for this application, except for flash floods (Fig. 5). We attribute this to the limited frequency of observation of most remote sensing techniques.

Regarding the spatial scale, most papers focused on local and regional scales. The availability of remote sensing at wider scales is increasingly higher (e.g., Observatory, 2021); however, this seems to be only partially considered. A plausible reason is the limited frequency of observation of the satellites. High temporal remote sensing imagery has a low spatial resolution. Few papers tackle this issue by increasing the resolution of the predicted flood maps, via a neural network, with a technique known as super-resolution (e.g., Li et al., 2015, 2016b). Super-resolution enhances the quality of an input low-resolution image (W. Yang et al., 2019). These papers show that MLPs improve the accuracy of super-resolution mapping with respect to other techniques, such as spatial attraction models. We argue that further improvements of super-resolution could be obtained by employing CNNs, which lend themselves naturally to such tasks, as demonstrated by applications in similar fields (Ma et al., 2019).

3.3.1 DL architecture

As the task of recognizing floods from a picture can be regarded as an image segmentation task, most deep learning models used are based on CNNs. There are also a few earlier papers that use MLPs (e.g., Li et al., 2016a; Amini, 2010) because CNNs were not yet adopted by researchers in the field. Dong et al. (2021) use a combination of RNNs and 1D CNNs to determine the temporal evolution of flooded and non-flooded nodes in an urban channel network, as described previously. In this case, the choice of recurrent and 1D convolutional layers is well motivated due to their temporal inductive bias.

3.3.2 Input and output data

Satellite data are the most used input for flood inundation applications (e.g., Sarker et al., 2019; Peng et al., 2019; Nogueira et al., 2017). Other input data sources include unmanned aerial vehicle data (UAV; e.g., Gebrehiwot et al., 2019; Ichim and Popescu, 2020), hydrographs (e.g., Hou et al., 2021), and DEMs (e.g., Hashemi-Beni and Gebrehiwot, 2021; Muñoz et al., 2021). Only Dong et al. (2021) differ from the other papers by considering sensors in the place of flood pictures. Inundation maps produced by 3D numerical models are also used as target prediction (Muñoz et al., 2021). The results from the numerical model can be used as a detailed reference for the DL model. Satellite data and UAV imagery are both remote sensing data that represent a flood event seen from above. The main differences concern the scale, the resolution, and the availability. UAVs are applicable only for small areas, but their resolution is higher than satellite data. UAVs can be readily used but may be unavailable in certain areas. On the other hand, satellite data are available worldwide but the frequency of observation can be limiting. Satellites can also struggle to extract information below clouded areas (e.g., Meraner et al., 2020). When combining information from different sources, the input data have different resolutions, leading to possible problems for some deep learning models, which take fixed-size inputs. One way to integrate different data resolutions is by data fusion (e.g., Muñoz et al., 2021). This process allows the creation of more consistent, accurate, and useful information than that provided by any individual data source.

3.3.3 Performance assessment

As defined in Sect. 3.3, flood inundation mapping determines which cells of the flood picture are represented as flooded or not. Thus, the task is regarded as a classification problem, as confirmed by the metrics used (Fig. 7). The selected papers often use several metrics (see Table A1 in the Appendix), but for clarity, we consider a single metric for each work. The metric selection depends on the employed ones and follows the considerations presented in Sect. 3.2.5, with preference for metrics such as F1, AUC, or recall, if available. Deep learning models have consistently shown improved performances in terms of the selected metrics (Table 3). Li et al. (2015, 2016b) compare optimization techniques with and without MLPs for super-resolution-based flooding. They show that a DL model slightly increases the performances. This may be because the models are based on MLPs and thus neglect any spatial structure in the data, which could be considered, instead, by CNNs. Most CNN models show noticeable improvements with respect to traditional threshold methods, such as the normalized difference water index (NDWI) and automatic threshold model (ATM; e.g., Wieland and Martinis, 2019; Isikdogan et al., 2017; Nemni et al., 2020), and with respect to machine learning models such

as random forest (RF) and support vector machine (SVM). This reflects similar results obtained in image detection tasks (Badrinarayanan et al., 2017).

3.4 Deep learning for flood susceptibility

Flood susceptibility determines the tendency to flood in a study area based on its physical characteristics and given a set of known past flood events. This is done by assigning to each location a level of susceptibility ranked from low to high (see Fig. 1b). The susceptibility depends on the distribution of the inputs, often called flood conditioning factors, in the function of recorded past flood events. The deep learning model then computes, for each point in the area, a score from 0 (non-flooded) to 1 (flooded). These scores are finally divided into several classes, generally using the natural (Jenks) breaks method (e.g., Fang et al., 2020a; Y. Wang et al., 2020; Khoirunisa et al., 2021), to obtain a susceptibility map. An exception is given by Jahangir et al. (2019) and Kia et al. (2012), who train their models to predict discharge values and then use a geographic information system (GIS) model for the mapping. In both cases, the model performs well when the recorded flood events occur in the predicted high-susceptibility areas.

There exist DL-related applications for all types of floods (see Fig. 6b). Furthermore, Fig. 6a shows that most of the works are concerned with regional or wider scales (e.g., Tien et al., 2020; Panahi et al., 2021; Khosravi et al., 2020). This is expected, since susceptibility mapping gives a qualitative estimate of which locations are prone to flooding. Operating on small scales may thus be limiting, both in terms of data availability and applicability for prevention strategies. The data requirements for an accurate estimate would probably be too high for a small area.

3.4.1 DL architectures

Most papers use MLP and CNN. Models based on MLPs consider single points or pixels as inputs (Tien et al., 2020; Ahmadlou et al., 2021; Khoirunisa et al., 2021), while CNNs consider the whole raster files (Zhao et al., 2020; Khosravi et al., 2020; Y. Wang et al., 2020). Since MLPs lack inductive bias, they provide less coherent results, meaning that the variation among neighboring cells can be high. This is partially solved by coupling the MLP architecture with other statistical techniques, such as frequency ratio (e.g., Darabi et al., 2021; Popa et al., 2019; Costache et al., 2020). Instead, CNNs have a spatial inductive bias; thus, they inherently consider the structure of the input, providing more coherent flood maps (e.g., Khosravi et al., 2020). However, Y. Wang et al. (2020) and Liu et al. (2021) show that 1D CNNs, which perform convolution on the input features for each domain's cell, are not suited for this problem, as they do not properly leverage any inductive bias. Some works showed that deep belief networks (DBNs), which are an unsupervised variation

of MLPs, could outperform standard MLPs in flood susceptibility mapping (e.g., Shirzadi et al., 2020; Pham et al., 2021).

3.4.2 Input and output data

The inputs for the deep learning models are several. We distinguish between the following five input typologies:

1. topographical inputs, which are derived from a digital elevation model, such as elevation, slope, and aspect;

2. meteorological inputs, related to the hydrological characteristics and derived from measuring stations and satellites, such as rainfall distribution and frequency;

3. geological inputs, related to the properties of the soil, such as lithology and soil type;

4. geographical inputs, related to observable surface characteristics and obtained through remote sensing, such as land use and normalized difference vegetation index; and

5. anthropogenic inputs, related to the presence of artificial environments, such as distance from roads.

Topographical data were the most frequent type of input. Many papers present a sensitivity analysis to determine which factors influenced the final results the most. On average, these were slope, land use, aspect, terrain curvature, and distance from the rivers (e.g., Khosravi et al., 2020; Fang et al., 2020a; Popa et al., 2019; Costache et al., 2020). A complete list of inputs is reported in the Appendix (Fig. B1). As there are several typologies of inputs, it is important to design an appropriate model to integrate heterogeneous environmental information.

As output data, most papers considered a flood inventory map given by a set of flooded and non-flooded locations. The flooded locations were derived from measurements and records taken from remote sensing and stations, while non-flooded locations were taken randomly from locations with no previous flood record.

3.4.3 Performance assessment

In flood susceptibility analysis, both classification and regression metrics are adopted (Fig. 7). While classification metrics are used to identify flooded or non-flooded areas, the purpose of regression metrics is often omitted unless the reference target is a discharge hydrograph (Jahangir et al., 2019; Kia et al., 2012). Both types of metrics are used in few papers (e.g., Panahi et al., 2021; Khosravi et al., 2020). Because of the problem's setup, classification metrics are more reliable in the performance assessment. Following the considerations in Sect. 3.2.5, we selected as preferable metric AUC, also because of its frequent availability for flood susceptibility mapping. For all the papers with comparisons,

Table 3. Performance of the deep learning and comparison with reference models for flood inundation.

Case study size (km^2)	Deep learning		Comparison		Reference
	Model	Metric	Model	Metric	
2.2	MLP	Accuracy = 70 %	ML	Accuracy = 57 %	Amini (2010)
225	MLP	Accuracy = 86.1 %	SAM	Accuracy = 83.9 %	Li et al. (2016b)
3600	MLP + PSO	Accuracy = 81.6 %	PSO	Accuracy = 79.7 %	Li et al. (2016a)
5625	MLP + GA	Accuracy = 81 %	GA	Accuracy = 79.3 %	Li et al. (2015)
0.00016	CNN	Precision = 0.927	–	–	Hou et al. (2021)
0.01	CNN	Accuracy = 95.5 %	SVM	Accuracy = 87.4 %	Gebrehiwot et al. (2019)
0.9	CNN	IoU = 84 %	SVM+RBF	IoU = 83 %	Nogueira et al. (2017)
5.2	CNN	Accuracy = 92.4 %	RG	Accuracy = 91.8 %	Hashemi-Beni and Gebrehiwot (2021)
10	CNN	Accuracy = 96.4 %	RF	Accuracy = 87.3 %	Ichim and Popescu (2020)
59.3	CNN	F1 = 0.955	RF	F1 = 0.922	Peng et al. (2019)
59.3	CNN	F1 = 0.90	RF	F1 = 0.84	Wieland and Martinis (2019)
200	CNN	F1 = 0.975	–	–	Muñoz et al. (2021)
237	CNN	F1 = 0.90	NDWI	F1 = 0.70	Isikdogan et al. (2017)
10 895	CNN	F1 = 0.94	M1	F1 = 0.78	Kang et al. (2018)
23 300	CNN	F1 = 0.88	–	–	Liu et al. (2019)
25 000	CNN	F1 = 0.92	ATM	F1 = 0.71	Nemni et al. (2020)
31 450	CNN	Recall = 62.8 %	SVM	Recall = 25 %	Sarker et al. (2019)
4600	RNN + CNN	F1 = 0.77	CNN	F1 = 0.734	Dong et al. (2021)

ML is the maximum likelihood, SAM is the spatial attraction mode, PSO is the particle swarm optimization, GA is the genetic algorithm, SVM is the support vector machine, RBF is the radial basis function, RG is the region growing, RF is the random forest, NDWI is the normalized difference water index, and ATM is the automatic threshold model.

deep learning models consistently showed improved performances with respect to the reference models, with few exceptions (Table 4). Deep boost (DB) is a machine learning algorithm, based on deep decision trees (Cortes et al., 2014), which could slightly outperform MLP in a few works (Ahmed et al., 2021; Chakrabortty et al., 2021b). Combining optimization algorithms, such as particle swarm optimization, with MLPs, to improve the training, has a limited effect on the performance improvement (Kalantar et al., 2021; Ngo et al., 2018). Moreover, CNNs increase the performance with respect to traditional models more than MLPs. Fang et al. (2020a) show that encoding spatial sequentiality with LSTMs works slightly better than 1D CNNs and 3D CNNs; however, they avoid a comparison with 2D CNNs.

3.5 Deep learning for flood hazard

Flood hazard predicts the depth, velocity, and extent of floods. This application produces maps which evaluate, to a certain event, its maximum inundation (e.g., Guo et al., 2021; Berkhahn et al., 2019; Löwe et al., 2021) or how it evolves in time (e.g., Lin et al., 2020a; Zhou et al., 2021). While most studies consider the probability of different events using return periods (e.g., Kabir et al., 2020; Guo et al., 2021), there are a few papers which determine the water depth map for a single event (e.g., Hu et al., 2019; Chang et al., 2010). However, no papers were identified that predict the flow velocities. Since the simulation results are taken as ground-truth data for training, deep learning models for flood hazard mapping are used as surrogate models in place of numerical models.

The most-studied types of floods are river and urban floods. As regards the spatial scale, the models are carried out at local and regional scales. This is probably due to the computational burden of performing several simulations at larger scales to train the deep learning model.

3.5.1 DL architecture

The deep learning models are mainly based on MLPs and RNNs. In particular, RNNs are applied when a spatiotemporal estimation of the water depths is performed. CNNs were initially discarded but have been used more in recent years (e.g., Guo et al., 2021; Löwe et al., 2021; Kabir et al., 2020). Hu et al. (2019) and Jacquier et al. (2021) use an LSTM and an MLP, respectively, in combination with a reduced order modeling framework. In the first case, the DL model is applied on the reduced space, while in the latter DL is used as surrogate for the decomposition method.

3.5.2 Input and output data

The inputs are hyetographs, which represent the rainfall precipitation or intensity in time (e.g., Berkhahn et al., 2019; Kao et al., 2021; Guo et al., 2021), or hydrographs, which represent the discharge in time (e.g., Chu et al., 2020; Zhou et al., 2021; Lin et al., 2020a). Other inputs such as the DEM and the roughness coefficient, also used for numerical models, are sometimes considered as additional inputs (e.g., Guo

Table 4. Performance of the deep learning and comparison with reference models for flood susceptibility.

Case study size (km^2)	Deep learning		Comparison		Reference
	Model	Metric	Model	Metric	
27	MLP + ensemble	AUC = 0.847	RF	AUC = 0.821	Darabi et al. (2021)
132	MLP	R^2 = 0.802	–	–	Khoirunisa et al. (2021)
147	MLP + PSO	AUC = 0.98	MLP	AUC = 0.96	Kalantar et al. (2021)
207	MLP	R^2 = 0.82	SCS	R^2 = 0.71	Jahangir et al. (2019)
465	MLP	AUC = 0.917	PSO	AUC = 0.929	Chakrabortty et al. (2021b)
1128	MLP	AUC = 0.901	DB	AUC = 0.917	Ahmed et al. (2021)
1465	MLP	AUC = 0.960	SVM	AUC = 0.936	Tien et al. (2020)
1510	MLP	AUC = 0.970	SVM	AUC = 0.960	Ngo et al. (2018)
2600	MLP + AHP	AUC = 0.953	MLP + FR	AUC = 0.942	Costache et al. (2020)
4673	MLP	AUC = 0.93	DB	AUC = 0.96	Chakrabortty et al. (2021b)
5264	MLP + FR	AUC = 0.97	FR	AUC = 0.937	Popa et al. (2019)
12 050	MLP	AUC = 0.974	–	–	Ahmadlou et al. (2021)
132 000	MLP	R^2 = 0.98	–	–	Kourgialas and Karatzas (2017)
131	CNN	AUC = 0.90	RF	AUC = 0.78	Zhao et al. (2020)
605	CNN	AUC = 0.84	RNN	AUC = 0.82	Lei et al. (2021)
1543	CNN	AUC = 0.937	SVM	AUC = 0.883	Y. Wang et al. (2020)
12 000	CNN	AUC = 0.832	ANFIS	AUC = 0.70	Panahi et al. (2021)
1 649 195	CNN	AUC = 0.75	–	–	Khosravi et al. (2020)
90 016	CNN + FMV	AUC = 0.912	SVM-FMV	AUC = 0.898	Liu et al. (2021)
1543	LSTM	AUC = 0.965	3D CNN	AUC = 0.956	Fang et al. (2020a)

PSO is the particle swarm optimization, SCS is the soil conservation system model, SVM is the support vector machine, AHP is the analytic hierarchy process, RF is the random forest, FR is the frequency ratio, DB is the deep boost, ANFIS is the adaptive neuro-fuzzy inference system, and FMV is the fuzzy membership value.

et al., 2021; Chang et al., 2010; Huang et al., 2021b). Löwe et al. (2021) performed a forward selection to identify relevant topographic variables, showing that aspect and local depressions improve the model's prediction for urban floods.

The output is a water depth map. For the datasets, it is obtained via numerical models based on the 2D shallow water equations. 1D, 1D–2D, and 3D models are also used (Kao et al., 2021; Chang et al., 2010; Hu et al., 2019). The main reason why numerical models are used is to simulate events that have never occurred or have never been observed, such as floods with high return periods. Even though observed data were not employed, they could be used in future research to corroborate the transferability of such methods. When training only on the predictions of numerical models, the results of the deep learning models are limited in terms of accuracy by the numerical models' one, i.e., if the numerical model does not represent reality then neither will the DL model. Thus, when the model is deployed on real data, there may also be some generalization issues caused by the difference between the training and testing data. The inclusion of real measured data may thus also improve the accuracy with respect to numerical models.

3.5.3 Performance assessment

In flood hazard, regression metrics are used to evaluate the water depth, while classification metrics are used to evaluate the flood extent, as done for flood inundation (Fig. 7). While for flood susceptibility and inundation DL models were used to improve the performances, in flood hazard their main focus is to improve the speed, while still maintaining reasonably low errors with respect to the numerical predictions. This is highlighted in Table 5, for all papers which provide information on computational times of both numerical and deep learning models. However, the comparison of speed-up across different papers is often unrealistic, since it depends on the number of performed numerical simulations and on the type of numerical model. A similar consideration persists for the error scores, as they depend on the scale of the case study and on its resolution. Moreover, the real error of models trained on numerical results depends on that of the underlying numerical simulator. Hence, the latter must be reliable to have trustworthy predictions in real scenarios. A final remark regards the loss function employed in the training of the DL models. The minimization of the squared errors does not guarantee that the solution will have physical meaning. For flood hazard mapping, a possible solution is then to enforce the conservation of the mass or momentum equations by adding such terms in the loss function. This provides additional biases on the predicted solution and was shown to increase its performance in representing the numerical models (e.g., Zhang et al., 2021).

Table 5. Performance of the deep learning and comparison with numerical models for flood hazard.

Case study size (km^2)	Deep learning model	Numerical model	Comparison measure	Speed-up	Reference
0.6	MLP	2D	RMSE = 0.0013 m	1000×	Berkhahn et al. (2019)
7.7	MLP	2D	RMSE = 0.16 m	100×	Chu et al. (2020)
21	MLP + clustering	2D	RMSE < 0.3 m for 99.7 % of domain	–	Huang et al. (2021a)
31	MLP	1D–2D	MAE = 0.06 m	1000×	Chang et al. (2010)
92	MLP	2D	RMSE < 0.3 m for 82 % of domain	–	Lin et al. (2020a)
92	MLP	2D	MSE < 0.2 m^2 for 91 % of domain	–	Lin et al. (2020b)
–	MLP + ROM	2D	RE = 2.8 %	33×	Jacquier et al. (2021)
10	CNN	2D	MAE = 0.0012 m	–	Hosseiny (2021)
10.5	CNN	2D	MAE < 1 m for 93 % of domain	2000×	Guo et al. (2021)
14.5	CNN	2D	RMSE = 0.11 m	38×	Kabir et al. (2020)
72	CNN	2D	RMSE = 0.22 m	–	Yokoya et al. (2020)
400	CNN	2D	RMSE = 0.08 m	–	Löwe et al. (2021)
18.5	LSTM + ROM	3D	RMSE = 0.01 m	1500×	Hu et al. (2019)
271	LSTM	2D	RMSE = 0.08 m	–	Kao et al. (2021)
1479	RNN	2D	RMSE = 0.056 m	21×	Zhou et al. (2021)

ROM is the reduced order modeling, and MSE is the mean squared error.

4 Knowledge gaps

We identified knowledge gaps regarding the applications in flood management, usability, generalization, modeling limitations, and data availability. Some other minor gaps were shown in the previous section. Based on these gaps, future research directions are proposed in Sect. 5.

4.1 Flood applications and usability

Deep learning has proven useful for assessing flood-prone areas from the location of past events, identifying flooded areas from remote sensing images, and working as a surrogate model for numerical simulations. However, there are still several other applications within this field that could benefit from deep learning models. In particular, we address two flood management applications, i.e., flood risk and real-time flood warning. We also define two desired types of maps, i.e., flood arrival time maps and probabilistic hazard maps. Then, we discuss dam and dike breach flood events.

Flood risk combines the probability that a certain event occurs with the associated consequences, such as economic impacts or loss of life. The expected annual loss is a common measure obtained from flood risk assessment and depends on (i) flood hazard, given by event-specific flood characteristics, such as water depth and flow velocity, (ii) exposure, related to the elements at risk, such as buildings and critical infrastructure, and (iii) vulnerability, i.e., the inability of a system to withstand the effects of the event, given, for example, by intensity–damage curves. Flood risk maps are obtained by combining flood hazard maps with damage models. Other approaches are based on MCDA, since the exact flood magnitude and damage are uncertain (de Brito and Evers, 2016). This is done by incorporating various factors that determine flood risk, such as hazard, the performance of defenses, topography, and exposure. However, MCDA is based on expert knowledge and is thus subjective. DL models solve this issue and can also yield a higher accuracy, as shown for flood susceptibility mapping. Thus, DL-based approaches could provide alternative methods for assessing flood risk. In addition to the inputs used for flood susceptibility, such as elevation and land use, flood risk mapping may require also other inputs such as population density, spatial estimates of economic value, and building types. Up until now, only Chen et al. (2021) combined DL and flood risk

assessment. They showed that ML and DL approaches can estimate flood risk at regional scale but do not compare their results against other methods, such as MCDA. One drawback of their approach is that the resulting maps were qualitative, while quantitative results should be preferable for risk assessment.

Real-time flood warning is another application that has not been widely addressed. This is needed by local authorities to inform the public of when and where a flood may occur. While several papers mention real-time prediction, most can be used only after the event has occurred, since they require as input the complete hyetograph or hydrograph of the event. There are a few examples based on RNNs which could forecast floods in near-real time using sensors (Kao et al., 2021) and rainfall distribution (Dong et al., 2021). However, few situations are covered and, thus, more research should focus on filling this gap. An alternative method is to predict the rainfall in real time and then retrieve the corresponding water depth map by using a similarity measure on a large dataset of previous simulations (Chang et al., 2020). However, such a solution may be challenging because of the large storage requirements. Using DL for surrogate modeling instead showed substantial speed improvements, thus allowing for real-time simulations and forecasts. Similar achievements have already been obtained for rainfall nowcasting, where the deep learning models can accurately forecast the near-future rainfall (e.g., Shi et al., 2015; Ravuri et al., 2021).

Arrival time maps estimate the time employed by a flood to reach a certain water depth threshold. They can encode both spatial and temporal information in the same map. So, for a practitioner, they carry at one place detailed information not only on where to intervene but also when to execute mitigation measures. Despite these promises, they have seldom been used in flood management; consequently, they have also not been exploited with DL methods. Using DL for arrival map estimation may be a promising direction to identify critical infrastructure and set up corresponding evacuation plans in real time. This is because DL has shown the potential for surrogate modeling (see Table 5) and because arrival maps can be obtained from flood hazard maps taken over different time intervals of a flood event. This application may be particularly important for exceptional flood events, such as dike breaches and dam breaks, where little forecast can be made until a failure initiates (Yakti et al., 2018).

Probabilistic hazard mapping captures the model uncertainty related to its inputs and outputs. As pointed out by Di Baldassarre et al. (2010), uncertainties can result in deterministic maps which are only spuriously accurate. But probabilistic maps can account for the uncertainties by assigning a probability of flooding to each domain element. This analysis is generally carried out with probabilistic methods such as Monte Carlo simulations (e.g., Papaioannou et al., 2017). However, since they require a vast amount of simulations, only simpler numerical models are used. DL models could be used as surrogates to speed up computation and im-

prove the accuracy of the simpler models. Nonetheless, brute force simulations, such as Monte Carlo, may require up to hundreds of thousands of simulations to obtain a satisfactory measure of the uncertainty (Kalos and Whitlock, 2009). Thus, we need models that can intrinsically work with probabilistic input distributions of parameters.

Dam break and dike breach floods concern a relevant category of flood events that has been poorly approached with deep learning models. The motivation is probably related to the rarity of such events and the complexity of the phenomena. However, their catastrophic and unexpected effects make their modeling necessary in several situations. Moreover, the effect of flood defenses' failure is often disregarded, also because the location and modality of possible failures are uncertain. A common way to include the failure of structures is to investigate all possible combinations of locations and boundary conditions, but it can be constrictive both for time and storage capacities. Probabilistic hazard mapping may be a relevant application to include the uncertainty in the failure probability of the flood defense (Domeneghetti et al., 2013).

4.2 Generalization

Generalization refers to the capacity of a model to extrapolate from a training dataset into unseen testing data. This means that a DL model can correctly predict scenarios unused in its development. This property is particularly relevant because training requires data, model setup, and time. In the context of flood modeling, there are two main generalization objectives: (i) boundary conditions, e.g., different rainfall events, and (ii) topographical changes, i.e., different case studies. However, the transference between different areas is challenging for DL models because of the difference in input and output data. In fact, except for flood inundation mapping, most reviewed papers focused on generalizing different boundary conditions (e.g., Guo et al., 2021; Berkhahn et al., 2019). Instead, only a few papers tested the model on areas not considered during training. Löwe et al. (2021) could generate flood hazard maps for unseen areas within the same study region as the training dataset, as there was little variability in inputs and outputs. Zhao et al. (2021c) instead pretrained a model for flood susceptibility on an urban area and then used it for another similar area. They showed that pretraining improves predictions with respect to a model trained from scratch, both in cases of low and high data availability. These works show that such approaches are in their infancy and have been tested on limited datasets. A DL model which cannot generalize to new areas has to be trained every time for a new study case. Thus, it may have limited advantages over a hydraulic model, since it requires more effort, data, and time. Instead, a general DL model which can generalize to new areas could emphasize the advantages over numerical models. This concept was experimented also for rainfall–runoff modeling where DL models outperformed state-of-

the-art alternatives in the prediction of ungauged basins in new study areas (Kratzert et al., 2019b).

4.3 Modeling limitations

Complex interactions with the natural and built environment, such as dikes or buildings, are difficult to include in deep learning models. Kabir et al. (2020) showed that flood defenses can be included if they are present in the simulations used for training and testing. However, no solution presented so far can directly include new flood defenses in it. Building can be statically included as well in the DEM (e.g., Löwe et al., 2021), but bridges and other hydraulic structures that influence the behavior of the floods may be harder to include, due to their strong influence on the flow path.

4.4 Data availability

Deep learning models usually require large quantities of data to achieve good performances. While simulations can provide potentially limitless data, observed data are scarce and depend on the study area. Simulations may also encounter instability issues depending on the numerical schemes and study area. Remote sensing has provided large quantities of data since its vast development in the past decades, but satellite data are still limited by their frequency of observations and dependency on favorable meteorological conditions. Also, UAVs cannot cover wide areas at once. Precipitation and water depth data are available only in a few locations where the measuring stations are present. Thus, new data sources are needed to overcome these limitations.

Another issue, which emerges also from Sect. 3.2.5, is the lack of a unified framework to compare different approaches with each other. This can be achieved by creating flood-based benchmark datasets for each mapping application. For flood inundation, some datasets have been already used across different works (e.g., Bonafilia et al., 2020). However, works on both flood susceptibility and hazard mapping consider different datasets, focusing on different geographic areas or flood types. One possibility could then be to unify different case studies in a single dataset, for each application, allowing us to assess the validity of a model more objectively. For flood susceptibility, case studies with the same input availability could be merged in a dataset with many flood types, scales, and geographical areas. A similar reasoning could be made for flood hazard mapping, selecting, for each case study, initial and boundary conditions for specific return periods.

5 Future research directions

The present review shows that flood practitioners still need to be up to date with the latest and most successful deep learning models. We suggest that the outstanding identified issues can be approached by resorting to deep learning state-of-the-art advancements to our field. As such, we propose future re-
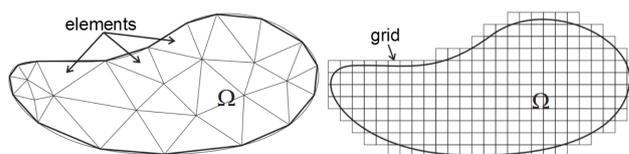

Figure 8. The irregular geometrical structure of the mesh allows capturing information in a more efficient way than regular grids by following the properties of the underlying system (figure taken from Ferreira et al., 2015).

search directions to transfer this knowledge and address the above-identified gaps.

5.1 Mesh-based deep learning

Current deep learning models lack generalization across different case studies, meaning that they can work exclusively for a specific purpose or area. They also cannot represent complex interactions with the natural and built environment. Both issues may depend on the regular grids used in the reviewed papers, which are unable to follow the geometric properties of irregular inputs, as illustrated in Fig. 8. Hence, the model cannot exploit many data patterns, ultimately limiting its generalizability and, for the same motivation, being unable to account for the irregular geometrical structures. Unstructured meshes may solve this problem by discretizing the domain more flexibly (Mavriplis, 1997). A mesh is a structure composed of a collection of nodes, edges, and faces used to discretize a continuous domain. Meshes are commonly used for numerical simulations in many physical systems (e.g., Ferraro et al., 2020; Bomers et al., 2019). Their flexible definition allows to increase the resolution where needed and coarsen it otherwise, ultimately decreasing the computational time and improving efficiency (Candy, 2017). Moreover, they are equivalent to regular grids if the mesh is structured. Thus, the following principles could also be transferred to rasters, if needed. Unstructured meshes, nonetheless, inherit similar problems as those typical of numerical models, such as mesh generation and the need to explicitly define how each node is connected. Standard DL models, such as CNNs, cannot be applied on meshes. There are currently several lines of work which, instead, can use meshes as a learning framework. They are here referred to as mesh-based neural networks. The two highly promising mesh-based approaches for flood applications are geometric deep learning and physics-based deep learning.

5.1.1 Geometric deep learning

Geometric deep learning provides a generic framework to work with any type of data by enforcing symmetries with respect to transformations, such as translations and rotations (Bronstein et al., 2017). Symmetries result in inductive biases, which address the curse of dimensionality by decreas-

https://doi.org/10.5194/hess-26-4345-2022

Hydrol. Earth Syst. Sci., 26, 4345–4378, 2022

ing the required training data (e.g., R. Wang et al., 2020) and enabling the processing of different data types, such as meshes. From a flooding perspective, symmetries can be understood and motivated by referring to the example in Sect. 2.2.1. For instance, analogous to the translation, the rotation of a domain should result in an equivalent rotation of the predictions. Among the several geometric deep learning models which can work with meshes, graph neural networks (GNNs) are the most developed ones. Graphs are structures defined by a set of nodes and edges and can be considered as the underlying skeleton of a mesh. GNNs allow us to model data on graphs by considering how the elements are connected (Wu et al., 2021; Gama et al., 2020). They take as input the information encoded in the nodes, in the edges, and in the graph structure, and then process it with neural networks in a similar manner to the CNNs and RNNs with grid elements and sequential data, respectively. For example, nodes can carry information on the elevation of a point or its boundary conditions, while edges may encode the spatial distance between nodes. Several variations in GNNs exist that give more importance to certain parts of the data by weighting information from different neighbors (Wu et al., 2021; Isufi et al., 2021). There already exist promising works which simulate fluid dynamics with mesh-based GNNs, with increased generalization, accuracy, and stability, with respect to CNNs (e.g., Pfaff et al., 2020; Lino et al., 2021). However, GNNs consider only pairwise geometrical properties as connections between nodes, thus neglecting the mesh structure. Recent developments focused on extending the GNN framework to include it. Mesh convolutional neural networks adapt GNNs to include a representation of the local geometry, which preserves the angles between edges (De Haan et al., 2020; Zhou et al., 2020). Simplicial neural networks (Yang et al., 2021; Ebli et al., 2020) and cell complex neural networks (Bodnar et al., 2021; Hajij et al., 2020), instead, generalize GNNs to higher-order structures. They can also consider information on triangular and polyhedral elements, which can represent, for example, a flooded area or a volume. This inclusion of the mesh properties in such approaches may further enhance the power of GNNs. Even though they are still in their infancy, their potential for learning on meshes could reveal to be useful also for flood modeling in future research.

5.1.2 Physics-based deep learning

While promising, the aforementioned approaches ignore any underlying physical laws present in flood modeling and let the model figure them out. But these physical laws provide additional inductive biases; hence, we could include them in modeling to enhance the performance. Physics-based neural networks and neural operators are approaches that account for them.

Physics-informed neural networks (PINNs) employ physical laws to constrain the model solution (Raissi et al., 2019). The idea is to parameterize a partial differential equation (PDE) solution with a neural network, while keeping the same physical formulation. Then, each partial derivative in the equations is determined via automatic differentiation. Many works have shown the capabilities of PINNs to follow the underlying PDEs in fluid dynamics (e.g., Mao et al., 2020; X. Yang et al., 2019). This is relevant also in flood modeling where PDEs such as shallow water equations or the Navier–Stokes equations are employed (e.g., Mahesh et al., 2022). However, PINNs can only be trained for a specific boundary condition (e.g., a specific rain event) and can subsequently only simulate that specific event (Kovachki et al., 2021).

Neural operators, instead, can learn the mappings between function spaces, i.e., they learn a whole family of equations (Kovachki et al., 2021). In other words, they can approximate any differential operator. Moreover, since neural operators learn a mapping between infinite-dimensional spaces, they are invariant with respect to the chosen discretization. Thus, their solution is transferable to any mesh resolution. While many approaches have been proposed, such as DeepONets (Lu et al., 2019) or multipole graph neural operator (Li et al., 2020), Fourier neural operators (FNOs) have currently achieved the best results (Li et al., 2021). In general, the idea is to extract features from the input function, process them in the function space, and, finally, map them to the output function. In FNOs, the function space is given by the Fourier space, which allows us to use fast Fourier transforms, providing faster approximations of the integral operator. Results show that FNOs improve the speed of several PDEs by up to 3 orders of magnitude. Jiang et al. (2021) used FNOs for simulating sea surface height, showing increased performance with respect to CNNs and noticeable speed-up compared to the numerical simulator. Consequently, they could also be used in flood management to overcome computational speed limitations while preserving the underlying physics, allowing also for a more reliable real-time flood warning. Thanks to the inductive biases given by the physical laws, both physics-based neural networks and neural operators also require less data.

5.2 Probabilistic deep learning

Uncertainties in floods are often determined via probabilistic hazard mapping. These maps show the inundation depths and extents together with their confidence intervals and are traditionally obtained with Monte Carlo simulations (e.g., Domeneghetti et al., 2013). To avoid brute-force simulations and provide uncertainty guarantees, certain deep learning models can consider uncertainty in the model inputs. An example of these models is deep Gaussian processes (DGPs). DGPs are models composed by the stacking of Gaussian processes (GPs), in a similar fashion to neural networks (Damianou and Lawrence, 2013). A GP is a collection of random variables whose joint distribution is a Gaussian (Rasmussen, 2003). They benefit from the properties of normal distribu-

tions, and thus, their output can be obtained analytically. The advantage of DGPs over GPs is that they can extract patterns in data better, thanks to their increased complexity. DGPs can determine the distribution of the output and could, therefore, be used in probabilistic hazard modeling to determine the range of variation in the predicted flood hazard map. No example of DGPs used for flood mapping exists yet. However, GPs have been used for the statistical estimation of the correlation between flooding and sea level rise (Vandenberg-Rodes et al., 2016).

Along with those related to the model's input, uncertainties are also present in the model's prediction. To account for this kind of uncertainty, we can use Bayesian neural networks (BNNs). BNNs are models with stochastic components trained using Bayesian inference. They assign prior distributions to the model parameters to provide an estimate of the model's confidence on the final prediction (Blundell et al., 2015). If, for different parameter sampling, the output is unvaried, then the model has a good confidence on the prediction, and vice versa, if different parameters give different results. Jacquier et al. (2021) used BNNs to determine the confidence intervals in flood hazard maps, providing a measure of the model's reliability.

5.3 Data augmentation

Even though remote sensing and measuring stations provide noticeable amounts of data, several parts of the world still lack enough data to deploy deep learning models. New satellite missions and added sensor networks throughout the world increasingly provide new data sources (e.g., van de Giesen et al., 2014). But here we focus on how DL itself can be one solution for data scarcity.

The flexibility of DL partially overcomes data scarcity by facilitating the use of a wider variety of data sources. For instance, several papers already employ cameras to detect floods and measure the associated water depth (e.g., Vandaele et al., 2021; Jafari et al., 2021; Moy De Vitry et al., 2019). Structural monitoring with cameras can provide reliable data sources where they were previously hard to obtain, such as in urban environments. Social media information can also be used to identify flood events and flooded areas, via tweets or posted pictures (e.g., Rossi et al., 2018; Pereira et al., 2020). In this case, the information's validity and reliability must be considered before its use for real application. Moreover, the heterogeneity of the sources of these data needs to be carefully taken into account when deploying a DL model.

Another approach can be to generate artificial data to supplement scarce data. This can be done using generative adversarial networks (GANs), which create new data from a given dataset (Goodfellow et al., 2014). GANs are composed of two neural networks, named a generator and discriminator, whose purpose is, respectively, to generate new data and to detect if the given data are real or fake. A trained GAN can produce new fake but plausible data, facilitating data augmentation, i.e., providing more training samples. Interesting applications of GANs could overcome some limitations of satellite data (Lütjens et al., 2020, 2021), predict flood maps (Hofmann and Schüttrumpf, 2021) or meteorological forecasts (Ravuri et al., 2021), and create realistic scenarios of flood disasters for projected climate change variations (Schmidt et al., 2019). GANs could also be used to generate a plausible urban drainage system or topography for cities that do not have any sewer construction plan or in areas where only low-resolution data are available (e.g., Fang et al., 2020b).

However, GANs are difficult to train (Goodfellow, 2016). Variational autoencoders (VAEs) are another type of generative model which can overcome this issue. Different from standard autoencoders, VAEs model the latent space with probability distributions that aim to ensure good generative properties to the model (Kingma and Welling, 2013). Once the model is trained, new synthetic data can be generated by taking new samples from the latent distributions. Nonetheless, because of the model's definition, the predictions are less precise than GANs. As such, VAEs and GANs offer a tradeoff between the reality of the prediction and the availability of training data.

6 Conclusions

This paper presented a review of current applications of deep learning models for flood mapping. The chosen search criteria yielded a total of 58 papers published between 2010 and 2021. From our analysis, we conclude that there are common patterns across works that can be summarized as follows:

- Flood inundation, susceptibility, and hazard mapping were investigated using deep learning models. Flood inundation considers, as the main data, images of floods, mostly taken via satellite. The main and most accurate deep learning models were CNNs. In flood susceptibility, deep learning models consider several inputs, with the most important being slope, land use, aspect, terrain curvature, and distance from the rivers. The main deep learning model used were MLPs, often in combination with other statistical techniques, although CNNs provided more accurate results. Deep learning for flood hazard mapping generally involves developing surrogates of numerical models that estimate water depths in a study area. For this application, there are no deep learning model preferences. However, RNNs are preferable for spatiotemporal simulations.

- MLPs and CNNs were the most common type of deep learning model considered in flood mapping, while RNNs were used less often. To overcome their lack of inductive biases and achieve good accuracy, MLPs are often coupled with other statistical techniques. On the

other hand, thanks to their spatial and temporal inductive biases, CNNs and RNNs were found to regularly outperform other models.

- Most papers dealt with river and urban floods, while only a few works described applications for flash, coastal, and dam break floods. Case studies were mainly addressed at local or regional scales, arguably due to the availability of high-resolution data. Conversely, the community should further investigate the suitability of deep learning models for flood applications at larger scales.

- Concerning the development data, we found that models producing susceptibility and inundation maps rely on the availability of real flood observations. Instead, DL-based surrogate models for hazard mapping require target data from numerical simulations.

In terms of comparison with traditional and machine learning approaches, we found the following:

- Regardless of the application, results show that deep learning solutions outperform traditional approaches and other machine learning techniques.

- Deep learning models used for surrogate modeling provide significant speed-up (up to 3 orders of magnitude) while maintaining sufficient accuracy.

This review did not consider works featuring ML methods alone. Therefore, further research is needed to thoroughly compare ML against DL methods, especially with respect to explainability, generalization ability, and data requirements. This review also outlined several knowledge gaps which can be addressed via deep learning to improve the state of the art of flood mapping. To solve these gaps, we proposed the following possible solutions based on recent advances in fundamental machine learning research:

- Flood risk could be addressed in a similar manner to that of flood susceptibility by using physical and economical characteristics to obtain a risk map. Flood arrival time maps can provide both spatial and temporal information of a flood event and may be obtained similarly as for flood hazard maps.

- Current deep learning models struggle to generalize across different case studies and regions, implying that a new model must be created each time. Further problems occur when modeling the complex interactions with the natural and built environment. While some of the reviewed papers provide initial suggestions to tackle these issues, the community should invest more efforts in this direction. A possible solution to these problems is to use novel deep learning architectures that include meshes as learning frameworks. Mesh-based neural networks, such as graph neural networks and neural operators, can

consider arbitrarily shaped domains and thus provide the required flexibility to generalize across case studies and model the effects of complex interactions.

- Physics-based deep learning provides a reliable framework for flood modeling, since it considers the underlying physical equations. Probabilistic hazard mapping can take advantage of deep Gaussian processes or Bayesian neural networks to determine the uncertainties associated with the model and its inputs.

- Deep learning necessitates large quantities of data which are difficult to collect in several areas of the world. New data sources such as camera pictures and videos or social media information can potentially be used thanks to deep learning models. Moreover, generative models, such as GANs and VAEs, can be employed to produce synthetic data for such data-scarce regions, based on training data collected elsewhere.

While our review draws insights for future research directions from the machine learning literature, further understanding may emerge from a broader review, including deep learning applications across other water-related and natural-hazard-related fields, and featuring a bibliometric analysis (Donthu et al., 2021; Fazeli-Varzaneh et al., 2021). This approach may facilitate cross-fertilization between sister disciplines, especially with respect to the successful implementation of advanced deep learning methods for spatial analysis. We expect deep learning to be a promising tool to improve and speed up flood mapping. Nonetheless, deep learning models are black box models, meaning that the underlying operations are unknown. Thus, their deployment in real emergencies has to be done with caution. As deep learning for flood mapping is still novel, we advise its use in critical situations to be always validated by traditional models and expert knowledge until robust and corroborated models are available. The above concern highlights the main challenge that deep learning models for flood management need to face. However, deep learning models are still in their infancy and carry the large potential to aid researchers for many applications, especially where traditional models cannot provide sufficient accuracy or speed. In particular, deep-learning-based flood mapping approaches could provide an added value for regions with limited data or limited resources to invest in setting up time-consuming hydraulic models.

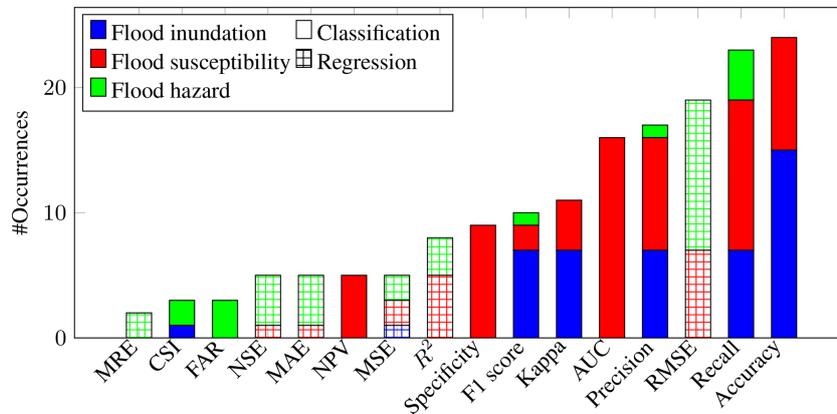## Appendix A: Comparison metrics

Figure A1. Distribution of the comparison metrics in the reviewed papers per type of application. AUC is the area under the ROC curve, CSI is the critical success index, FAR is the false alarm ratio, MAE is the mean average error, MRE is the mean relative error, MSE is the mean squared error, NPV is the negative predictive value, NSE is the Nash–Sutcliffe efficiency, RMSE is the root mean squared error, and R^2 is the coefficient of determination.

Appendix B: Flood susceptibility inputs
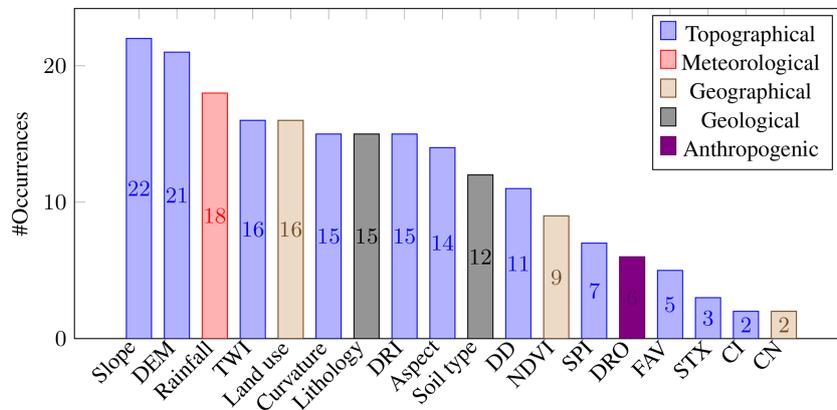
Figure B1. Distribution of the inputs for flood susceptibility for the 23 reviewed papers. The inputs are categorized in topographical, meteorological, geographical, geological, and anthropogenic factors. Inputs which were considered only once were discarded from this graph. CI is the convergence index, CN is the curve number, DD is the drainage density, DEM is the digital elevation model, DRI is the distance from rivers, DRO is the distance from roads, FAV is the flow accumulation value, NDVI is the normalized difference vegetation index, SPI is the stream power index, STX is the sediment transport index, and TWI is the topographic wetness index.

Appendix C: Reviewed papers

Table C1. A brief description of the reviewed papers, following the same ordering as in Table 2.

Reference	Brief description
Li et al. (2015)	An MLP combined with genetic algorithm is used for super-resolution mapping of wetland inundation. Images are taken from Landsat remote sensing for two areas in China and Australia.
Li et al. (2016a)	An MLP optimized with particle swarm optimization is used for super-resolution mapping. Images are taken from Landsat remote sensing for two areas in China and Australia.
Nogueira et al. (2017)	Satellite image patches, collected from eight flooding events, are classified in flooded and non-flooded areas using a CNN.
Gebrehiwot et al. (2019)	A set of images taken from unmanned aerial vehicles is used for flood inundation mapping using a CNN. The dataset is composed of three study areas in North Carolina, USA, with a total of 100 images.
Ichim and Popescu (2020)	A combination of five different CNN-based models is used for flood inundation mapping. A set of 2000 images derived from UAVs in a Romanian rural region is classified into flooded, vegetation, and non-flooded areas.
Hou et al. (2021)	A flood propagation experiment in a small-scale laboratory is used to develop a CNN model for inundation mapping. The dataset is composed of a sequence of 1930 images detected from cameras.
Hashemi-Beni and Gebrehiwot (2021)	An encoder–decoder CNN is developed for flood inundation mapping. The study area is the same as Gebrehiwot et al. (2019).
Wieland and Martinis (2019)	An encoder–decoder CNN model is trained to identify five classes (water, land, snow, shadow, and cloud) from Landsat and Sentinel-2 satellite images.
Kang et al. (2018)	A CNN model is developed for flood detection from satellite images for three study areas in China.
Sarker et al. (2019)	A CNN model is trained to predict flood extent from Landsat satellite images across Australia.
Nemni et al. (2020)	A CNN model is trained to predict flood extent from Sentinel-1 synthetic aperture radar (SAR) imagery. The United Nations Satellite Centre (UNOSAT) flood dataset, which covers flood events from Africa and East Asia, is used for the model development.
Isikdogan et al. (2017)	A CNN model is trained to identify five classes (water, land, snow, shadow, and cloud) from Landsat satellite images.
Amini (2010)	High-resolution images are classified into five categories with an MLP. The study area is located in an Iranian city.
Li et al. (2016b)	An MLP is developed for super-resolution mapping. Images are taken from Landsat remote sensing for two areas in China and Australia.
Peng et al. (2019)	A CNN which considers pre- and post-flood satellite images is used for urban flood detection. The study area is in Texas, USA, and considers two hurricane events.
Dong et al. (2021)	A combination of 1D CNN and GRU is used for prediction of flood inundation in urban area in Texas, USA. The dataset is composed by a channel sensor network. Given information on the temporal evolution of water depth and precipitation, for each sensor, the model predicts which nodes will be flooded.
Liu et al. (2019)	A CNN which considers pre- and post-flood SAR images is used for coastal flood detection. The study area is in Texas, USA, and considers six pairs of satellite images for a hurricane event.
Muñoz et al. (2021)	A CNN model is developed for compound flood mapping for the Atlantic coast of USA. The model takes as input Landsat and SAR satellite data, along with a DEM of the study areas, and combines them with data fusion. The model distinguishes then between several categories, such as flooded areas and vegetation.

Hydrol. Earth Syst. Sci., 26, 4345–4378, 2022

https://doi.org/10.5194/hess-26-4345-2022

Table C1. Continued.

Reference	Brief description
Syifa et al. (2019)	An MLP which considers pre- and post-flood satellite images is used for flood inundation mapping after a dam break in Brazil.
Khoirunisa et al. (2021)	An MLP is developed to obtain a flood susceptibility map for a Taiwanese city, using nine flood conditioning factors. The model is then compared with SOBEK, a 2D hydraulic model. The data are given by 307 flood events collected between 2015 and 2019.
Jahangir et al. (2019)	An MLP is employed in a river basin in Iran to predict discharges, using seven flood conditioning factors. The discharges are then converted into a flood susceptibility map via a GIS software. The data are given by hydrometric stations in the basin.
Ahmadlou et al. (2021)	Several MLP architectures are presented for the development of two flood susceptibility maps in Iran and India using, respectively, 9 and 12 flood conditioning factors. The datasets are composed of 147 and 300 flood events, respectively.
Popa et al. (2019)	A combination of MLP with a frequency ratio is used to determine flash flood and river flood potential indexes for a Romanian catchment, using, respectively, 14 and 13 flood conditioning factors. The datasets are composed, respectively, of 168 and 172 flood locations.
Kia et al. (2012)	An MLP is employed in a river basin in Malaysia to predict discharges, which are then converted into a flood susceptibility map via GIS. The model considers seven flood conditioning factors.
Ahmed et al. (2021)	There are two MLP models developed to obtain a flood susceptibility map for a catchment in Bangladesh, using 12 flood conditioning factors. The data are given by 521 flood events collected during 2019.
Chakrabortty et al. (2021b)	There are two MLP models developed to obtain a flood susceptibility map for a catchment in India, using 15 flood conditioning factors. The models are then used estimate future flood susceptibility scenarios by considering rainfall variations due to climate change.
Saeed et al. (2021)	An MLP model is developed to obtain a flood susceptibility map for a basin in Pakistan, using nine flood conditioning factors.
Y. Wang et al. (2020)	There are three CNN models, based on 1D, 2D, and 3D convolutional layers proposed for a region in China using 13 flood conditioning factors. The dataset is based on 108 historical flood events. The model is also compared with support vector machine.
Khosravi et al. (2020)	A national-scale flood susceptibility map is developed for Iran, using a CNN. The model uses 10 flood conditioning factors and a dataset composed of 2769 flood events.
Fang et al. (2020a)	An LSTM model is developed for flood susceptibility in a Chinese county. The model uses 11 flood conditioning factors and a dataset of 108 flood events. The model is also compared with MLP, 1D CNN, and 3D CNN.
Tien et al. (2020)	An MLP, based on nine flash flood conditioning factors, is developed for a province in Vietnam. The dataset contains 732 flood events.
Ngo et al. (2018)	A combination of MLP with the firefly algorithm optimization technique is proposed for flash flood susceptibility in a Vietnamese region. The model considers 12 flood conditioning factors and a dataset of 654 flooded areas.
Costache et al. (2020)	There are two MLP models, combined with analytical hierarchy process and frequency ratio, respectively, developed for flash flood susceptibility mapping in a catchment in Romania. The model considers 10 flood conditioning factors and 178 flooded locations.
Chakrabortty et al. (2021a)	There are two MLP models developed to obtain a flash flood susceptibility map for a catchment in India, using 14 flood conditioning factors.
Kourgialas and Karatzas (2017)	A national-scale flash flood susceptibility map is obtained with an MLP for Greece. The model considers seven flood conditioning factors and a dataset of 600 flood events.

Table C1. Continued.

Reference	Brief description
Panahi et al. (2021)	There are two models, CNN and RNN, developed for flash flood susceptibility in a province in Iran. The model considers nine flood conditioning factors and a dataset of 143 flood events.
Liu et al. (2021)	A 1D CNN model combined with fuzzy membership value is proposed for flood susceptibility mapping for a basin in China, using nine flood conditioning factors. The dataset is based on 485 flood locations. The model is also compared with a support vector machine.
Darabi et al. (2021)	An MLP ensemble model is used for urban flood susceptibility mapping in a Iranian city. The model uses six flood conditioning factors and a dataset of 118 flooded locations.
Kalantar et al. (2021)	There are three MLP models, one of which optimized with particle swarm optimization, developed for flood susceptibility mapping in a urban catchment in Australia. The model considers 13 flood conditioning factors and 128 flooded locations.
Zhao et al. (2020)	There are two CNN models used to assess urban flood susceptibility for a Chinese catchment. The model considers nine flood conditioning factors and 202 flooded locations. The model is also compared with support vector machine and random forest.
Zhao et al. (2021c)	A pre-trained CNN model, taken from Zhao et al. (2020), is deployed in two urban areas, with data-rich and data-scarce scenarios.
Lei et al. (2021)	There are two models, CNN and RNN, developed for urban flood susceptibility in Seoul, South Korea. The model considers 10 flood conditioning factors and a dataset of 295 flood events.
Chu et al. (2020)	An MLP is used to estimate the water depth for a river reach in Australia. The dataset is generated using a 2D hydrodynamic model, TUFLOW, and considers 10 flood events.
Lin et al. (2020b)	An MLP is used to estimate the water depth and extent for a river reach in a German city. The dataset is generated using a 2D hydrodynamic model, HEC-RAS, and considers 180 flood events.
Lin et al. (2020a)	An MLP is used to forecast water depth and flood extent for a river reach in a German city. The dataset is the same as in Lin et al. (2020b).
Huang et al. (2021a)	There are two models, MLP and RNN, developed for water depth estimation for a USA river. Before being given as input to the models, the data are clustered based on slope, drainage area, and hydrologic length. The dataset is generated using a 2D hydrodynamic model and considers 20 flood events.
Xie et al. (2021)	There are three MLPs used to estimate the water depth for a river reach in Australia (also considered by Chu et al. (2020)). The dataset is generated using a 2D hydrodynamic model, TUFLOW, and considers nine flood events.
Jacquier et al. (2021)	A combination MLP with reduced-order modeling is applied to a river and a dam break flood simulations. Moreover, the model provides uncertainty estimates using ensembles and Bayesian neural networks. The dataset is obtained using a 2D hydrodynamic model, CuteFlow.
Kabir et al. (2020)	A 1D CNN model is used to predict a flood hazard map, from a discharge hydrograph, for a river reach in England. The dataset is generated using a 2D hydrodynamic model, LISFLOOD-FP, and considers 10 flood events. The model is also compared with a support vector regression.
Hosseiny (2021)	An encoder–decoder CNN is used for flood mapping in a river reach in the USA. The dataset is generated using a 2D hydrodynamic model, iRIC, and considers seven flood events.
Guo et al. (2021)	An encoder–decoder CNN is used for flood mapping in three catchments, i.e., two in Switzerland and one in Portugal. The dataset is generated using a cellular automata flood model, CADDIES, and considers 18 flood events.

Table C1. Continued.

Reference	Brief description
Zhou et al. (2021)	An LSTM model is used to predict the temporal evolution of water depth in few representative locations along an Australian river. The water depths are then interpolated to obtain a flood hazard map. The dataset is generated using a 2D hydrodynamic model, TUFLOW, and considers 74 flood events.
Kao et al. (2021)	A stacked encoder–decoder LSTM is used to determine flood hazard maps in time from precipitation. The study area is located in Taiwan. The dataset is generated using a storm water management model, HEC-1, plus a 2D hydrodynamic model, and considers 24 flood events.
Chang et al. (2010)	An MLP is developed to forecast 1 h ahead flood hazard maps in a Taiwanese county. The data are also pre-processed by clustering. The dataset is generated using a storm water management model, HEC-1, plus a 2D hydrodynamic model, and considers 120 flood events.
Yokoya et al. (2020)	An encoder–decoder CNN is used for flash flood and debris flow mapping in Japan. The dataset is generated using a 2D hydrodynamic model coupled with a debris flow module and considers 160 flood events. The model is then trained to estimate flood depths and debris flows from pre- and post-flood event images.
Berkhahn et al. (2019)	An MLP is used to predict maximum water levels for two urban areas. The dataset is generated using a coupled sewer–surface model, HE 2D, and considers 64 flood events.
Löwe et al. (2021)	An encoder–decoder CNN is used for flood mapping in a urban area in Denmark. The dataset is generated using a 2D hydrodynamic model, MIKE 21, and considers 53 flood events.
Hu et al. (2019)	An LSTM model is developed to simulate a tsunami in Japan. The model is trained in a lower dimensional space to reduce the problem's complexity. The dataset is obtained using a 3D hydrodynamic model, Fluidity, and consists of 100 snapshots of a modeled tsunami event.

References

Abdullah, M. F., Siraj, S., and Hodgett, R. E.: An Overview of Multi-Criteria Decision Analysis (MCDA) Application in Managing Water-Related Disaster Events: Analyzing 20 Years of Literature for Flood and Drought Events, Water, 13, 1358, https://doi.org/10.3390/w13101358, 2021.

Ahmadlou, M., Al-Fugara, A., Al-Shabeeb, A., Arora, A., Al-Adamat, R., Pham, Q., Al-Ansari, N., Linh, N., and Sajedi, H.: Flood susceptibility mapping and assessment using a novel deep learning model combining multilayer perceptron and autoencoder neural networks, J. Flood Risk Manage., 14, e12683, https://doi.org/10.1111/jfr3.12683, 2021.

Ahmed, N., Hoque, M. A.-A., Arabameri, A., Pal, S. C., Chakrabortty, R., and Jui, J.: Flood susceptibility mapping in Brahmaputra floodplain of Bangladesh using deep boost, deep learning neural network, and artificial neural network, Geocarto Int., 1–22, https://doi.org/10.1080/10106049.2021.2005698, 2021.

Amini, J.: A method for generating floodplain maps using IKONOS images and DEMs, Int. J. Remote Sens., 31, 2441–2456, https://doi.org/10.1080/01431160902929230, 2010.

Ávila, A., Justino, F., Wilson, A., Bromwich, D., and Amorim, M.: Recent precipitation trends, flash floods and land-

slides in southern Brazil, Environ. Res. Lett., 11, 114029, https://doi.org/10.1088/1748-9326/11/11/114029, 2016.

Badrinarayanan, V., Kendall, A., and Cipolla, R.: Seg-Net: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation, 39, 2481–2495, https://doi.org/10.1109/TPAMI.2016.2644615, 2017.

Balestriero, R., Pesenti, J., and LeCun, Y.: Learning in High Dimension Always Amounts to Extrapolation, arXiv [preprint], https://doi.org/10.48550/arXiv.2110.09485, 2021.

Battaglia, P. W., Hamrick, J. B., Bapst, V., Sanchez-Gonzalez, A., Zambaldi, V., Malinowski, M., Tacchetti, A., Raposo, D., Santoro, A., Faulkner, R., Gulcehre, C., Song, F., Ballard, A., Gilmer, J., Dahl, G., Vaswani, A., Allen, K., Nash, C., Langston, V., Dyer, C., Heess, N., Wierstra, D., Kohli, P., Botvinick, M., Vinyals, O., Li, Y., and Pascanu, R.: Relational inductive biases, deep learning, and graph networks, arXiv [preprint], 1–40, https://doi.org/10.48550/arXiv.1806.01261, 2018.

Berkhahn, S., Fuchs, L., and Neuweiler, I.: An ensemble neural network model for real-time prediction of urban floods, J. Hydrol., 575, 743–754, 2019.

Blundell, C., Cornebise, J., Kavukcuoglu, K., and Wierstra, D.: Weight uncertainty in neural network, in: International Conference on Machine Learning, PMLR, 1613–1622, 2015.

Bobée, B. and Rasmussen, P. F.: Recent advances in flood frequency analysis, Rev. Geophys., 33, 1111–1116, 1995.

Bodnar, C., Frasca, F., Otter, N., Wang, Y., Lio, P., Montufar, G. F., and Bronstein, M.: Weisfeiler and Lehman go cellular: CW networks, Advances in Neural Information Processing Systems, 34, 2625–2640, 2021.

Bomers, A., Schielen, R. M., and Hulscher, S. J.: The influence of grid shape and grid size on hydraulic river modelling performance, Environ. Fluid Mech., 19, 1273–1294, https://doi.org/10.1007/s10652-019-09670-4, 2019.

Bonafilia, D., Tellman, B., Anderson, T., and Issenberg, E.: Sen1Floods11: a georeferenced dataset to train and test deep learning flood algorithms for Sentinel-1, IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW), 2020, 835–845, https://doi.org/10.1109/CVPRW50498.2020.00113, 2020.

Bowes, B. D., Tavakoli, A., Wang, C., Heydarian, A., Behl, M., Beling, P. A., and Goodall, J. L.: Flood mitigation in coastal urban catchments using real-time stormwater infrastructure control and reinforcement learning, J. Hydroinfo., 23, 529–547, 2021.

Bradley, A. P.: The use of the area under the ROC curve in the evaluation of machine learning algorithms, Pattern Recog., 30, 1145–1159, 1997.

Bronstein, M. M., Bruna, J., Lecun, Y., Szlam, A., and Vandergheynst, P.: Geometric Deep Learning: Going beyond Euclidean data, IEEE Signal Proc. Mag., 34, 18–42, https://doi.org/10.1109/MSP.2017.2693418, 2017.

Bronstein, M. M., Bruna, J., Cohen, T., and Veličković, P.: Geometric deep learning: Grids, groups, graphs, geodesics, and gauges, arXiv [preprint], https://doi.org/10.48550/arXiv.2104.13478, 2021.

Candy, A. S.: A consistent approach to unstructured mesh generation for geophysical models, arXiv [preprint], https://doi.org/10.48550/arXiv.1703.08491, 2017.

Chakrabortty, R., Chandra Pal, S., Rezaie, F., Arabameri, A., Lee, S., Roy, P., Saha, A., Chowdhuri, I., and

Moayedi, H.: Flash-flood hazard susceptibility mapping in Kangsabati River Basin, India, Geocarto Int., 1–23, https://doi.org/10.1080/10106049.2021.1953618, 2021a.

Chakrabortty, R., Pal, S. C., Janizadeh, S., Santosh, M., Roy, P., Chowdhuri, I., and Saha, A.: Impact of Climate Change on Future Flood Susceptibility: an Evaluation Based on Deep Learning Algorithms and GCM Model, Water Res. Manage., 35, 4251–4274, 2021b.

Chang, D.-L., Yang, S.-H., Hsieh, S.-L., Wang, H.-J., and Yeh, K.-C.: Artificial intelligence methodologies applied to prompt pluvial flood estimation and prediction, Water, 12, 3552, https://doi.org/10.3390/w12123552, 2020.

Chang, L.-C., Shen, H.-Y., Wang, Y.-F., Huang, J.-Y., and Lin, Y.-T.: Clustering-based hybrid inundation model for forecasting flood inundation depths, J. Hydrol., 385, 257–268, https://doi.org/10.1016/j.jhydrol.2010.02.028, 2010.

Chen, J., Huang, G., and Chen, W.: Towards better flood risk management: Assessing flood risk and investigating the potential mechanism based on machine learning models, J. Environ. Manage., 112810, https://doi.org/10.1016/j.jenvman.2021.112810, 2021.

Chicco, D. and Jurman, G.: The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation, BMC Genom., 21, 1–13, 2020.

Cho, K., van Merrienboer, B., Bahdanau, D., and Bengio, Y. On the Properties of Neural Machine Translation: Encoder–Decoder Approaches, in: Proceedings of SSST-8, Eighth Workshop on Syntax, Semantics and Structure in Statistical Translation, Doha, Qatar, Association for Computational Linguistics, 103–111, 2014.

Chu, H., Wu, W., Wang, Q., Nathan, R., and Wei, J.: An ANN-based emulation modelling framework for flood inundation modelling: Application, challenges and future directions, Environ. Modell. Softw., 124, 104587, 2020.

Cian, F., Marconcini, M., and Ceccato, P.: Normalized Difference Flood Index for rapid flood mapping: Taking advantage of EO big data, Remote Sens. Environ., 209, 712–730, https://doi.org/10.1016/j.rse.2018.03.006, 2018.

Cortes, C., Mohri, M., and Syed, U.: Deep boosting, in: International conference on machine learning, PMLR, 1179–1187, 2014.

Costabile, P., Costanzo, C., and Macchione, F.: Performances and limitations of the diffusive approximation of the 2-d shallow water equations for flood simulation in urban and rural areas, Appl. Numer. Math., 116, 141–156, https://doi.org/10.1016/j.apnum.2016.07.003, 2017.

Costache, R., Ngo, P., and Bui, D.: Novel ensembles of deep learning neural network and statistical learning for flash-flood susceptibility mapping, Water, 12, 1549, 2020.

Damianou, A. and Lawrence, N. D.: Deep gaussian processes, in: Artificial intelligence and statistics, PMLR, 207–215, 2013.

Darabi, H., Rahmati, O., Naghibi, S., Mohammadi, F., Ahmadisharaf, E., Kalantari, Z., Torabi Haghighi, A., Soleimanpour, S., Tiefenbacher, J., and Tien Bui, D.: Development of a novel hybrid multi-boosting neural network model for spatial prediction of urban flood, Geocarto Int., https://doi.org/10.1080/10106049.2021.1920629, 2021.

de Brito, M. M. and Evers, M.: Multi-criteria decision-making for flood risk management: a survey of the current state

of the art, Nat. Hazards Earth Syst. Sci., 16, 1019–1033, https://doi.org/10.5194/nhess-16-1019-2016, 2016.

De Haan, P., Weiler, M., Cohen, T., and Welling, M.: Gauge equivariant mesh CNNs anisotropic convolutions on geometric graphs, arXiv, https://doi.org/10.48550/arXiv.2003.05425, 2020.

de Moel, H., van Alphen, J., and Aerts, J. C. J. H.: Flood maps in Europe – methods, availability and use, Nat. Hazards Earth Syst. Sci., 9, 289–301, https://doi.org/10.5194/nhess-9-289-2009, 2009.

de Moel, H., Jongman, B., Kreibich, H., Merz, B., Penning-Rowsell, E., and Ward, P. J.: Flood risk assessments at different spatial scales, Mitig. Adapt. Strat. Gl., 20, 865–890, 2015.

Delgado, R. and Tibau, X.-A.: Why Cohen's Kappa should be avoided as performance measure in classification, PloS One, 14, e0222916, https://doi.org/10.1371/journal.pone.0222916, 2019.

Destro, E., Amponsah, W., Nikolopoulos, E. I., Marchi, L., Marra, F., Zoccatelli, D., and Borga, M.: Coupled prediction of flash flood response and debris flow occurrence: Application on an alpine extreme flood event, J. Hydrol., 558, 225–237, https://doi.org/10.1016/j.jhydrol.2018.01.021, 2018.

Di Baldassarre, G., Schumann, G., Bates, P. D., Freer, J. E., and Beven, K. J.: Flood-plain mapping: a critical discussion of deterministic and probabilistic approaches, J. Sci. Hydrol., 55, 364–376, 2010.

Domeneghetti, A., Vorogushyn, S., Castellarin, A., Merz, B., and Brath, A.: Probabilistic flood hazard mapping: effects of uncertain boundary conditions, Hydrol. Earth Syst. Sci., 17, 3127–3140, https://doi.org/10.5194/hess-17-3127-2013, 2013.

Dong, S., Yu, T., Farahmand, H., and Mostafavi, A.: A hybrid deep learning model for predictive flood warning and situation awareness using channel network sensors data, Comput.-Aided Civ. Inf., 36, 402–420, 2021.

Donthu, N., Kumar, S., Mukherjee, D., Pandey, N., and Lim, W. M.: How to conduct a bibliometric analysis: An overview and guidelines, J. Business Res., 133, 285–296, 2021.

Dottori, F., Alfieri, L., Bianchi, A., Skoien, J., and Salamon, P.: A new dataset of river flood hazard maps for Europe and the Mediterranean Basin, Earth Syst. Sci. Data, 14, 1549–1569, https://doi.org/10.5194/essd-14-1549-2022, 2022.

Ebli, S., Defferrard, M., and Spreemann, G.: Simplicial Neural Networks, arXiv [preprint], https://doi.org/10.48550/arXiv.2010.03633, 2020.

European Union: Directive 2007/60/EC of the European Counil and European Parliment of 23 October 2007 on the assessment and management of flood risks, Official Journal of the European Union, 27–34, http://eur-lex.europa.eu/legal-content/EN/TXT/PDF/?uri=CELEX:32007L0060&from=EN (last access: 20 February 2022), 2007.

Fang, Z., Wang, Y., Peng, L., and Hong, H.: Predicting flood susceptibility using LSTM neural networks, J. Hydrol., 594, 125734, https://doi.org/10.1016/j.jhydrol.2020.125734, 2020a.

Fang, Z., Yang, T., and Jin, Y.: DeepStreet: A deep learning powered urban street network generation module, arXiv [preprint], https://doi.org/10.48550/arXiv.2010.04365, 2020b.

Fazeli-Varzaneh, M., Bettinger, P., Ghaderi-Azad, E., Kozak, M., Mafi-Gholami, D., and Jaafari, A.: Forestry Research in the Middle East: A Bibliometric Analysis, Sustainability, 13, 8261, https://doi.org/10.3390/su13158261, 2021.

Ferraro, D., Costabile, P., Costanzo, C., Petaccia, G., and Macchione, F.: A spectral analysis approach for the a priori generation of computational grids in the 2-D hydrodynamic-based runoff simulations at a basin scale, J. Hydrol., 582, 124508, https://doi.org/10.1016/j.jhydrol.2019.124508, 2020.

Ferreira, L. A., Fonseca, A. R., Lima, N. Z., Mesquita, R. C., and Salgado, G. C.: Graphical interface for electromagnetic problem solving using meshless methods, Journal of Microwaves, Optoelectron. Elec. Appl., 14, SI–54 to SI, 2015.

Gama, F., Isufi, E., Leus, G., and Ribeiro, A.: Graphs, convolutions, and neural networks: From graph filters to graph neural networks, IEEE Signal Processing Magazine, 37, 128–138, 2020.

Gebrehiwot, A., Hashemi-Beni, L., Thompson, G., Kordjamshidi, P., and Langan, T. E.: Deep convolutional neural network for flood extent mapping using unmanned aerial vehicles data, Sensors (Switzerland), 19, https://doi.org/10.3390/s19071486, 2019.

Glenis, V., McGough, A. S., Kutija, V., Kilsby, C., and Woodman, S.: Flood modelling for cities using Cloud computing, J. Cloud Comput., 2, 1–14, 2013.

Goodfellow, I.: Nips 2016 tutorial: Generative adversarial networks, arXiv [preprint], https://doi.org/10.48550/arXiv.1701.00160, 2016.

Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y.: Generative adversarial nets, Adv. Neural Info. Proc. Syst., 27, https://doi.org/10.48550/arXiv.1406.2661, 2014.

Goodfellow, I., Bengio, Y., and Courville, A.: Deep Learning, MIT Press, http://www.deeplearningbook.org (last access: 8 August 2022), 2016.

Guo, Z., Leitao, J. P., Simões, N. E., and Moosavi, V.: Data-driven flood emulation: Speeding up urban flood predictions by deep convolutional neural networks, J. Flood Risk Manage., 14, e12684, https://doi.org/10.1111/jfr3.12684, 2021.

Hajij, M., Istvan, K., and Zamzmi, G.: Cell complex neural networks, arXiv [preprint], https://doi.org/10.48550/arXiv.2010.00743, 2020.

Hashemi-Beni, L. and Gebrehiwot, A.: Flood Extent Mapping: An Integrated Method Using Deep Learning and Region Growing Using UAV Optical Data, IEEE J. Sel. Top. Appl., 14, 2127–2135, 2021.

Hess, L., Melack, J., Filoso, S., and Wang, Y.: Delineation of inundated area and vegetation along the Amazon floodplain with the SIR-C synthetic aperture radar, IEEE T. Geosci. Remote, 33, 896–904, https://doi.org/10.1109/36.406675, 1995.

Hochreiter, S. and Schmidhuber, J.: Long short-term memory, Neural computation, 9, 1735–1780, 1997.

Hofmann, J. and Schüttrumpf, H.: floodGAN: Using Deep Adversarial Learning to Predict Pluvial Flooding in Real Time, Water, 13, https://doi.org/10.3390/w13162255, 2021.

Horritt, M. S. and Bates, P. D.: Evaluation of 1D and 2D numerical models for predicting river flood inundation, J. Hydrol., 268, 87–99, https://doi.org/10.1016/S0022-1694(02)00121-X, 2002.

Hosseiny, H.: A deep learning model for predicting river flood depth and extent, Environ. Modell. Softw., 145, 105186, https://doi.org/10.1016/j.envsoft.2021.105186, 2021.

Hou, J., Li, X., Bai, G., Wang, X., Zhang, Z., Yang, L., Du, Y., Ma, Y., Fu, D., and Zhang, X.: A deep learning technique based flood propagation experiment, J. Flood Risk Manage., https://doi.org/10.1111/jfr3.12718, 2021.

Hu, R., Fang, F., Pain, C., and Navon, I.: Rapid spatio-temporal flood prediction and uncertainty quantification using a deep learning method, J. Hydrol., 575, 911–920, https://doi.org/10.1016/j.jhydrol.2019.05.087, 2019.

Huang, P.-C., Hsu, K.-L., and Lee, K.: Improvement of Two-Dimensional Flow-Depth Prediction Based on Neural Network Models By Preprocessing Hydrological and Geomorphological Data, Water Resour. Manage., 35, 1079–1100, 2021a.

Huang, P. C., Hsu, K. L., and Lee, K. T.: Improvement of Two-Dimensional Flow-Depth Prediction Based on Neural Network Models By Preprocessing Hydrological and Geomorphological Data, Water Resour. Manage., 35, 1079–1100, https://doi.org/10.1007/s11269-021-02776-9, 2021b.

Ichim, L. and Popescu, D.: Segmentation of vegetation and flood from aerial images based on decision fusion of neural networks, Remote Sens., 12, https://doi.org/10.3390/rs12152490, 2020.

Ireland, G., Volpi, M., and Petropoulos, G. P.: Examining the Capability of Supervised Machine Learning Classifiers in Extracting Flooded Areas from Landsat TM Imagery: A Case Study from a Mediterranean Flood, Remote Sens., 7, 3372–3399, https://doi.org/10.3390/rs70303372, 2015.

Isikdogan, F., Bovik, A. C., and Passalacqua, P.: Surface water mapping by deep learning, IEEE T. Geosci. Remote, 10, 4909–4918, 2017.

Isufi, E., Gama, F., and Ribeiro, A.: EdgeNets: Edge varying graph neural networks, IEEE T. Pattern Anal., https://doi.org/10.1109/TPAMI.2021.3111054, 2021.

Jacquier, P., Abdedou, A., Delmas, V., and Soulaïmani, A.: Non-intrusive reduced-order modeling using uncertainty-aware Deep Neural Networks and Proper Orthogonal Decomposition: Application to flood modeling, J. Comput. Phys., 424, 109854, https://doi.org/10.1016/j.jcp.2020.109854, 2021.

Jafari, N., Li, X., Chen, Q., Le, C.-Y., Betzer, L., and Liang, Y.: Real-time water level monitoring using live cameras and computer vision techniques, Comput. Geosci., 147, https://doi.org/10.1016/j.cageo.2020.104642, 2021.

Jahangir, M., Mousavi Reineh, S., and Abolghasemi, M.: Spatial predication of flood zonation mapping in Kan River Basin, Iran, using artificial neural network algorithm, Weather Clim. Extr., 25, https://doi.org/0.1016/j.wace.2019.100215, 2019.

Jiang, P., Meinert, N., Jordão, H., Weisser, C., Holgate, S., Lavin, A., Lütjens, B., Newman, D., Wainwright, H., Walker, C., and Barnard, P.: Digital Twin Earth–Coasts: Developing a fast and physics-informed surrogate model for coastal floods via neural operators, arXiv [preprint], https://doi.org/10.48550/arXiv.2110.07100, 2021.

Jonkman, S. and Vrijling, J.: Loss of life due to floods, J. Flood Risk Manage., 1, 43–56, https://doi.org/10.1111/j.1753-318x.2008.00006.x, 2008.

Kabir, S., Patidar, S., Xia, X., Liang, Q., Neal, J., and Pender, G.: A deep convolutional neural network model for rapid prediction of fluvial flood inundation, J. Hydrol., 590, 125481, https://doi.org/10.1016/j.jhydrol.2020.125481, 2020.

Kalantar, B., Ueda, N., Saeidi, V., Janizadeh, S., Shabani, F., Ahmadi, K., and Shabani, F.: Deep Neural Network Utilizing Remote Sensing Datasets for Flood Hazard Susceptibility Mapping in Brisbane, Australia, Remote Sens., 13, 2638, https://doi.org/10.3390/rs13132638, 2021.

Kang, W., Xiang, Y., Wang, F., Wan, L., and You, H.: Flood Detection in Gaofen-3 SAR Images via Fully Convolutional Networks, Sensors, 18, 2915, https://doi.org/10.3390/s18092915, 2018.

Kao, I. F., Liou, J. Y., Lee, M. H., and Chang, F. J.: Fusing stacked autoencoder and long short-term memory for regional multistep-ahead flood inundation forecasts, J. Hydrol., 598, 126371, https://doi.org/10.1016/j.jhydrol.2021.126371, 2021.

Kalos, M. H. and Whitlock, P. A.: Monte carlo methods. John Wiley & Sons, 2009.

Kazakis, N., Kougias, I., and Patsialis, T.: Assessment of flood hazard areas at a regional scale using an index-based approach and Analytical Hierarchy Process: Application in Rhodope-Evros region, Greece, Sci. Total Environ., 538, 555–563, https://doi.org/10.1016/j.scitotenv.2015.08.055, 2015.

Khoirunisa, N., Ku, C.-Y., and Liu, C.-Y.: A GIS-based artificial neural network model for flood susceptibility assessment, Int. J. Environ. Res. Publ. Health, 18, 1–20, 2021.

Khosravi, K., Panahi, M., Golkarian, A., Keesstra, S. D., and Saco, P. M.: Convolutional neural network approach for spatial prediction of flood hazard at national scale of Iran, J. Hydrol., 591, 125552, https://doi.org/10.1016/j.jhydrol.2020.125552, 2020.

Kia, M. B., Pirasteh, S., Pradhan, B., Mahmud, A. R., Sulaiman, W. N. A., and Moradi, A.: An artificial neural network model for flood simulation using GIS: Johor River Basin, Malaysia, Environ. Earth Sci., 67, 251–264, 2012.

Kingma, D. P. and Welling, M.: Auto-encoding variational bayes, arXiv [preprint], https://doi.org/10.48550/arXiv.1312.6114, 2013.

Kourgialas, N. N. and Karatzas, G. P.: A national scale flood hazard mapping methodology: The case of Greece – Protection and adaptation policy approaches, Sci. Total Environ., 601, 441–452, https://doi.org/10.1016/j.scitotenv.2017.05.197, 2017.

Kovachki, N., Li, Z., Liu, B., Azizzadenesheli, K., Bhattacharya, K., Stuart, A., and Anandkumar, A.: Neural operator: Learning maps between function spaces, arXiv [preprint], https://doi.org/10.48550/arXiv.2108.08481, 2021.

Kratzert, F., Herrnegger, M., Klotz, D., Hochreiter, S., and Klambauer, G.: NeuralHydrology – Interpreting LSTMs in Hydrology, Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), Explainable AI: Interpreting, Explaining and Visualizing Deep Learning, 347–362, https://doi.org/10.1007/978-3-030-28954-6_19, 2019a.

Kratzert, F., Klotz, D., Herrnegger, M., Sampson, A. K., Hochreiter, S., and Nearing, G. S.: Toward Improved Predictions in Ungauged Basins: Exploiting the Power of Machine Learning, Water Resour. Res., 55, 11344–11354, https://doi.org/10.1029/2019WR026065, 2019b.

Kummu, M., De Moel, H., Ward, P. J., and Varis, O.: How close do we live to water? A global analysis of population distance to freshwater bodies, PloS One, 6, e20578, https://doi.org/10.1371/journal.pone.0020578, 2011.

LeCun, Y. and Bengio, Y.: Convolutional Networks for Images, Speech and Time Series , The MIT Press , 255–258, 1995.

LeCun, Y., Bengio, Y., and Hinton, G.: Deep learning, Nature, 521, 436–444, 2015.

Lei, X., Chen, W., Panahi, M., Falah, F., Rahmati, O., Uuemaa, E., Kalantari, Z., Ferreira, C. S. S., Rezaie, F., Tiefenbacher, J. P. and Lee, S.: Urban flood modeling using deep-learning

approaches in Seoul, South Korea, J. Hydrol., 601, 126684, https://doi.org/10.1016/j.jhydrol.2021.126684,2021.

Lendering, K., Jonkman, S., and Kok, M.: Effectiveness of emergency measures for flood prevention, J. Flood Risk Manage., 9, 320–334, 2016.

Li, L., Chen, Y., Xu, T., Liu, R., Shi, K., and Huang, C.: Super-resolution mapping of wetland inundation from remote sensing imagery based on integration of back-propagation neural network and genetic algorithm, Remote Sens. Environ., 164, 142–154, https://doi.org/10.1016/j.rse.2015.04.009, 2015.

Li, L., Chen, Y., Xu, T., Huang, C., Liu, R., and Shi, K.: Integration of Bayesian regulation back-propagation neural network and particle swarm optimization for enhancing sub-pixel mapping of flood inundation in river basins, Remote Sens. Lett., 7, 631–640, https://doi.org/10.1080/2150704X.2016.1177238, 2016a.

Li, L., Xu, T., and Chen, Y.: Improved urban flooding mapping from remote sensing images using generalized regression neural network-based super-resolution algorithm, Remote Sens., 8, 625, https://doi.org/10.3390/rs8080625, 2016b.

Li, Z., Kovachki, N., Azizzadenesheli, K., Liu, B., Bhattacharya, K., Stuart, A., and Anandkumar, A.: Neural operator: Graph kernel network for partial differential equations, arXiv [preprint], https://doi.org/10.48550/arXiv.2003.03485, 2020.

Li, Z., Kovachki, N., Azizzadenesheli, K., Liu, B., Bhattacharya, K., Stuart, A., and Anandkumar, A.: Fourier Neural Operator for Parametric Partial Differential Equations, arXiv [preprint], https://doi.org/10.48550/arXiv.2010.08895, 2021.

Lin, L., Di, L., Yu, E. G., Kang, L., Shrestha, R., Rahman, M. S., Tang, J., Deng, M., Sun, Z., Zhang, C., et al.: A review of remote sensing in flood assessment, in: 2016 Fifth International Conference on Agro-Geoinformatics (Agro-Geoinformatics), IEEE, 1–4, 2016.

Lin, Q., Leandro, J., Gerber, S., and Disse, M.: Multi-step flood inundation forecasts with resilient backpropagation neural networks: Kulmbach case study, Water, 12, https://doi.org/10.3390/w12123568, 2020a.

Lin, Q., Leandro, J., Wu, W., Bhola, P., and Disse, M.: Prediction of Maximum Flood Inundation Extents With Resilient Backpropagation Neural Network: Case Study of Kulmbach, Front. Earth Sci., 8, https://doi.org/10.3389/feart.2020.00332, 2020b.

Lino, M., Cantwell, C., Bharath, A. A., and Fotiadis, S.: Simulating Continuum Mechanics with Multi-Scale Graph Neural Networks, arXiv [preprint], https://doi.org/10.48550/arXiv.2106.04900, 2021.

Liu, B., Li, X., and Zheng, G.: Coastal Inundation Mapping From Bitemporal and Dual-Polarization SAR Imagery Based on Deep Convolutional Neural Networks, J. Geophys. Res.-Oceans, 124, 9101–9113, 2019.

Liu, J., Wang, J., Xiong, J., Cheng, W., Sun, H., Yong, Z., and Wang, N.: Hybrid Models Incorporating Bivariate Statistics and Machine Learning Methods for Flash Flood Susceptibility Assessment Based on Remote Sensing Datasets, Remote Sens., 13, 4945, https://doi.org/10.3390/rs13234945, 2021.

Lu, L., Jin, P., and Karniadakis, G. E.: DeepONet: Learning nonlinear operators for identifying differential equations based on the universal approximation theorem of operators, CoRR, abs/1910.03193, http://arxiv.org/abs/1910.03193 (last access date: 18 August 2022), 2019.

Lütjens, B., Leshchinskiy, B., Requena-Mesa, C., Chishtie, F., Díaz-Rodriguez, N., Boulais, O., Piña, A., Newman, D., Lavin, A., Gal, Y., et al.: Physics-informed gans for coastal flood visualization, arXiv [preprint], https://doi.org/10.48550/arXiv.2010.08103, 2020.

Lütjens, B., Leshchinskiy, B., Requena-Mesa, C., Chishtie, F., Díaz-Rodríguez, N., Boulais, O., Sankaranarayanan, A., Piña, A., Gal, Y., Raïssi, C., et al.: Physically-Consistent Generative Adversarial Networks for Coastal Flood Visualization, arXiv [preprint], https://doi.org/10.48550/arXiv.2104.04785, 2021.

Löwe, R., Böhm, J., Jensen, D. G., Leandro, J., and Rasmussen, S. H.: U-FLOOD – Topographic deep learning for predicting urban pluvial flood water depth, J. Hydrol., 603, 126898, https://doi.org/10.1016/j.jhydrol.2021.126898, 2021.

Ma, X., Hong, Y., Song, Y., and Chen, Y.: A super-resolution convolutional-neural-network-based approach for subpixel mapping of hyperspectral images, IEEE J. Sel. Top. Appl., 12, 4930–4939, 2019.

Mahesh, R. B., Leandro, J., and Lin, Q.: Physics Informed Neural Network for Spatial-Temporal Flood Forecasting, in: Climate Change and Water Security, edited by Kolathayar, S., Mondal, A., and Chian, S. C., Springer Singapore, Singapore, 77–91, 2022.

Mahmoud, S. H. and Gan, T. Y.: Multi-criteria approach to develop flood susceptibility maps in arid regions of Middle East, J. Cleaner Prod., 196, 216–229, https://doi.org/10.1016/j.jclepro.2018.06.047, 2018.

Manavalan, R.: SAR image analysis techniques for flood area mapping-literature survey, Earth Sci. Inf., 10, 1–14, 2017.

Manjusree, P., Kumar, L. P., Bhatt, C. M., Rao, G. S., and Bhanumurthy, V.: Optimization of threshold ranges for rapid flood inundation mapping by evaluating backscatter profiles of high incidence angle SAR images, Int. J. Dis. Risk Sci., 3, 113–122, 2012.

Mao, Z., Jagtap, A. D., and Karniadakis, G. E.: Physics-informed neural networks for high-speed flows, Comput. Meth. Appl. Mech. Eng., 360, 112789, https://doi.org/10.1016/j.cma.2019.112789, 2020.

Martinis, S., Twele, A., and Voigt, S.: Towards operational near real-time flood detection using a split-based automatic thresholding procedure on high resolution TerraSAR-X data, Nat. Hazards Earth Syst. Sci., 9, 303–314, https://doi.org/10.5194/nhess-9-303-2009, 2009.

Tabari, H.: Climate change impact on flood and extreme precipitation increases with water availability, Sci. Rep., 10, 1–10, 2020.

Mavriplis, D.: Unstructured grid techniques, Annu. Rev. Fluid Mech., 29, 473–514, 1997.

Meraner, A., Ebel, P., Zhu, X. X., and Schmitt, M.: Cloud removal in Sentinel-2 imagery using a deep residual neural network and SAR-optical data fusion, ISPRS J. Photo. Remote Sens., 166, 333–346, 2020.

Ming, X., Liang, Q., Xia, X., Li, D., and Fowler, H. J.: Real-Time Flood Forecasting Based on a High-Performance 2-D Hydrodynamic Model and Numerical Weather Predictions, Water Resour. Res., 56, e2019WR025583, https://doi.org/10.1029/2019WR025583, 2020.

Mitchell, T. M.: Machine Learning, McGraw-Hill, 1997.

Mosavi, A., Ozturk, P., and Chau, K.-W.: Flood Prediction Using Machine Learning Models: Literature Review, Water, 10, 1536, https://doi.org/10.3390/w10111536, 2018.

Moy de Vitry, M., Kramer, S., Wegner, J. D., and Leitão, J. P.: Scalable flood level trend monitoring with surveillance cameras using a deep convolutional neural network, Hydrol. Earth Syst. Sci., 23, 4621–4634, https://doi.org/10.5194/hess-23-4621-2019, 2019.

Muñoz, D., Muñoz, P., Moftakhari, H., and Moradkhani, H.: From local to regional compound flood mapping with deep learning and data fusion techniques, Sci. Total Environ., 782, 146927, https://doi.org/10.1016/j.scitotenv.2021.146927, 2021.

Nemni, E., Bullock, J., Belabbes, S., and Bromley, L.: Fully Convolutional Neural Network for Rapid Flood Segmentation in Synthetic Aperture Radar Imagery, Remote Sens., 12, 2532, https://doi.org/10.3390/rs12162532, 2020.

Neumann, B., Vafeidis, A. T., Zimmermann, J., and Nicholls, R. J.: Future coastal population growth and exposure to sea-level rise and coastal flooding-a global assessment, PloS One, 10, e0118571, https://doi.org/10.1371/journal.pone.0118571, 2015.

Ngo, P. T. T., Hoang, N. D., Pradhan, B., Nguyen, Q. K., Tran, X. T., Nguyen, Q. M., Nguyen, V. N., Samui, P., and Bui, D. T.: A novel hybrid swarm optimized multilayer neural network for spatial prediction of flash floods in tropical areas using sentinel-1 SAR imagery and geospatial data, Sensors, 18, 3704, https://doi.org/10.3390/s18113704, 2018.

Nogueira, K., Fadel, S. G., Dourado, I. C., De O. Werneck, R., Muñoz, J. A., Penatti, O. A., Calumby, R. T., Li, L. T., Dos Santos, J. A., and Da S. Torres, R.: "Exploiting ConvNet Diversity for Flooding Identification", in IEEE Geoscience and Remote Sensing Letters, Vol. 15, no. 9, 1446–1450, Sept. 2018, https://doi.org/10.1109/LGRS.2018.2845549, 2017.

Observatory, D. F.: Space-based Measurement, Mapping, and Modeling of Surface Water, https://floodobservatory.colorado.edu/, last access: 11 November 2021.

Oord, A. v. d., Dieleman, S., Zen, H., Simonyan, K., Vinyals, O., Graves, A., Kalchbrenner, N., Senior, A., and Kavukcuoglu, K.: Wavenet: A generative model for raw audio, arXiv [preprint], https://doi.org/10.48550/arXiv.1609.03499, 2016.

Panahi, M., Jaafari, A., Shirzadi, A., Shahabi, H., Rahmati, O., Omidvar, E., Lee, S., and Bui, D.: Deep learning neural networks for spatially explicit prediction of flash flood probability, Geosci. Front., 12, https://doi.org/10.1016/j.gsf.2020.09.007, 2021.

Papaioannou, G., Vasiliades, L., Loukas, A., and Aronica, G. T.: Probabilistic flood inundation mapping at ungauged streams due to roughness coefficient uncertainty in hydraulic modelling, Adv. Geosci., 44, 23–34, 2017.

Peng, B., Meng, Z., Huang, Q., and Wang, C.: Patch Similarity Convolutional Neural Network for Urban Flood Extent Mapping Using Bi-Temporal Satellite Multispectral Imagery, Remote Sens., 11, 2492, https://doi.org/10.3390/rs11212492, 2019.

Pereira, J., Monteiro, J., Silva, J., Estima, J., and Martins, B.: Assessing flood severity from crowdsourced social media photos with deep neural networks, Multimedia Tools Appl., 79, 26197–26223, https://doi.org/10.1007/s11042-020-09196-8, 2020.

Pfaff, T., Fortunato, M., Sanchez-Gonzalez, A., and Battaglia, P. W.: Learning Mesh-Based Simulation with Graph Networks, International Conference on Learning Representations (ICLR), https://doi.org/10.48550/arXiv.2010.03409, 2020.

Pham, B. T., Luu, C., Van Phong, T., Trinh, P. T., Shirzadi, A., Renoud, S., Asadi, S., Van Le, H., von Meding, J., and Clague, J. J.: Can deep learning algorithms outperform benchmark machine learning algorithms in flood susceptibility modeling?, J. Hydrol., 592, 125615, https://doi.org/10.1016/j.jhydrol.2020.125615, 2021.

Popa, M., Peptenatu, D., Draghici, C., and Diaconu, D.: Flood hazard mapping using the flood and Flash-Flood Potential Index in the Buzau River catchment, Romania, Water, 11, 2116, 2019.

Prestininzi, P.: Suitability of the diffusive model for dam break simulation: Application to a CADAM experiment, J. Hydrol., 361, 172–185, 2008.

Raissi, M., Perdikaris, P., and Karniadakis, G. E.: Physics-informed neural networks: A deep learning framework for solving forward and inverse problems involving nonlinear partial differential equations, J. Comput. Phys., 378, 686–707, https://doi.org/10.1016/j.jcp.2018.10.045, 2019.

Rasmussen, C. E.: Gaussian processes in machine learning, in: Summer school on machine learning, Springer, 63–71, 2003.

Ravuri, S., Lenc, K., Willson, M., Kangin, D., Lam, R., Mirowski, P., Fitzsimons, M., Athanassiadou, M., Kashem, S., Madge, S., et al.: Skillful Precipitation Nowcasting using Deep Generative Models of Radar, arXiv [preprint], https://doi.org/10.48550/arXiv.2104.00954, 2021.

Rossi, C., Acerbo, F., Ylinen, K., Juga, I., Nurmi, P., Bosca, A., Tarasconi, F., Cristoforetti, M., and Alikadic, A.: Early detection and information extraction for weather-induced floods using social media streams, Int. J. Dis. Risk Reduct., 30, 145–157, https://doi.org/10.1016/j.ijdrr.2018.03.002, 2018.

Rumelhart, D. E., Hinton, G. E., and Williams, R. J.: Learning representations by back-propagating errors, Nature, 323, 533–536, 1986.

Saeed, M., Li, H., Ullah, S., Rahman, A.-u., Ali, A., Khan, R., Hassan, W., Munir, I., and Alam, S.: Flood Hazard Zonation Using an Artificial Neural Network Model: A Case Study of Kabul River Basin, Pakistan, Sustainability, 13, 13953, https://doi.org/10.3390/su132413953, 2021.

Sarker, C., Mejias, L., Maire, F., and Woodley, A.: Flood Mapping with Convolutional Neural Networks Using Spatio-Contextual Pixel Information, Remote Sens., 11, 2331, https://doi.org/10.3390/rs11192331, 2019.

Schmidt, V., Luccioni, A., Mukkavilli, S. K., Balasooriya, N., Sankaran, K., Chayes, J., and Bengio, Y.: Visualizing the consequences of climate change using cycle-consistent adversarial networks, arXiv [preprint], https://doi.org/10.48550/arXiv.1905.03709, 2019.

Serinaldi, F., Loecker, F., Kilsby, C. G., and Bast, H.: Flood propagation and duration in large river basins: a data-driven analysis for reinsurance purposes, Nat. Hazards, 94, 71–92, https://doi.org/10.1007/s11069-018-3374-0, 2018.

Shi, X., Chen, Z., Wang, H., Yeung, D. Y., Wong, W. K., and Woo, W. C.: Convolutional LSTM network: A machine learning approach for precipitation nowcasting, Adv. Neural Info. Proc. Syst., 2015, 802–810, 2015.

Shirzadi, A., Asadi, S., Shahabi, H., Ronoud, S., Clague, J. J., Khosravi, K., Pham, B. T., Ahmad, B. B., and Bui, D. T.: A novel ensemble learning based on Bayesian Belief Network coupled with an extreme learning machine for flash flood

susceptibility mapping, Eng. Appl. Art. Intel., 96, 103971, https://doi.org/10.1016/j.engappai.2020.103971, 2020.

Sikorska, A. E., Viviroli, D., and Seibert, J.: Flood-type classification in mountainous catchments using crisp and fuzzy decision trees, J Am. Water Resour. Assoc., 5, 2–2, https://doi.org/10.1111/j.1752-1688.1969.tb04897.x, 2015.

Sit, M., Demiray, B. Z., Xiang, Z., Ewing, G. J., Sermet, Y., and Demir, I.: A comprehensive review of deep learning applications in hydrology and water resources, Water Sci. Technol., 82, 2635–2670, https://doi.org/10.2166/wst.2020.369, 2020.

Sridharan, B., Bates, P. D., Sen, D., and Kuiry, S. N.: Local-inertial shallow water model on unstructured triangular grids, Adv. Water Res., 152, 103930, https://doi.org/10.1016/j.advwatres.2021.103930, 2021.

Syifa, M., Park, S. J., Achmad, A. R., Lee, C.-W., and Eom, J.: Flood mapping using remote sensing imagery and artificial intelligence techniques: a case study in Brumadinho, Brazil, J. Coast. Res., 90, 197–204, 2019.

Taormina, R. and Galelli, S.: Deep-learning approach to the detection and localization of cyber-physical attacks on water distribution systems, J. Water Res. Plan. Manage., 144, 04018065, https://doi.org/10.1061/(ASCE)WR.1943-5452.0000983, 2018.

Tehrany, M. S., Lee, M.-J., Pradhan, B., Jebur, M. N., and Lee, S.: Flood susceptibility mapping using integrated bivariate and multivariate statistical models, Environ. Earth Sci., 72, 4001–4015, 2014.

Teng, J., Jakeman, A. J., Vaze, J., Croke, B. F., Dutta, D., and Kim, S.: Flood inundation modelling: A review of methods, recent advances and uncertainty analysis, Environ. Modell. Softw., 90, 201–216, https://doi.org/10.1016/j.envsoft.2017.01.006, 2017.

Tien, D., Hoang, N.-D., Martínez-álvarez, F., Ngo, P.-T. T., Viet, P., Dat, T., Samui, P., and Costache, R.: A novel deep learning neural network approach for predicting flash flood susceptibility: A case study at a high frequency tropical storm area, Sci. Total Environ., 701, 134413, https://doi.org/10.1016/j.scitotenv.2019.134413, 2020.

van de Giesen, N., Hut, R., and Selker, J.: The trans-African hydrometeorological observatory (TAHMO), Wiley Interdisciplinary Reviews, Water, 1, 341–348, 2014.

Vandaele, R., Dance, S. L., and Ojha, V.: Deep learning for automated river-level monitoring through river-camera images: an approach based on water segmentation and transfer learning, Hydrol. Earth Syst. Sci., 25, 4435–4453, https://doi.org/10.5194/hess-25-4435-2021, 2021.

Vandenberg-Rodes, A., Moftakhari, H. R., AghaKouchak, A., Shahbaba, B., Sanders, B. F., and Matthew, R. A.: Projecting nuisance flooding in a warming climate using generalized linear models and Gaussian processes, J. Geophys. Res.-Oceans, 121, 8008–8020, https://doi.org/10.1002/2016JC012084, 2016.

Wang, R., Walters, R., and Yu, R.: Incorporating symmetry into deep dynamics models for improved generalization, arXiv [preprint], https://doi.org/10.48550/arXiv.2002.03061, 2020.

Wang, Y., Fang, Z., Hong, H., and Peng, L.: Flood susceptibility mapping using convolutional neural network frameworks, J. Hydrol., 582, 124482, https://doi.org/10.1016/j.jhydrol.2019.124482, 2020.

Wardhani, N. W. S., Rochayani, M. Y., Iriany, A., Sulistyono, A. D., and Lestantyo, P.: Cross-validation metrics for evaluating classification performance on imbalanced data, in: 2019 international conference on computer, control, informatics and its applications (ic3ina), IEEE, 14–18, 2019.

Wieland, M. and Martinis, S.: A modular processing chain for automated flood monitoring from multi-spectral satellite data, Remote Sens., 11, 2330, https://doi.org/10.3390/rs11192330, 2019.

Wu, Z., Pan, S., Chen, F., Long, G., Zhang, C., and Yu, P. S.: A Comprehensive Survey on Graph Neural Networks, IEEE T. Neur. Net. Lear., 32, 4–24, https://doi.org/10.1109/TNNLS.2020.2978386, 2021.

Xie, S., Wu, W., Mooser, S., Wang, Q., Nathan, R., and Huang, Y.: Artificial neural network based hybrid modeling approach for flood inundation modeling, J. Hydrol., 592, 125605, https://doi.org/10.1016/j.jhydrol.2020.125605, 2021.

Yakti, B. P., Adityawan, M. B., Farid, M., Suryadi, Y., Nugroho, J., and Hadihardaja, I. K.: 2D modeling of flood propagation due to the failure of way Ela natural dam, in: MATEC Web of Conferences, Vol. 147, EDP Sciences, https://doi.org/10.1051/matecconf/201814703009, 2018.

Yang, M., Isufi, E., and Leus, G.: Simplicial Convolutional Neural Networks, ICASSP 2022–2022 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), 8847–8851, https://doi.org/10.1109/ICASSP43922.2022.9746017, 2022.

Yang, W., Zhang, X., Tian, Y., Wang, W., Xue, J.-H., and Liao, Q.: Deep learning for single image super-resolution: A brief review, IEEE T. Multimedia, 21, 3106–3121, 2019.

Yang, X. I. A., Zafar, S., Wang, J.-X., and Xiao, H.: Predictive large-eddy-simulation wall modeling via physics-informed neural networks, Phys. Rev. Fluids, 4, 034602, https://doi.org/10.1103/PhysRevFluids.4.034602, 2019.

Yokoya, N., Yamanoi, K., He, W., Baier, G., Adriano, B., Miura, H., and Oishi, S.: Breaking limits of remote sensing by deep learning from simulated data for flood and debris-flow mapping, IEEE T. Geosci. Remote, https://doi.org/10.1109/TGRS.2020.3035469, 2020.

Youssef, A. M., Pradhan, B., and Sefry, S. A.: Flash flood susceptibility assessment in Jeddah city (Kingdom of Saudi Arabia) using bivariate and multivariate statistical models, Environ. Earth Sci., 75, 1–16, https://doi.org/10.1007/s12665-015-4830-8, 2016.

Zhang, S., Xia, Z., Yuan, R., and Jiang, X.: Parallel computation of a dam-break flow model using OpenMP on a multi-core computer, J. Hydrol., 512, 126–133, 2014.

Zhang, Z., Flora, K., Kang, S., Limaye, A. B., and Khosronejad, A.: Data-driven prediction of turbulent flow statistics past bridge piers in large-scale rivers using convolutional neural networks, Water Resour. Res., 58, e2021WR030163, https://doi.org/10.1029/2021WR030163, 2021.

Zhao, G., Pang, B., Xu, Z., Peng, D., and Zuo, D.: Urban flood susceptibility assessment based on convolutional neural networks, J. Hydrol., 590, 125235, https://doi.org/10.1016/j.jhydrol.2020.125235, 2020.

Zhao, G., Bates, P., Neal, J., and Pang, B.: Design flood estimation for global river networks based on machine learning models, Hydrol. Earth Syst. Sci., 25, 5981–5999, https://doi.org/10.5194/hess-25-5981-2021, 2021a.

Zhao, G., Balstrøm, T., Mark, O., and Jensen, M. B.: Multi-Scale Target-Specified Sub-Model Approach for Fast Large-

Scale High-Resolution 2D Urban Flood Modelling, Water, 13, 259, https://doi.org/10.3390/w13030259, 2021b.

Zhao, G., Pang, B., Xu, Z., Cui, L., Wang, J., Zuo, D., and Peng, D.: Improving urban flood susceptibility mapping using transfer learning, J. Hydrol., 602, 126777, https://doi.org/10.1016/j.jhydrol.2021.126777, 2021c.

Zhou, Y., Wu, C., Li, Z., Cao, C., Ye, Y., Saragih, J., Li, H., and Sheikh, Y.: Fully convolutional mesh autoencoder using efficient spatially varying kernels. Advances in Neural Information Processing Systems, 33, 9251–9262, 2020. Zhou, Y., Wu, C., Li, Z., Cao, C., Ye, Y., Saragih, J., Li, H. and Sheikh, Y., 2020. Fully convolutional mesh autoencoder using efficient spatially varying kernels. Advances in Neural Information Processing Systems, 33, pp.9251-9262.

Zhou, Y., Wu, W., Nathan, R., and Wang, Q. J.: A rapid flood inundation modelling framework using deep learning with spatial reduction and reconstruction, Environ. Modell. Softw., 143, 105112, https://doi.org/10.1016/j.envsoft.2021.105112, 2021.

Zounemat-Kermani, M., Matta, E., Cominola, A., Xia, X., Zhang, Q., Liang, Q., and Hinkelmann, R.: Neurocomputing in surface water hydrology and hydraulics: A review of two decades retrospective, current status and future prospects, J. Hydrol., 588, 125085, https://doi.org/10.1016/j.jhydrol.2020.125085, 2020.