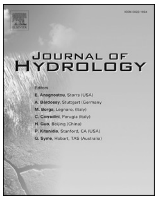
Contents lists available at ScienceDirect

Journal of Hydrology

journal homepage: www.elsevier.com/locate/jhydrol



U-FLOOD – Topographic deep learning for predicting urban pluvial flood water depth

Roland Löwe [a,*], Julian Böhm [a,b], David Getreuer Jensen [c], Jorge Leandro [d], Søren Højmark Rasmussen [c]

[a] Technical University of Denmark, Department of Environmental Engineering, Section of Climate and Monitoring, Miljøvej B115, 2800 Kgs. Lyngby, Denmark
[b] Technical University of Munich, Chair of Hydrology and River Basin Management, Arcisstr. 21, 80333 München, Germany
[c] EnviDan A/S, Vejlsøvej 23, 8600 Silkeborg, Denmark
[d] Chair of Hydromechanics and Hydraulic Engineering, Research Institute of Water and Environment, University of Siegen, Paul-Bonatz-Str. 9-11, 57068 Siegen, Germany

ARTICLE INFO

This manuscript was handled by Emmanouil Anagnostou, Editor-in-Chief, with the assistance of Xinyi Shen, Associate Editor

Keywords:
Urban pluvial flooding
Deep learning
Convolutional neural network
Hydrodynamic modelling
Emulation

ABSTRACT

This study investigates how deep-learning can be configured to optimise the prediction of 2D maximum water depth maps in urban pluvial flood events. A neural network model is trained to exploit patterns in hyetographs as well as in topographical data, with the specific aim of enabling fast predictions of flood depths for observed rain events and spatial locations that have not been included in the training dataset. A neural network architecture that is widely used for image segmentation (U-NET) is adapted for this purpose. Key novelties are a systematic investigation of which spatial inputs should be provided to the deep learning model, which hyper-parametrization optimizes predictive performance, and a systematic evaluation of prediction performance for locations and rain events that were not considered in training. We find that a spatial input dataset of only 5 variables that describe local terrain shape and imperviousness is optimal to generate predictions of water depth. Neural network architectures with between 97,000 and 260,000,000 parameters are tested, and a model with 28,000,000 parameters is found optimal. U-FLOOD is demonstrated to yield similar predictive performance as existing screening approaches, even though the assessment is performed for natural rain events and in locations unknown to the network, and flood predictions are generated within seconds. Improvements can likely be obtained by ensuring a balanced representation of temporal and spatial rainfall patterns in the training dataset, further improved spatial input datasets, and by linking U-FLOOD to dynamic sewer system models.

1. Introduction

Maps of pluvial urban flood hazard with high resolutions finer than 10 m are used for planning city layouts and water infrastructure, as well as in flood warning systems that are used to direct emergency services. For planning purposes, it is increasingly recognized that a variety of flood adaptation options need to be evaluated in a variety of scenarios of, for example, climate change and city development, to identify cost-effective and robust solutions (Bach et al., 2020; Löwe et al., 2017; Webber et al., 2019). Participatory planning approaches (Voinov et al., 2016) require fast and easy-to-use flood screening solutions, that enable non-experts such as architects to assess hazards for different city layouts. In addition, flood warning systems need to generate hazard maps from rainfall forecasts within a few minutes and possibly also quantify the effect of uncertain rainfall forecasts (Hofmann and Schüttrumpf, 2019; Li and Willems, 2020; Meneses et al., 2015).

Hydrodynamic models are the state-of-the-art for the assessment of urban pluvial flood hazard and are available in a variety of commercial software packages (Deltares, 2017; DHI, 2016; Innovyze, 2020). These models dynamically simulate the movement of water on the terrain surface in space and time, and are frequently linked to a dynamic simulation of water movement in the sewer network. All of the applications named in the previous paragraph demand short simulation times that are difficult to achieve with these models. A number of screening methods were therefore developed. Following the classification in Jamali et al. (2019), these can be divided into approaches that distribute surface water through a network of connected topographic depressions (Balstrøm and Crawford, 2018; Jamali et al., 2018; SCALGO, 2020), and

* Corresponding author.
E-mail address: rolo@env.dtu.dk (R. Löwe).

https://doi.org/10.1016/j.jhydrol.2021.126898
Received 17 March 2021; Received in revised form 17 June 2021; Accepted 28 August 2021
Available online 4 September 2021

approaches that apply cellular automata to distribute water on a raster surface (Guidolin et al., 2016; Jamali et al., 2019). These approaches can achieve substantial speedup factors up to 1000, but they are challenged by not considering time dynamics. Thus they have difficulties in distinguishing, for example, the effects of rain events with the same rainfall depth but different hyetographs. Approaches that do consider time dynamics achieve lower speedup factors in the order of 2 to 8 (Guidolin et al., 2016).

A third group of screening approaches that address flood hazard from a data-driven rather than a conceptual viewpoint has arisen in recent years. Roughly, these can be grouped into approaches that predict flood hazard based on rainfall input only, and approaches that consider physical catchment characteristics as input to the model. The former group of models can only be applied for locations that were included in the training dataset. Recent examples for such approaches are Berkhahn et al. (2019), who successfully trained feed-forward neural networks against urban pluvial flood maps generated by a hydrodynamic model; Bermúdez et al. (2018), who used a combination of feed forward networks to detect flooding from the sewer system and simulate maximum flood volume, which is then used to choose a discrete pre-simulated flood map; and Kabir et al. (2020) and Lin et al. (2020), who predicted fluvial urban flood maps using convolutional and feed-forward neural networks, respectively.

Data-driven approaches can also consider static catchment properties as input, and in this way be used to generate predictions for areas that were not included in the training dataset. Kratzert et al. (2019) were able to outperform established hydrological models in such a setting when simulating river flows using long short-term memory neural networks. Urban pluvial flooding occurs scattered throughout a city and data-driven prediction models therefore need to consider rather large amounts of data that characterize the urban layout. Deep learning, in particular using convolutional neural networks, can extract spatio-temporal features from data and is therefore attractive for predicting pluvial flood hazards. Originally developed for computer vision (e.g. He et al., 2016; Isola et al., 2017; Ronneberger et al., 2015), these techniques were also successfully applied for a number of problems in Earth sciences (Reichstein et al., 2019) and related applications such as wind power prediction (Zhu et al., 2020). Examples from hydrology include Pham et al. (2020), who used deep belief networks which, based on 15 discrete input variables that characterize the catchments, classify the degree to which a location is susceptible to flooding, and Zhao et al. (2020), who used convolutional neural networks with 9 continuous input variables for the same purpose. Considering the prediction of pluvial flood water depths in urban areas, Guo et al. (2021) trained a convolutional autoencoder with 4 input variables describing topography against 2D flood depth maps that were simulated by a hydrodynamic model. In a similar setting, Zahura et al. (2020) trained random forest regression models that predict flood water depth on discrete street sections based on 3 variables that characterize topography.

Extending the aforementioned work, our study makes three main contributions:

1. We use deep learning to predict flooding for locations and rain events that have not been included in the training dataset, and we only consider historical hyetographs for training and validation. While studies focussing on flood susceptibility validated their models on datasets that included new locations, the studies focussing on pluvial urban flood depths tested their models for rain events that were not considered in the training datasets, but not new locations. An out-of-sample evaluation of prediction performance is therefore not available so far. In addition, many of the existing flood screening tools (Berkhahn et al., 2019; Jamali et al., 2019; Thrysøe et al., 2021) were only tested on design storms, which may lead to optimistic conclusions.
2. We perform a comprehensive evaluation of which spatial input variables should be considered in the deep learning model when predicting 2D flood maps. Studies that considered comprehensive sets of input variable candidates focused on flood susceptibility in natural catchments (Avand et al., 2020; Pham et al., 2020; Zhao et al., 2020), i.e. not water depth maps, and a larger spatial scale than for the urban case. Studies focusing on water depth prediction in urban areas have not considered topographic input (Berkhahn et al., 2019), or focussed on a very limited set of variables (Guo et al., 2021; Zahura et al., 2020). Only Zahura et al. (2020) evaluated the importance of different input features, however, in a context of predicting water depth on selected street segments instead of 2D flood maps.
3. We systematically analyse the predictive performance of different deep learning configurations with varying complexity. While the studies focussing on flood susceptibility did perform meta-optimization or compared different network architectures (Avand et al., 2020; Pham et al., 2020; Zhao et al., 2020), this assessment is missing for the prediction of pluvial urban flood maps.

The aim of our study was to develop a deep learning network that can be used to predict 2D maps of maximum flood depth based on readily available geodata and rainfall input. If successful, this approach enables the creation of fast and accurate flood screening tools that can be trained based on a limited number of simulations from detailed physical models, and that avoid problems in distinguishing effects from rain events with similar rainfall depth but different hyetographs. To facilitate further research based on our work, we exclusively used openly available geodata for training the deep learning models that we made available along with our computer code.

2. Material and methods

To create a deep-learning approach for predicting pluvial flood hazards, we proceeded in three steps:

1. Identify the (spatial) input variables that are likely to yield the best prediction of flood hazard from a set of potential variables.
2. Evaluate the required complexity of the neural network to achieve reliable flood predictions.
3. Validate the predictive performance of the network in a k-fold cross validation procedure to obtain a reliable estimate of out-of-sample prediction capacity.

Note that in steps 1 and 2 model performance was also evaluated on a validation dataset using a simple holdout procedure. In the following subsections we first describe the considered case area and training data, then we illustrate the applied deep learning setup and finally we describe which experiments were performed in each of the three steps outlined above and how model performance was assessed.

2.1. Materials

2.1.1. Case area, flood simulations and geodata

We considered the city of Odense in Denmark as a case study. The city has approximately 200,000 inhabitants and is located in a typical moraine landscape close to the sea. We obtained terrain and landuse data from the Danish geodata portal Kortforsyningen (Agency for Data Supply and Efficiency, 2020; Agency for Data Supply and Efficiency and Danish Municipalities, 2020).

Fig. 1 shows the study area. The entire area covered 3740 × 4273 pixels in a resolution of 5 m. This resolution was deemed sufficient for flood screening purposes (Berkhahn et al., 2019; Löwe and Arnbjerg-Nielsen, 2020) and was applied for all datasets in the hydrodynamic simulation as well as the neural networks. Elevations in the terrain dataset were raised by 5 m in building locations. Only the urban areas (highlighted red and blue in Fig. 1) were considered when training and validating the neural networks, as the focus of the study was pluvial

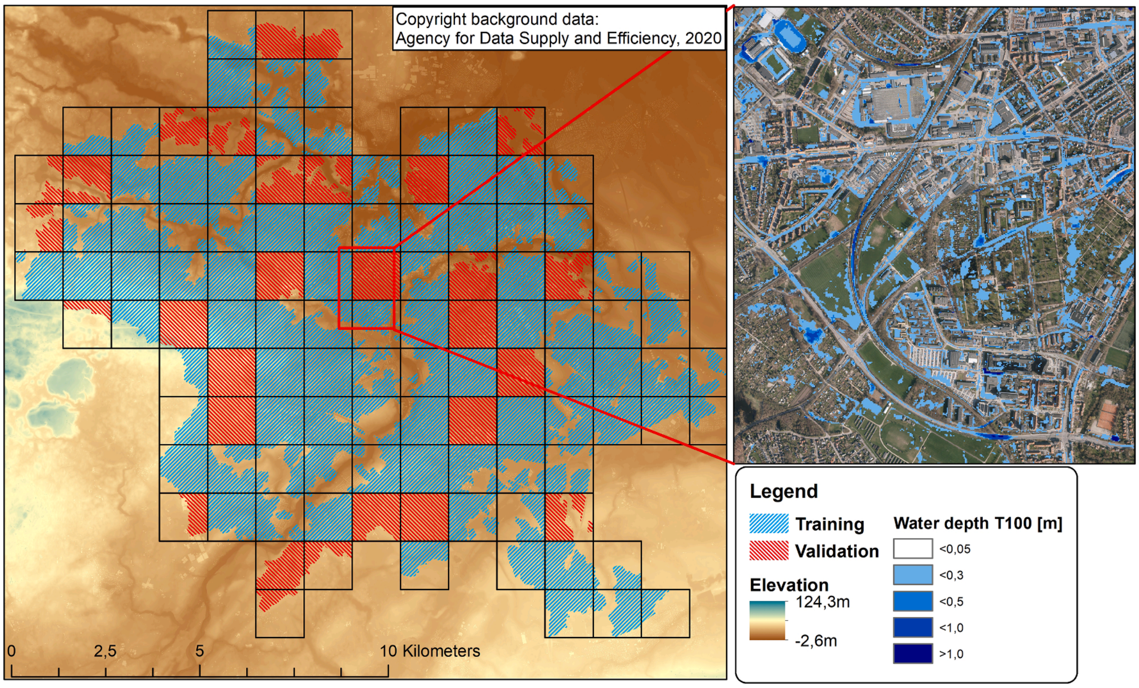


Fig. 1. Case study area in the city of Odense (Denmark). Highlighted squares are 256x256 pixels (1280x1280m). Only flooding occurring within the simulation area (marked blue and red) was considered in the training and validation of the neural network. Input data and flood depths were set to 0 for all other areas. The entire study area comprises 3740x4273 pixels with an edge length of 5 m. The area was subdivided into 119 squares to mark areas for model validation. Squares marked in red were used for validation in steps 1 and 2 of the model development procedure. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

urban flood hazard. Further, flooded areas related to natural streams and marine flooding were excluded from the analysis as these types of flooding are related to different physical processes and spatio-temporal scales than pluvial flooding. The area was subdivided into 119 squares to mark areas for model validation. Squares marked in red were used for validation in steps 1 and 2 of the model development procedure, while random subsets of all 119 squares were used for validation in step 3 (Sections 2.1.3 and 2.2.4).

We performed hydrodynamic flood simulations in 2D for the entire area shown in Fig. 1 using MIKE 21 (DHI, 2016). Following the same approach as Guo et al. (2021), Kaspersen et al. (2017) and Webber et al. (2019), we considered rainfall on each pixel of the 2D surface and performed a runoff computation for each individual pixel as outlined in the following. For each pixel the pervious and impervious area inside the pixel was computed, assuming that buildings and roads are 100% impervious and that all other areas are pervious. We assumed a wetting loss of 0.6 mm for each rain event, a maximum sewer system capacity of 12 mm/hr for precipitation on impervious areas (Webber, 2019), and a constant infiltration capacity of 29.3 mm/hr for precipitation on pervious areas (Kaspersen et al., 2017). In each simulation time step (1 s), separate effective precipitation rates were computed for pervious and impervious areas by subtracting the above losses from the observed rainfall. In each pixel, the two effective precipitation rates were multiplied by the impervious and pervious area inside the pixel, and then summed up to obtain the total runoff from the pixel. Subsequently, runoff was then routed over the terrain surface by the 2D hydrodynamic simulation. This approach to runoff computation generates realistic flood maps that can be used to demonstrate the feasibility of our deep learning approach. The sewer system, however, is represented in a simplified manner that does not necessarily reflect the effect of, for example, bottlenecks in the sewer pipes. We have chosen this approach, because it enables us to make all the data used in our study accessible. This would not be the case when performing 1D-2D flood simulations where proprietary information on the sewer network is required.

Hydrodynamic simulations were performed for each of the 53 rain events described in the following section. Fig. 1 shows an example of the generated flood maps. When training the deep learning models for the prediction of flood depths, we set any flood depths below 0.05 m to 0 as these water depths are not relevant for neither economic damage assessments nor warning systems. We have in addition removed puddles, defined as flooded areas consisting of less than 5 connected pixels, by applying a sieve filter (GDAL Development Team, 2020). Terrain data used for the simulations and flood maps are available from (Löwe, 2021).

2.1.2. Rainfall data

We considered rainfall observations in 1 min resolution from ten rain gauges distributed across Denmark that have been in continuous operation for at least 40 years. For each of the stations we identified rain events. Rain events were defined to start when a dry weather threshold for the rain intensity of 0.1 mm/h was exceeded, and to end when this threshold had not been exceeded for at least six hours. From each station we extracted the five rain events with the highest average rainfall intensity over a 30 min period. Where events were observed at multiple stations on the same day, we kept only the most intense event. This process resulted in a dataset of 43 rain events that were assumed representative for the types of events that can lead to pluvial flooding in Denmark.

A data-driven model needs to not only predict flood depth in extreme events, but also to distinguish events when flooding does or does not occur. We have therefore extended the rainfall dataset with an additional 10 events with medium to small rain intensities. These events had maximum rain intensities of 10 to 30 mm/hr averaged over a 30 min period. One such event was manually selected from each station, resulting in a total of 53 events.

Section S1 in the Supporting Information includes a table of key characteristics for each of the rain events, a time series plot of rain intensities for each event, as well as a histogram of the rain intensities in the dataset. Rain intensities were averaged to 10 min intervals, to limit the required storage for the hydrodynamic simulations, where rainfall input needed to be provided as a time series of 2D maps, each with a spatial extent of 3740x4273 pixels.

2.1.3. Training and validation data

Our dataset consisted of 53 flood maps covering the entire study area shown in Fig. 1 (one map per rain event). For training and validation, the network was presented with square “snapshots” of these maps that had a fixed spatial extent of 256x256 pixels (5 m resolution). For pixels outside the simulation area (neither red nor blue in Fig. 1), both input and output data were set to zero. The procedures for generating training and validation datasets for steps 1 and 2 of the model development procedure are outlined below. Separate datasets were generated for cross-validation in step 3 as detailed in Section 2.2.4.

2.1.3.1. Training dataset. The training dataset was created by randomly sampling combinations of rain events and spatial patches of 256x256 pixels. Other than the so-called squares used for validation (see below), these patches were placed randomly throughout the simulation area. We sampled 10,000 times by a) randomly selecting one of the 48 rain events not used for validation (with repetition, i.e. irrespective of whether the event was selected in a previous iteration), b) sampling a patch (256x256 pixels) at a random location anywhere in the study area. We then checked if the patch contained if the patch contained areas outside the blue areas in Fig. 1. Such areas were either not part of the simulation area or part of the validation data (red in Fig. 1) and should not be used for training. Input data and flood depths were set to 0 in the corresponding pixels. If the patch contained less than 20% valid pixels, it was discarded and a new patch was sampled. Otherwise, the patch was added to the training dataset.

The 10,000 patches corresponded to randomly located extracts of the 48 flood maps.

Each patch covered a unique part of the study area, but overlapped with other patches. Similar “data augmentation” procedures are commonly applied in image classification to reduce over-fitting (Rawat and Wang, 2017). The Supporting Information S4 contains an illustration of how often different parts of the study area were sampled in the training dataset.

2.1.3.2. Validation dataset for steps 1 and 2 of the model development procedure. We sub-divided the study area into 119 squares of 256x256 pixels with fixed locations (Fig. 1). All squares contained at least 20% pixels inside the simulation area (blue and red in Fig. 1). Subsequently, we a) randomly selected 29 of these squares (1/4 of the study area) (marked red in Fig. 1), b) manually selected five rain events that reflected varying rain intensities and temporal rainfall patterns. We have highlighted these events in Table S1 in the Supporting Information.

Flood maps within the 29 squares were never presented to the neural network in the training phase (in any rain event). Similarly, flood maps for the five rain events were never presented during training (in any location). Model validation was thus performed on locations and rain events that were unknown to the network. In total, the validation dataset consisted of 29*5 = 145 flood maps.

2.2. Data-driven model development

2.2.1. Modelling setup

Fig. 2 illustrates the deep learning setup applied in our study. 2D arrays of spatial inputs with a fixed extent of 256 × 256 pixels are processed through an encoder/decoder structure consisting of pairs of convolutional layers (panel A). Rainfall input is pre-processed and concatenated at the bottleneck of the spatial convolution (panel B). After the last decoding block, the network generates a prediction of maximum water depth in each individual pixel of within the considered patch for a given rain event.

The overall structure resembles the framework proposed by Guo et al. (2021), but we have made a number of distinct changes:

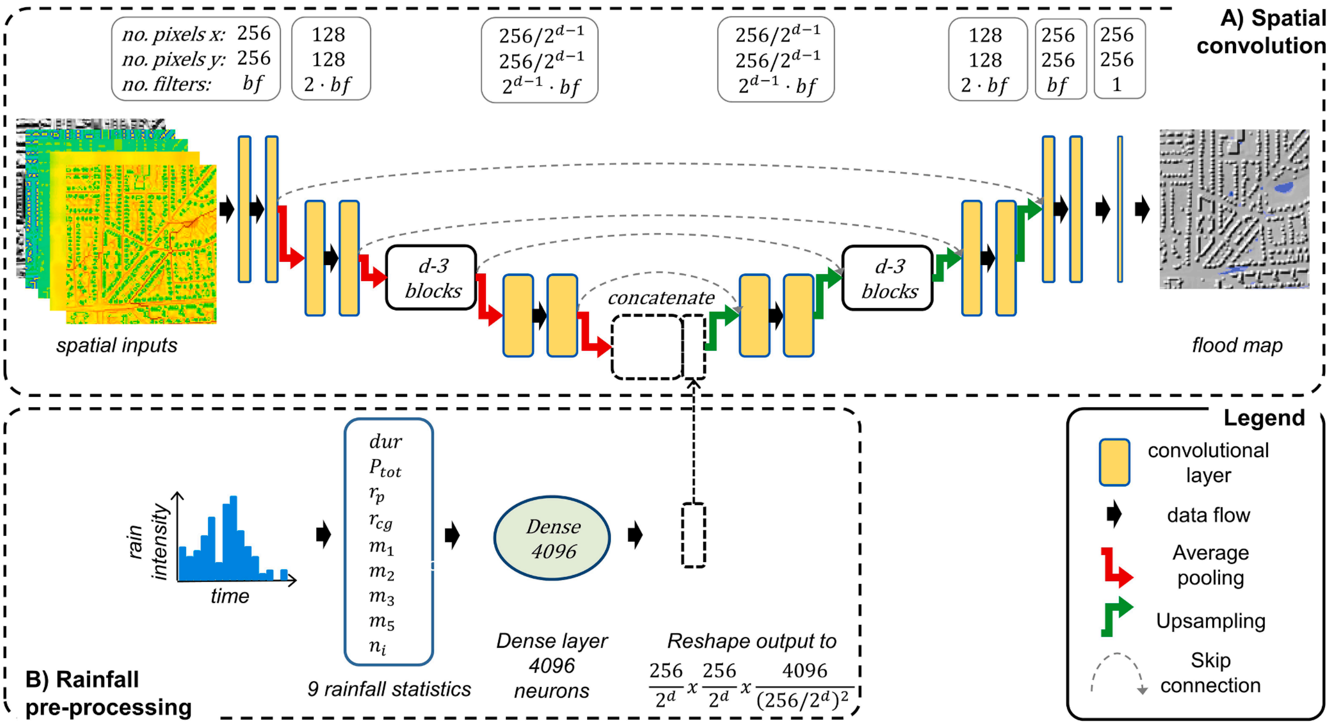


Fig. 2. Schematic of the applied modelling setup. Panel A: Spatial inputs are processed through a sequence of 2D convolutional blocks that are linked through average pooling. The network depth d corresponds to the number of encoding steps. The first convolutional block considers a number of bf filters. Panel B: Rainfall series are converted to 9 rainfall statistics. These are processed by a fully connected layer and concatenated to the spatial convolution at the narrowest part of the network.

1. Skip connections concatenate the output from an encoding block in the spatial convolution to the input of the corresponding decoding block, thus bypassing encoding steps in the deeper parts of the network. Skip connections were demonstrated to support smooth objective function landscapes and thus better convergence of deep neural networks (Li et al., 2018). With this modification, the spatial convolution in our framework strongly resembles the U-Net architecture (Ronneberger et al., 2015) which is widely applied for image segmentation.
2. We converted the input rainfall series into a set of 9 rain event characteristics before feeding it into the neural network. Preliminary work showed that such characterizations in combination with simple feed forward networks (Böhm, 2020; Eriksen and Dichmann, 2019) achieve strong performance in predicting flood extents. Directly feeding the rainfall series into a dense layer (Guo et al., 2021) leads to an excessive number of parameters when long rain events should be considered (up to 34 h in our dataset), and creates difficulties during training when the rainfall peak does not always occur at the same time point in the rain series.
3. We applied average pooling (Amidi and Amidi, 2019) in the encoding part of our network as it yielded stronger performance in initial tests (Supporting Information, Section S6) than the maximum pooling applied in other works (Guo et al., 2021; Ronneberger et al., 2015).

We considered three hyper-parameters for our setup:

- d –depth of the network - corresponds to the number of encoding / decoding steps,
- bf –number of filters considered in the convolutional layers of the first encoding step - In line with similar architectures (Ronneberger et al., 2015), the number of filters is increased by a factor of 2 in each encoding step. We have, however, considered a maximum number of 512 filters to limit the computational requirements, and
- k – edge length in pixels of the kernels applied in the convolution operations.

Leaky-ReLU activation functions (Maas et al., 2013) were considered for all layers with an activation threshold of 0.2. Like other works with similar architectures (Badrinarayanan et al., 2017; Chattopadhyay et al., 2020; Höhlein et al., 2020), we applied dropout regularization to the convolutional layers with a dropout rate of 0.5.

A patch size of 256x256 pixels (1280x1280m) was considered for the spatial input data and the generated flood maps. This size is frequently applied with convolutional networks, because it is a power of 2 and repeated pooling operations (that divide the size of the images by 2) can thus be performed without padding the pooling outputs. Patches with an edge length of above one kilometer should be sufficiently large to capture the most important effects leading to urban pluvial flooding that occurs on a very local scale (Löwe et al., 2020), while at the same time being sufficiently small to not cause memory problems during model training. A potential downside of working with fixed size images is that hydrological objects such as sinks will be cut off when located near the edge of a patch. This problem is mitigated by considering training patches that are not placed regularly (as in Fig. 1) but randomly throughout the study area.

2.2.2. Model development step 1 – feature definition and selection

2.2.2.1. Rainfall input. Urban pluvial flooding is linked to small spatial scales and short time scales. The amount of runoff varies depending on the shape and intensity of the specific rain event (e.g. Davidsen et al. (2017); Müller et al. (2017)). These characteristics need to be provided to enable the neural network to distinguish the effects of different rain events. Urban hydrologists have a long tradition of characterizing natural rain events with standardized characteristics that can be used to test the design of sewer systems with representative design storms (Jean et al., 2018). Inspired by rainfall characterizations from German design guidelines, Wartalska et al. (2020) used the following set of statistics to characterize the temporal distribution of rain depths during an event, which we have considered as input to our model:

- r_p – time index of the rainfall peak relative to the total duration of the event
- r_{cg} – time index of the median accumulated rainfall relative to the total duration of the event
- m_1 – ratio of cumulative precipitation before vs. after the rainfall peak
- m_2 – ratio of maximum rain intensity (10 min interval) vs. total rainfall depth
- m_3 – ratio of rainfall depth in the first third of an event vs. total rainfall depth
- m_5 – ratio of rainfall depth in the first half of an event vs. total rainfall depth
- n_i – ratio of maximum rain intensity (10 min interval) vs. average rain intensity

In addition to the above, we included the accumulated rainfall depth P_{tot} and its duration dur to characterize the magnitude and duration of an event. The statistics computed for each rain event are available in Section S1 of the Supporting Material.

2.2.2.2. Spatial input. When training a neural network against maximum water depth maps, the network does not learn the dynamics of water movement on terrain and spatial inputs that condense the hydrological characteristics of the catchment are likely to yield higher predictive power than elevation data alone. Based on expert reasoning and previous studies, we defined a set of 11 spatial variables that we considered potentially relevant for the prediction of urban pluvial flooding in a data-driven model. Table 1 summarizes these variables and the reasoning for their inclusion. Common for all input variables was that they must be possible to derive from geodata using standard raster processing operations to ensure that the deep learning model, once fitted, can be readily applied to new locations.

Several spatial datasets were characterized by long-tailed, right-skewed distributions where flood hazard does not change much whether the explanatory variable takes medium-large or extreme values. In line with standard mathematical modelling procedures (Brockhoff et al., 2018; Madsen, 2008), we applied data transformations in these cases. Subsequently, all variables were scaled to the intervals [-1,1] if negative values were present and [0,1] otherwise. Visualizations of all variables are provided in the Supporting Information, while the actual datasets can be obtained from Löwe (2021).

For the spatial inputs it is particularly relevant to consider only the datasets that actually improve prediction performance of the model. Too many model inputs will increase the computational demand in the training phase and the risk of overfitting. Based on the selection of input variables in Table 1, we employed two approaches:

1. Spearman's ranked correlation (Dodge, 2008)
   We computed the ranked correlation between simulated water depths and the input variables in Table 1. The computation was done on a pixel-by-pixel basis for the entire area highlighted in Fig. 1, and a separate correlation coefficient was estimated for each of the considered rain events, i.e. 53 correlation coefficients were obtained for each input variable. This approach provided insight on whether an input variable would be directly related to simulated water depths in a monotonic manner. However, it could not provide information on whether a variable enhances the predictive capacity of another variable. In addition, the pixel-by-pixel comparisons imply that the

Table 1
Spatial explanatory variables used for the deep learning model for urban pluvial flood hazard.

Variable	Data adjustments (listed in order of application)	Reasoning
DEM	Scaled to [0,1]	Surface elevation including buildings. Used in Guo et al. (2021); Pham et al. (2020); Zahura et al. (2020); Zhao et al. (2020)
ASP	Scaled to [-1,1]	Characterizes flow direction on terrain. Considered as 2 separate raster datasets (cosine and sine of aspect) to handle cyclic behaviour of flow direction (Guo et al., 2021). Used in (Guo et al., 2021; Pham et al., 2020)
CURV	Cuberoot transformed Scaled to [0,1]	Characterizes concaveness / convexity of terrain. Transformation is applied to reduce extremely leptokurtic distribution of values. Used in (Guo et al., 2021; Pham et al., 2020)
DEM_L	Scaled to [0,1]	DEM minus the focal mean of DEM within 100 m radius. Pluvial urban flooding is linked to spatial scales <1 km (Löwe et al., 2020) and should therefore be linked to local variations in elevation, rather than elevation above sea level, i.e. the elevation signal for a flat urban area in the mountains should be the same as for a flat urban area close to the coast. Not used in any previous studies.
SDEPTH	Scaled to [0,1]	Water depth in terrain sinks. Computed as difference between elevation of the outlet point of a sink and terrain elevation. 0 for all cells located outside sinks. Not used in any previous studies.
IMP	–	Imperviousness in each raster cell, ranging from 0 to 1. Affects amount of runoff generated from the cell. Computed from building and road data that are both assumed 100% impervious. (Zhao et al., 2020) used a related index based on Landsat data.
SLOPE	Scaled to [0,1]	Terrain slope, computed based on the focal mean of terrain elevation within 100 m radius to ensure that the hydrologic behaviour of an area is captured, not the location of edges of buildings, curbs or similar. These features should already be captured by DEM, ASP and CURV. Used in (Guo et al., 2021; Pham et al., 2020; Zhao et al., 2020)
FLACC	Cutoff at 250ha Cuberoot transformed Scaled to [0,1]	Number of cells flowing into a given pixel. Describes the likelihood of a depression to be flooded. Very large accumulation values are linked to natural streams. We have therefore defined an upper cutoff at 250 ha. Values follow a leptokurtic distribution and are therefore transformed. Previously used by (Pham et al., 2020).
FLIMP	Cutoff at 25ha Cuberoot transformed Scaled to [0,1]	Total impervious area upstream from a given cell. Computed by weighting the computation of FLACC with IMP value between 0 and 1 in each cell. Defines the amount of runoff that should be expected in small rain events. Very large values are linked to natural streams, so a cutoff was defined at 25 ha. Values follow a leptokurtic distribution and are therefore transformed. Not used in any previous studies.
FLSLO	Cutoff at 250 ha Cuberoot transformed Scaled to [0,1]	Flow accumulation weighted by the SLOPE in each cell. Used as expression of average flow velocity on the path towards a cell that could quantify ratio between infiltration and runoff. Not used in any previous studies.
TWI	Squareroot transformed. Scaled to [0,1]	Defined as $\ln(\alpha/\tan(\beta))$ with α being the contributing area per unit contour length and β the local terrain slope (Beven and Kirkby, 1979). Measures the tendency of an area to accumulate runoff. Used by (Pham et al., 2020; Zahura et al., 2020; Zhao et al., 2020).

method could not capture effects where the input variable, for example, needs to be aggregated in space before an impact on the water depths becomes clear.

2. Forward selection
   To address the above short-comings, we employed a procedure that was inspired by stepwise regression modelling procedures (Brockhoff et al., 2018; Pardoe et al., 2020). We used the model illustrated in Fig. 2 with hyper-parameters depth $d = 4$, number of filters in the first encoding block $bf = 32$ and kernel size $k = 3$. This configuration was considered as a reasonable starting point as it performed well in initial tests and as it has a similar complexity as the setup employed by Guo et al. (2021). We trained 11 different neural networks with this configuration, each of the them considering a single spatial input from Table 1. We evaluated each networks' predictive performance based on the scores described in Section 2.3 and selected the one that performed best. Based on this network, we created 10 new networks, all of which considered the successful input from step 1 and one of the remaining 10 inputs. The procedure was repeated until a subjective inspection of the results suggested that the inclusion of additional input variables no longer improves predictive performance on the validation dataset.

2.2.3. *Model development step 2 – network complexity*

To evaluate what is a parsimonious neural network for obtaining reliable predictions of flood water depth, we performed a grid search on the hyper parameters listed in Section 2.2.1. We varied

- the network depth d between 2 and 6, where a network with $d = 6$ consists of 6 encoding blocks and six decoding blocks, each consisting of two convolutional layers,
- the number of filters bf in the first convolutional layer between 16, 32 and 64, and
- the size k of the kernels applied in the convolution layers between 3, 5 and 7.

Larger values for d and bf increase the number of parameters in the network that need to be estimated, but increase its flexibility to reproduce patterns. Larger kernel sizes enable the convolutional layers to make use of spatial information from a larger area, but strongly increase computational expense. Many recent popular network architectures employ $k = 3$ (Chen et al., 2018; He et al., 2016; Ronneberger et al., 2015). The hyper-parameter ranges outlined above led to neural networks that had from 97,000 up to 260,000,000 parameters that needed to be estimated.

2.2.4. *Model development step 3 – k-fold cross-validation*

To obtain a more robust estimate of predictive model performance, we selected the best performing model architecture from step 2, and trained it in a k-fold cross-validation procedure with five folds. For this purpose, we created five non-overlapping validation datasets, each consisting of 10 rain events and 24 of the squares shown in Fig. 1. The squares were randomly assigned to the validation datasets. A stratified sampling approach was applied for the rain events. We sorted the rain events by their maximum 30 min rainfall intensity and then divided them into 10 groups, each containing 5 events (the three least intense events were not used for validation). Each of the 5 events in each group was then randomly assigned to a different validation set. For each of the five folds, a separate training dataset was sampled following the same procedure as outlined in Section 2.1.3. We did again consider permutations of size $n = 10,000$.

2.2.5. Objective function and optimization

All networks were fitted by minimizing the mean-squared error (MSE) between the maximum depth flood maps predicted by the neural network and those generated in the hydrodynamic simulation. In this process, a square root transform was applied to the flood depths to reduce the skewedness of the distribution of simulated flood depths.

We applied the Adam optimizer over 500 epochs. As suggested by Smith (2017), a triangular learning rate schedule was considered where the learning rate over a period of 10 epochs varied between an upper limit and a lower limit of $5{\cdot}10^{-5}$. The upper limit exponentially decreased from a starting value of 10^{-3} as a function of $0.95^{epochnumber}$. The implementation of the learning rate scheduler is available in Löwe (2021). The triangular learning rate could avoid divergence of the networks during initial tests where skip connections had not been implemented. Once a suitable architecture had been implemented, it was not deemed essential.

2.3. Performance evaluation

To assess the performance of the data-driven models in the various steps of the model development procedure, we compared maximum flood depth maps generated by the hydrodynamic simulation against flood maps predicted by the neural network and computed the score values shown in Table 2. Score values were computed considering only those pixels where either the hydrodynamic model (HD) or the neural network (NN) predicted flooding. Similar to Jamali et al. (2019), a depth threshold *thr* of 0.05 m was applied for all scores. For the critical success index (*CSI*), separate score values were computed for thresholds of 0.05 m and 0.3 m to distinguish prediction accuracy in areas with greater water depth.

Table 2
Score values used for measuring the level of agreement between maximum water depths predicted by neural network and simulation from the hydrodynamic model. References indicate related studies where the indices were used.

Score	Purpose	Equation	Range	Best value
$RMSE$[m] (Jamali et al., 2019)	Average deviation of prediction water depths	$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\sum_{j=1}^{5}\left(Y_i^{NN} - Y_i^{HD}\right)^2}$	$0 - \infty$	0
CSI_{thr}[%] (Bennett et al., 2013; Davidsen et al., 2017b; Jamali et al., 2019, 2018)	Binary comparison on pixel to pixel basis	$CSI_{thr} = \frac{H_{thr}}{H_{thr} + M_{thr} + FA_{thr}}$	$0 - 1$	1
NSE[-] (Berkhahn et al., 2019)	Create maps for variation of prediction accuracy in space	$NSE = 1 - \frac{\sum_{i=1}^{n}\sum_{j=1}^{5}\left(Y_{i,j}^{NN} - Y_{i,j}^{HD}\right)^2}{\sum_{i=1}^{n}\sum_{j=1}^{5}\left(Y_{i,j}^{NN} - \overline{Y_i^{HD}}\right)^2}$	$-\infty - 1$	1
A_{NN}/A_{HD}[-]	Ratio of total area flooded >0.05 m	$\frac{A_{NN}}{A_{HD}} = \frac{H_{0.05} + FA_{0.05}}{H_{0.05} + M_{0.05}}$	$0 - \infty$	1

Y_i^{NN},Y_i^{HD}- water depths predicted in *i*-th pixel by neural network (NN) and hydrodynamic model (HD) in validation rain event *j* (5 events in total).
H_{thr}, M_{thr},FA_{thr} – hits (water depth in pixel above threshold *thr* in NN and HD), misses (water depth in pixel above threshold only in HD), false alarms (water depth in pixel above threshold only in NN).
$\overline{Y_i^{HD}}$ – average of the maximum water depths computed for pixel *i* in all 5 validation rain events *j*.

2.4. Technical implementation

Neural networks were implemented in Tensorflow version 2.3.1 (Abadi et al., 2016) using the interface with Python 3.8.2. Model training was performed in a high performing computing environment using Nvidia Tesla V100-PCIE GPU's with 16 and 32 GB memory.

3. Results

3.1. Step 1 – feature selection

3.1.1. Spearman correlation

Fig. 3 illustrates the ranked correlation coefficients between the candidate input datasets introduced in Table 1, and the simulated water depths in the 53 considered rain events. One correlation coefficient was computed per input dataset and per rain event.

Not very surprisingly, the figure suggests a clear correlation between the datasets characterising local deformations of the terrain and water depth. Pluvial flood water accumulates in local depressions, and thus locations with elevations that are lower than the neighbourhood average (DEM_L) and locations inside sinks (identified by positive values of SDEPTH as well as TWI) are good predictors of high water depths. For the same reason, we could identify a strong correlation between water depth and the CURV dataset, which distinguishes concave and convex regions of the terrain (Supporting Information, Figure S12). In addition, it was noticeable that DEM_L exhibits a substantially stronger correlation with water depth than DEM.

Slightly lower correlations were observed for the input variables that characterize the size of the upstream area contributing to the water flow at a given location (FLACC, FLIMP, FLSLO). These variables should be

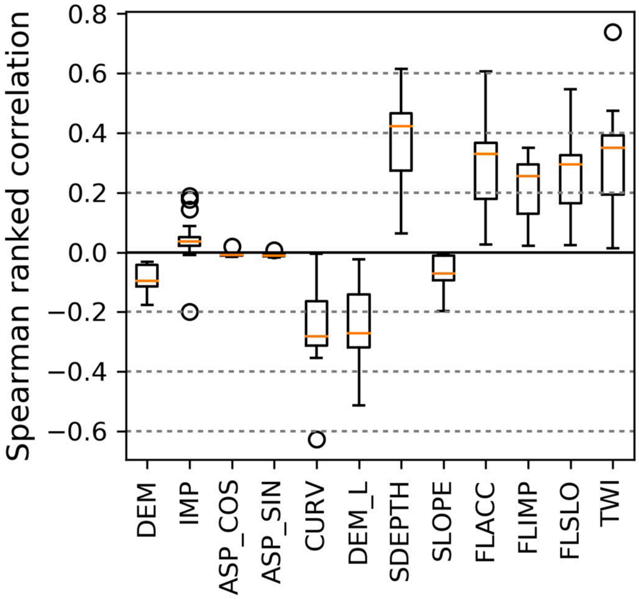


Fig. 3. Spearman ranked correlation between simulated water depth and candidate input datasets for the neural network models. Correlation values were computed separately for each of the considered rain events, and the boxplot shows the spread of correlation values across events.

useful to explain the amount of water and thus the frequency of flooding at a given location, but the main flow paths form narrow lines in these datasets that are not linked to, for example, the extent of sinks in which water will spread out. In a direct pixel-to-pixel comparison, these variables therefore have lower predictive power. In a similar manner, SLOPE can be used to identify flat regions where water is likely to accumulate, and IMP contains some information on the amount of runoff generated in an area, but neither can be used to point out the exact extent and locations of the flood areas. Finally, considering Fig. 3, the relationship between the ASP variables and water depth would need to be assumed to be random.

The results in Fig. 3 have to be interpreted carefully with regards to choosing the inputs that should be included in the data-driven model. Several of the input datasets are correlated (e.g. SDEPTH, DEM_L, TWI) and are therefore less likely to supplement each other when combined in a predictive model. Further, input data that do not exhibit correlation with the water depth maps in a pixel-by-pixel comparisons, may increase the predictive power of other variables, or gain predictive power due to spatial aggregation in the convolutional blocks of the neural network. For example, IMP may be useful to distinguish the water depth in different sinks with similar SDEPTH values, and the two ASP datasets can only characterize the terrain when considered in combination.

3.1.2. Forward selection

The aim of the forward selection procedure was to mitigate the limitations outlined in the previous section by assessing whether an input dataset actually increases the predictive power of a neural network. Fig. 4 illustrates the score values computed in the different steps of this procedure. In each panel, the top row corresponds to the first step of the procedure (network with only one input dataset), and the colours in different columns illustrate the score value computed when considering the corresponding dataset as model input. The second row

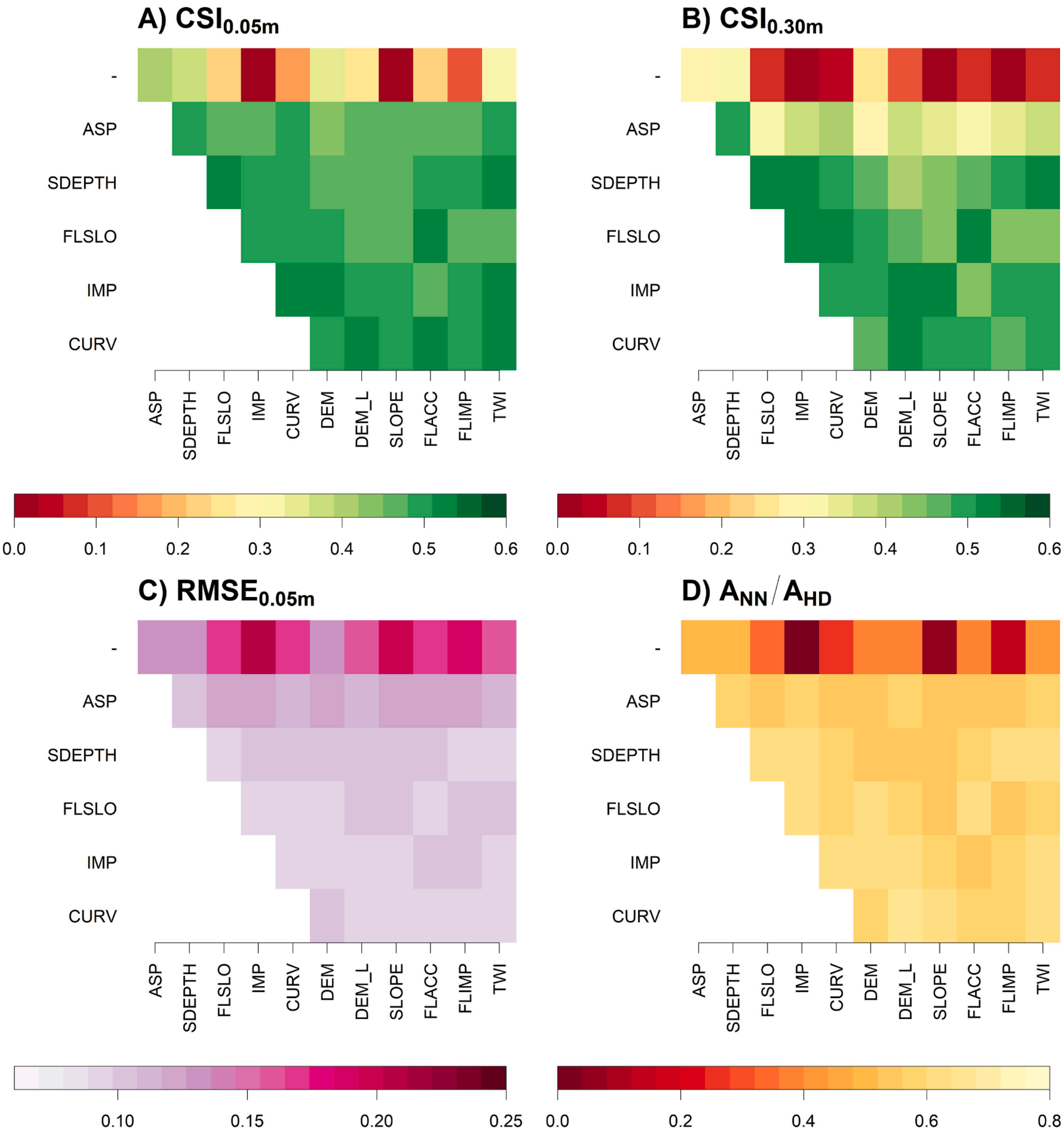


Fig. 4. Score values obtained in the forward selection procedure. From top to bottom row, models included an increasing number of input datasets, and each column shows the score values obtained when adding the corresponding input dataset to the model. For example, all models in the 4th row included ASP, SDEPTH and FLSLO and score values reflect the effect of including any of the other datasets. Panels A and B: CSI scores for water depth thresholds of 0.05 and 0.3 m. Panel C: RMSE for pixels were either hydrodynamic model or neural network predicted water depths above 0.05 m. Panel D: Ratio of total area flooded with depths >0.05 m in the neural network and the hydrodynamic model. Note that color scales vary between panels.

then shows the results for the second step of the procedure, where the best performing input from step 1 was included in the model and combined with each of the other inputs. Note that we considered the two ASP datasets as a single combined entity, as their individual inclusion as input variables has no physical justification. Thus, whenever ASP was included in the model, two datasets (cosine and sine) were included.

Fig. 4 illustrates that, considering a model with only one type of spatial input, the ASP data yielded the best score values on the validation dataset. In addition, the DEM variable yielded similar predictive power as the DEM_L variable. These results are in direct contradiction to the conclusions drawn based on Fig. 3. However, the neural network can process information in a different way than the pixel-by-pixel comparisons performed when computing Spearman correlation or similar measures such as information gain (e.g. Pham et al. (2020)). This underlines that it is important not to solely rely on such measures when choosing the inputs for deep learning algorithms. The ASP data are at an advantage over other variables, as they are represented by two separate datasets and thus provide greater flexibility to represent different trends. Nevertheless, these data consistently yielded an improvement in predictive power, also in experiments where we added them to models with other inputs (not documented), and they were also selected as a key input for modelling water depths by Guo et al. (2021).

As the selection process proceeded, we obtained slight improvements in predictive power. We were not able to achieve a further increase in predictive power once five datasets had been included in the model and stopped the process at this point. The selected input datasets were (in this order) ASP, SDEPTH, FLSLO, IMP and CURV. In the evaluation of predictive power, we focused on panels A-C in Fig. 4, while the ratio of total flooded areas was considered only for information reasons. All neural networks consistently underestimated the total flooded area (see Section 3.4). Networks that performed better in terms of total flooded area, usually achieved this improvement at the cost of an increased number of false predictions, and thus a reduced prediction skill.

Fig. 4 further highlights the correlation between the four datasets characterizing surface flow paths. Neither TWI, FLIMP or FLACC yielded improved prediction skill once FLSLO had been included in the model. For reference, we considered a model where principal component analysis was used to reduce the number of spatial inputs from 11 to 7 orthogonal datasets. This model yielded comparable performance as the one obtained by forward selection (Supporting Information S6).

3.2. Step 2 – optimization of hyper-parameters and residual analysis

3.2.1. Effect of hyper-parameters on model performance

Fig. 5 shows $CSI_{0.05m}$ values computed for different combinations of the hyper-parameter values k, bf and d, considering a water depth threshold $thr = 0.05$ m. Fig. 6 illustrates the training time per epoch for the same combinations of hyper-parameters. A total of 45 hyper-parameter combinations was considered with training times that, for 500 epochs, ranged from 6.5 h to 5 days. The variation of $RMSE$ scores for different hyper-parameter settings is consistent with $CSI_{0.05m}$. An overview of all score values for different hyper-parameter combinations is enclosed in the Supporting Information, Section S5.

From Fig. 5 and Fig. 6 it is clear that kernel edge lengths $k>3$ do not in general improve the predictive performance of the network, while increasing the computational expense. When considering small network depths ($d = 2$ and 3), a small increase of CSI with increasing kernel sizes was apparent for some models. The probable reason would be that larger kernel sizes increase the region over which the network can perform spatial aggregation, which is otherwise quite limited for shallow networks. The trend was, however, not very pronounced.

An increasing number of filters in the convolutional layers (parameter bf) generally improved the model results, but the computational expense also increased roughly linearly with the number of filters. We experienced memory issues when considering $bf > 64$, and have therefore considered this as an upper limit.

Increasing network depth generally led to higher CSI values. However, this was generally only true until depths in $d = 4$ or 5. Deeper networks were over-fitted and thus led to lower CSI values. For $bf = 16$ we also tested greater network depths $d = 7$ which, however, led to a further decline in performance compared to $d = 6$ (Supporting Information S5).

The best performing model was identified to have parameters $k = 3$, $bf = 64$ and $d = 5$. This model can perform spatial aggregation over a region of approximately 500x500 m. Löwe and Arnbjerg-Nielsen (2020) could obtain robust estimates of average imperviousness in an urban area from high-resolution building data at a similar spatial scale.

3.2.2. Flood maps and residual analysis

Fig. 7 compares predicted maximum water depths generated by MIKE 21 and the neural network with $k = 3$, $bf = 64$ and $d = 5$. The displayed water depths were taken from the validation dataset for rain

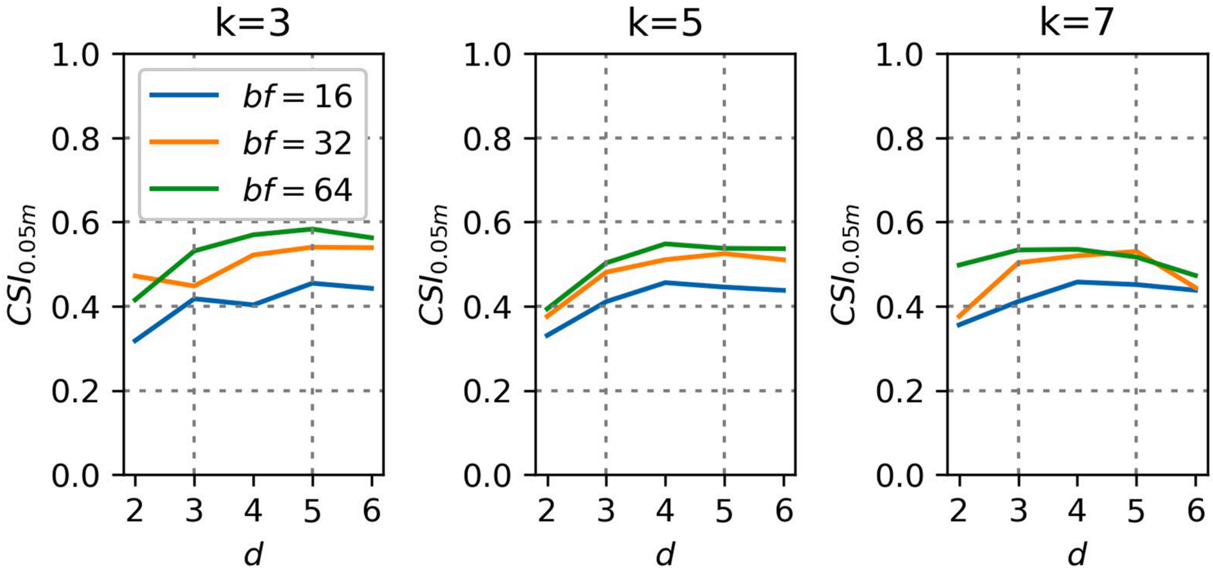


Fig. 5. CSI scores computed for a water depth threshold of 0.05 m when considering different combinations of the network hyper-parameters depth (d), number of filters in the first convolutional block (bf) and edge length of the convolutional kernels (k).

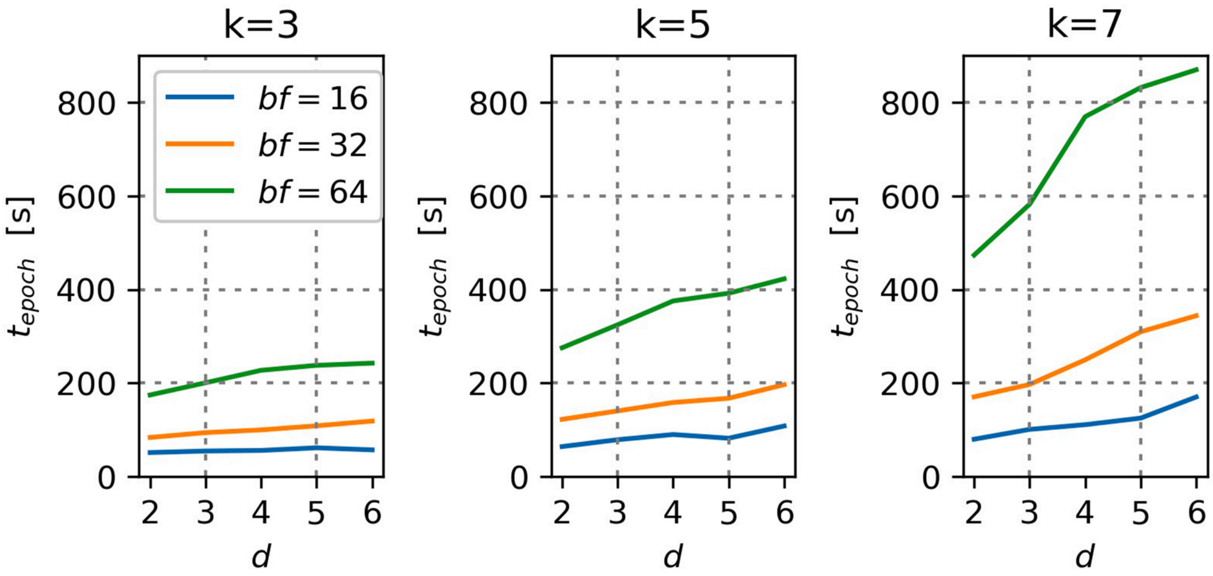


Fig. 6. Training time per epoch (t_{epoch}) when considering different combinations of the network hyper-parameters depth (d), number of filters in the first convolutional block (bf) and edge length of the convolutional kernels (k).

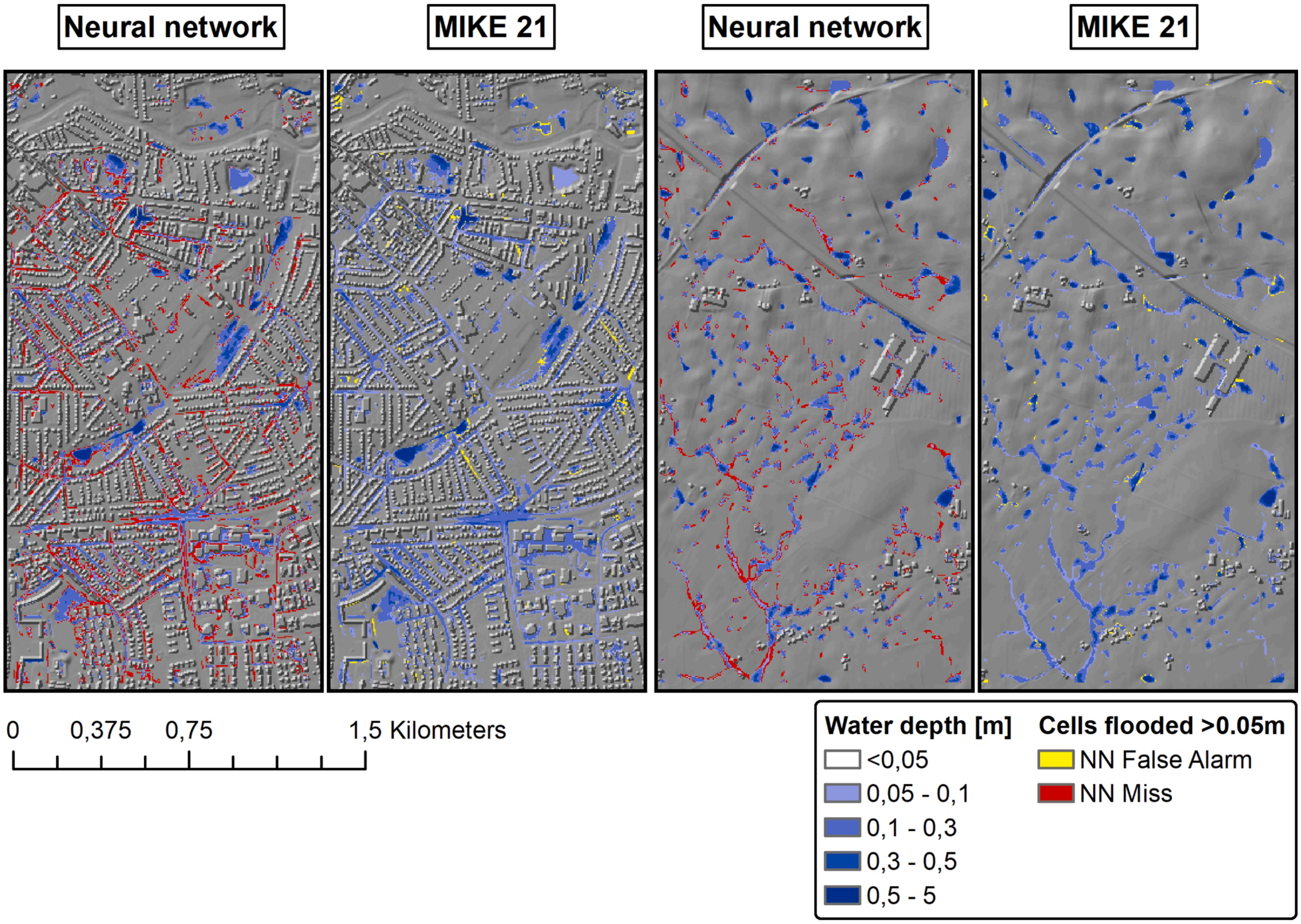


Fig. 7. Water depths predicted by neural network and the hydrodynamic model MIKE 21 in rain event no. 10 (see Supporting Information). Blue colors illustrate different water depths predicted by the two models. Cells where the NN failed to predict water depths >0.05 m are highlighted red, cells where the neural network falsely predicted water depths >0.05 m are highlighted yellow. The two figures on the left depict a scene in the city center, while the figures on the right show an area in the outskirts. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

event no. 10 for an area in the centre of Odense. Areas where the neural network failed to predicted water depths > 0.05 m are highlighted red, while areas where the neural network falsely predicted water depths > 0.05 m are highlighted yellow. Fig. 8 shows pixel-wise *NSE* values computed for the validation rain events for the same area as in Fig. 7, and for a more rural area. From the two figures, several characteristics of the neural network predictions are evident:

1. The neural network did not predict water depths that are entirely unreasonable. We did also not identify any such issues in other locations or for other rain events. This is important, because the mathematical formulation of the neural network does not account for physical constraints such as mass balances.
2. The neural network could identify hotspots of flooding where water accumulates, and it did also generate accurate predictions of water levels in these locations.
3. The network was not able to accurately represent shallow flooding along transport stretches, i.e. outside sinks. This behaviour is particularly evident from Fig. 8, where low *NSE* values cluster along roads and flow paths leading through backyards, and it explains the consistent underestimation of total flooded area by the neural network (Fig. 4). The five input datasets considered in this model were thus not sufficient to identify these locations.

To characterize the behaviour of the model in different rain events, we plotted the water volumes that the sinks (or terrain depressions) in our validation dataset contained based on the maximum water depths predicted by the neural network (NN) and the hydrodynamic model MIKE 21 (HD) (Fig. 9). We plotted blue spot volumes for all five rain events in the validation dataset, but only three of the events generated flooding in either of the models. This implies that the neural network was able to determine that no flooding should occur in events no. 44 and 47. This was the case even though these events had rather high maximum rain intensities of 34 and 16 mm/hr (averaged over a 30 min period).

However, while the predictions of water volumes could be considered accurate for event no. 9 and 10, we could also identify a clear underestimation of water volumes in event 39. This event had a maximum rain intensity of 42 mm/hr (averaged over a 30 min period). Several events with such intensities were included in the training dataset. However, event 39 was characterized by two pronounced rainfall peaks with similar intensity. It was the only event with this characteristic in our dataset and it is therefore reasonable to assume that the network simply did not learn to distinguish this type of rain event.

3.3. *Step 3 – k-fold cross validation*

Table 3 shows the score values computed for the water depth predictions generated by the neural networks in the 5-fold cross validation approach, and compares them against the score values obtained for the best performing model in step 2 of the model development procedure.

A clear increase in prediction error can be observed when comparing the score values obtained during model validation in the five folds against the best performing model from step 2. The same hyperparameters were considered in both steps and all models were trained on 10,000 patches. However, in step 3 these patches were sampled from a smaller number of rain events but a bigger part of the study area than in step 2. The increased error can thus be attributed to an insufficient representation of rainfall patterns in the training phase, which underlines the need to present the network with a representative sample of rain events in the training phase.

4. Discussion

4.1. *Model configuration and performance*

Our results show similar RMSE scores (0.08 m) for predicted flood water depths as existing screening approaches (Jamali et al., 2019) with lower *CSI* scores in the order of 0.5. It should be noted that our results were obtained for natural rain events and locations not considered in the training phase, unlike other studies which focused on artificial design storms (Jamali et al., 2019; Jamali et al., 2018) or known locations (Guo

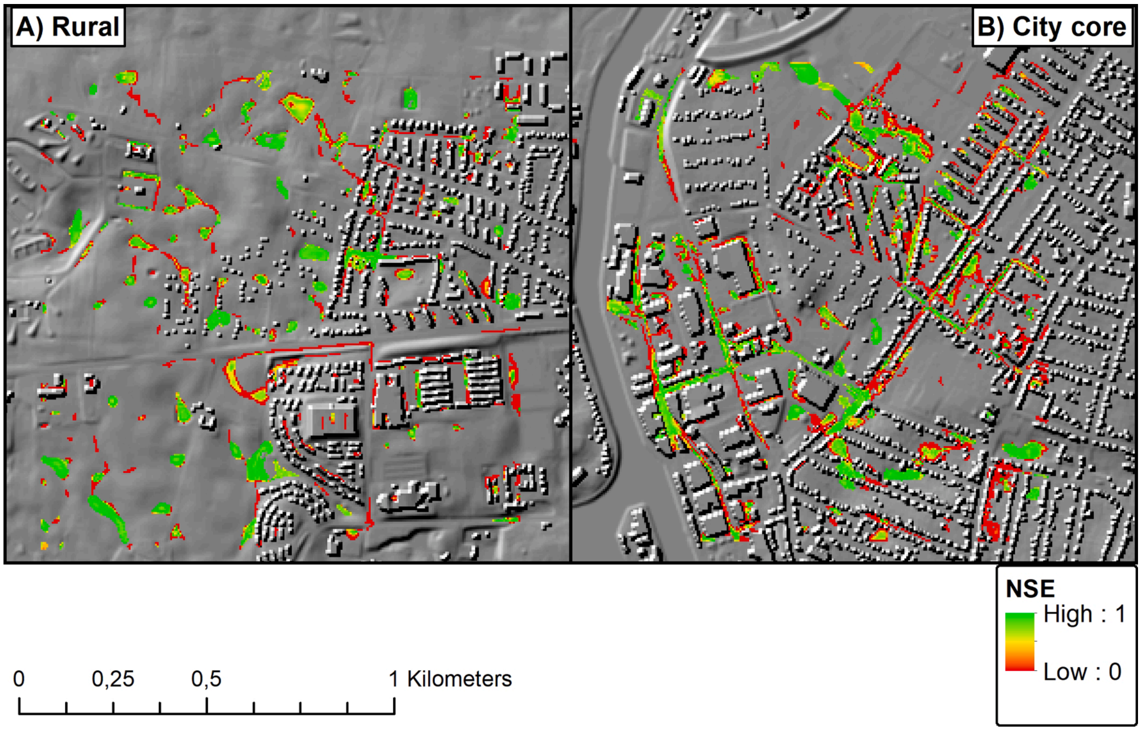


Fig. 8. NSE computed for predicted water depth in each pixel during 5 validation rain events. Panel A shows a rural-dominated area, while panel B shows the same urbanized area as Fig. 7. Only pixels where either U-FLOOD or MIKE 21 predicted water depths >5 cm in any of the 5 rain events were considered.

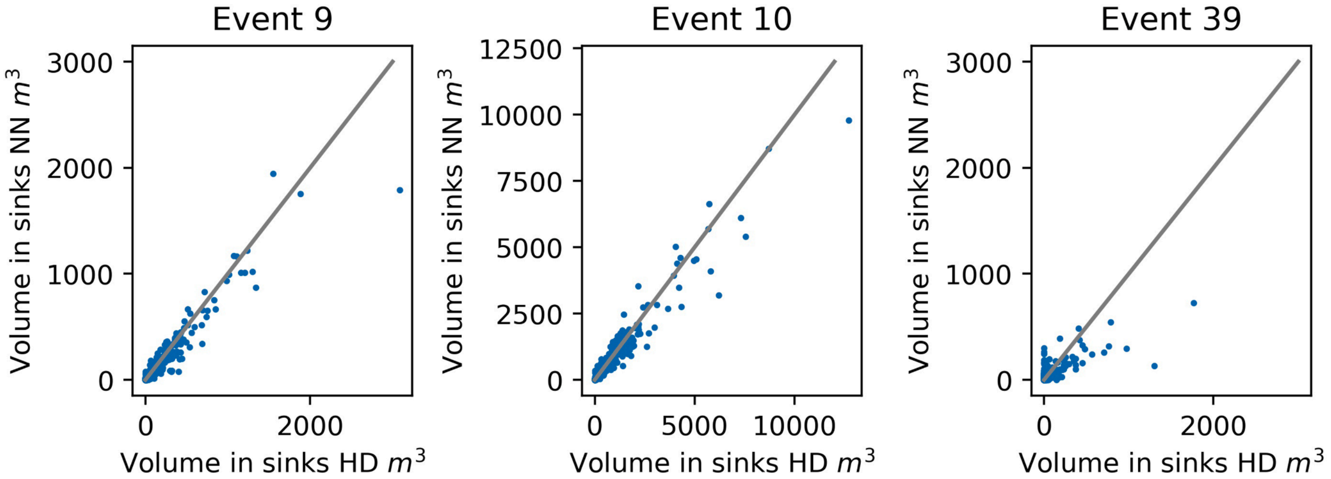


Fig. 9. Scatter plot of water volumes in sinks predicted by neural network (NN) and MIKE 21 (HD). Results are shown only for events 9, 10 and 39 (see Supporting Information), as neither model predicted flooding in events 44 and 47.

Table 3
Score values obtained in a 5-fold cross validation procedure as compared to the validation results obtained for the best performing model in step 2 of the model development procedure. The same hyper-parameters were used in steps 2 and 3, but different training and validation datasets were considered. A_{NN} and A_{HD} corresponded to the total flooded area with water depths >0.05 that were predicted by the neural network and the hydrodynamic model, respectively.

	RMSE	$CSI_{0.05m}$	$CSI_{0.3m}$	A_{NN}/A_{HD}
Step 2	0,080	0,583	0,592	71%
Fold 1	0,130	0,453	0,235	60%
Fold 2	0,121	0,447	0,438	50%
Fold 3	0,103	0,459	0,276	70%
Fold 4	0,111	0,451	0,272	86%
Fold 5	0,196	0,474	0,353	81%

et al., 2021). Water depths in depressions were generally predicted well, while the prediction of shallow flooding on surface flow paths outside depressions remains a challenge.

The forward selection procedure resulted in a model with five spatial input datasets. This procedure can account for interactions between variables and exploit the neural networks ability to perform spatial aggregation. It is therefore preferred over methods that perform one-variable-at-a-time comparisons between model inputs and the output variable such as ranked correlation or mutual information. It was also clear that considering too many model inputs increased the prediction error of the neural network due to overfitting in the training phase. Note that the forward selection procedure is not exhaustive. It adds variables one at a time and may therefore not discover all relevant combinations of variables. In addition, the selection of model inputs and hyper-parameters is entangled, because different models can exploit spatial information in different ways. More extensive procedures for selecting input variables and hyper-parameters will, however, pose computational challenges.

Our model was able to accurately predict water depths for many rain events, as well as to distinguish situations where no flooding should occur. However, the increased prediction error for the double-peak event no. 39 also illustrated a vulnerability in appropriately capturing the temporal dynamics of the rain events. The number of rain events that could be considered in the training dataset is limited, because time-consuming hydrodynamic simulations need to be performed for each rain event, and because the best performing network with $d = 5, k = 3$ and $bf = 64$ required a training time of up to 35 h (for 500 epochs) using a high-end GPU. Other studies applying data-driven models with a similar complexity have similarly not considered this issue (Berkhahn et al., 2019; Guo et al., 2021; Zahura et al., 2020), or focused on temporal resolutions of one day and above (Pham et al., 2020; Zhao et al., 2020). Different ways of specifying rainfall input to the neural network may also lead to higher prediction performance of the model. Based on previous experiences (Böhm, 2020; Eriksen and Dichmann, 2019), we experimented with neural networks that considered statistical rainfall characteristics based on average rain intensities for varying time intervals, as well as with 1D convolutional layers that extract temporal patterns from the rainfall time series. None of these so far yielded a better model performance than the setup outlined in Fig. 2 (Supporting Information S6).

4.2. Technical modelling setup

In line with the common approach for image segmentation and previous work of Guo et al. (2021), we have applied a modelling setup that generates predictions of water depth for a fixed-size patch of 256x256 pixels. This approach is counter-intuitive from a hydrological perspective, because hydrological features such as flow paths or sinks will be cut on the edge of the image. It is important to note that the model input inside an image is calculated based on the entire catchment before it is provided to the network. For example, flow accumulation values will not be affected by the cutting operation, because they are calculated on the entire terrain dataset, not on the 256x256 patch. In the prediction phase, the prediction window can be flexibly positioned in the desired location. In principle, the convolutional network enables the consideration of images of varying sizes in the training and the prediction phase. The image size could then be adjusted to the hydrological features in a specific location. However, challenges in setting up efficient input pipelines for the neural network may arise.

4.3. Limitations

U-FLOOD was trained based on hydrodynamic simulation results generated for a city in a moraine, coastal landscape. The hydrodynamic simulations considered a uniform distribution of rainfall in space, a simplified representation of the sewer system in the form of reducing the effective rainfall, and a constant, uniform infiltration rate for runoff computation on green areas. In addition, we excluded fluvial flooding from the training data. These factors limit the transferability of U-FLOOD in its current configuration as outlined below:

1. Different landscape types (e.g. alpine), different soil conditions - The consideration of varying infiltration rates requires an additional spatial input dataset that was not included in our work. Training the

model for different infiltration rates would also enable the consideration of antecedent rainfall which reduces the infiltration capacity of the soil (Davidsen et al., 2017a). Different landscape types might require revisiting the spatial input variables to the model. For example, terrain slope may well play a more prominent role in alpine regions than in our study.

2. Non-uniform distribution of rainfall in space – U-FLOOD uses simple rainfall statistics (or alternatively 1D hyetographs) as input to predict flood depth. This approach cannot be used when spatial variations of rainfall are relevant. Instead, setups that learn spatiotemporal rainfall dynamics would likely be required (e.g. Zhu et al. (2020)) and training efforts would increase substantially. Note that U-FLOOD was trained to predict pluvial flooding in catchments with extents in the order of 2 km. At such scales, spatially uniform rainfall is commonly assumed for flood screening in infrastructure design and weather models (as input to flood warning systems) hardly generate forecasts with resolutions finer than some km.
3. Non-uniform sewer system capacity – We assumed that the sewer system has a capacity of 12 mm/h in all locations in the catchment. This assumption will not be valid in many situations. A more dynamic consideration of network capacity could be achieved by training U-FLOOD to consider surcharge volumes from a 1D sewer model as input, similar to (Jamali et al., 2019; Jamali et al., 2018). This could be implemented by considering a spatial input dataset where flow paths are weighted according to the surcharge volume encountered along the flow path (similar to FLSLO), and it would also allow for the consideration of spatial rainfall variation in the 1D sewer model.

4.4. Outlook

As outlined in the previous section, future research should focus on generating increased training datasets that make U-FLOOD applicable to a wider range of situations. Technical improvements to consider are, in particular, the development of spatial input data that yield information on shallow water depths on flow paths (e.g. elevation above the nearest flow path, or potential water depth based on rough Manning calculations as demonstrated by Thrysøe et al. (2021)), and the implementation of stratified sampling schemes that consider the temporal dynamics of rain events. Finer model resolutions than 5 m are relevant for many practical situations and could be achieved with multi-scale neural networks (Nah et al., 2017) that process different input datasets at different resolutions. Finally, deep learning offers the possibility to consider multiple reference datasets when training. We can therefore envision models that learn from both hydrodynamic simulations and sparse flood observations, possibly enabling the calibration of flood models across catchments.

5. Conclusions

We have presented a setup for predicting urban pluvial flood hazards in high resolution and on short time scales using convolutional neural networks that are inspired by the widely applied U-NET architecture. Based on our results, we draw the following conclusions:

1. Neural networks that consider topographic inputs can be used for predicting pluvial flood hazards, also in spatial locations and for rain events that were not included in the training dataset, because they learn to associate terrain properties with the likelihood of flood occurrence.
2. The spatial input data provided to the neural network need to be selected in a setting that is similar to the final model used for prediction. This ensures that interactions between different input datasets and the ability of the neural network to perform spatial aggregation are considered when selecting input data.
3. Neural networks with topographic input data need to be designed in a parsimonious manner. Too many input datasets lead to overfitting and increased prediction error. In our study, a combination of five datasets describing terrain aspect and curvature, the depth of terrain depressions, imperviousness and flow accumulation yielded the best performing model.
4. Deeper networks improve prediction accuracy only up to a certain level. In our study, this level was reached with a model that had 28 million trainable parameters. Once trained, this network generated predictions of maximum water depth for an area of 1280x1280 m in less than one second.
5. The rain events presented to the neural network during training need to reflect the range of rain events for which the model should generate predictions. This applies not only for event depth and intensities, but also the temporal evolution of events. The computational demands for training deep neural networks limit the number of rain events than can be considered, thus creating a demand for stratified sampling schemes that consider the temporal evolution of events.

Our study contributes to a so far limited literature, where only two studies have applied similar approaches for predicting 2D urban flood maps in high resolution and on short time scales, and no evaluations of a large set of potential model inputs and of predictive performance on historical rainfall data are available. Interesting opportunities arise, for example, by considering our setup to train neural networks for flood prediction based on a simultaneous consideration of simulation results and flood observations from different cities, or by using U-FLOOD to minimize the number of hydrodynamic simulations required to assess expected flood damages.

CRediT authorship contribution statement

Roland Löwe: Conceptualization, Data curation, Investigation, Methodology, Software, Writing - original draft. **Julian Böhm:** Conceptualization, Investigation, Methodology, Writing - review & editing. **David Getreuer Jensen:** Conceptualization, Methodology, Writing - review & editing. **Jorge Leandro:** Conceptualization, Methodology, Writing - review & editing. **Søren Højmark Rasmussen:** Conceptualization, Methodology, Writing - review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

We thank the Danish Agency for Data Supply and Efficiency (SDFE), Danish Municipalities, and the Danish Meteorological Institute (DMI) for the provision of open data that formed the basis for this research. This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Appendix A. Supplementary data

Supplementary data to this article can be found online at https://doi.org/10.1016/j.jhydrol.2021.126898.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G.S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jozefowicz, R., Jia, Y., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Schuster, M., Monga, R., Moore, S., Murray, D., Olah, C., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., Zheng, X., 2016. Tensorflow: A system for large-scale machine learning, in: 12th Symposium on Operating Systems Design and Implementation 2016. pp. 265–283.

Agency for Data Supply and Efficiency, 2020. DHM/Nedbør (0.4m grid), Orto Forår [WWW Document]. URL download.kortforsyningen.dk (accessed 9.30.20). Usage conditions: https://download.kortforsyningen.dk/content/vilkår-og-betingelser.

Agency for Data Supply and Efficiency and Danish Municipalities, 2020. GeoDanmark [WWW Document]. URL download.kortforsyningen.dk (accessed 9.30.20). Usage conditions: https://www.geodanmark.dk/brugeradgang/vilkaar-for-data-anvendelse/.

Amidi, A., Amidi, S., 2019. Convolutional Neural Networks cheatsheet [WWW Document]. URL https://stanford.edu/~shervine/teaching/cs-230/cheatsheet-convolutional-neural-networks (accessed 2.25.21).

Avand, M., Moradi, H., lasboyee, M.R., 2020. Using machine learning models, remote sensing, and GIS to investigate the effects of changing climates and land uses on flood probability. J. Hydrol. 595, 125663. https://doi.org/10.1016/j.jhydrol.2020.125663.

Bach, P.M., Kuller, M., McCarthy, D.T., Deletic, A., 2020. A spatial planning-support system for generating decentralised urban stormwater management schemes. Sci. Total Environ. 726, 138282. https://doi.org/10.1016/j.scitotenv.2020.138282.

Badrinarayanan, V., Kendall, A., Cipolla, R., Member, S., 2017. SegNet : a deep convolutional encoder-decoder architecture for image segmentation. IEEE Trans. Pattern Anal. Mach. Intell. 39, 2481–2495. https://doi.org/10.1109/TPAMI.2016.2644615.

Balstrøm, T., Crawford, D., 2018. Arc-Malstrøm: A 1D hydrologic screening method for stormwater assessments based on geometric networks. Comput. Geosci. 116, 64–73. https://doi.org/10.1016/j.cageo.2018.04.010.

Bennett, N.D., Croke, B.F.W., Guariso, G., Guillaume, J.H.A., Hamilton, S.H., Jakeman, A.J., Marsili-Libelli, S., Newham, L.T.H., Norton, J.P., Perrin, C., Pierce, S. A., Robson, B., Seppelt, R., Voinov, A.A., Fath, B.D., Andreassian, V., 2013. Characterising performance of environmental models. Environ. Model. Softw. 40, 1–20. https://doi.org/10.1016/j.envsoft.2012.09.011.

Berkhahn, S., Fuchs, L., Neuweiler, I., 2019. An ensemble neural network model for real-time prediction of urban floods. J. Hydrol. 575, 743–754. https://doi.org/10.1016/j.jhydrol.2019.05.066.

Bermúdez, M., Ntegeka, V., Wolfs, V., Willems, P., 2018. Development and comparison of two fast surrogate models for urban pluvial flood simulations. Water Resour. Manag. 32 (8), 2801–2815. https://doi.org/10.1007/s11269-018-1959-8.

Beven, K.J., Kirkby, M.J., 1979. A physically based, variable contributing area model of basin hydrology. Hydrol. Sci. Bull. 24, 43–69. https://doi.org/10.1080/02626667909491834.

Böhm, J., 2020. Real-time forecasting of flood inundation maps using artificial neural networks. Technical University of Denmark. https://findit.dtu.dk/en/catalog/2598811327.

Brockhoff, P.B., Møller, J.K., Andersen, E.W., Bacher, P., Christiansen, L.E., 2018. Introduction to Statistics at DTU. DTU Compute, Kgs. Lyngby, Denmark.

Chattopadhyay, A., Hassanzadeh, P., Pasha, S., 2020. Predicting clustered weather patterns: a test case for applications of convolutional neural networks to spatio-temporal climate data. Sci. Rep. 10, 1317. https://doi.org/10.1038/s41598-020-57897-9.

Chen, L.-C., Zhu, Y., Papandreou, G., Schroff, F., Adam, H., 2018. Encoder-decoder with atrous separable convolution for semantic image segmentation. In: Ferrari, V., Hebert, M., Sminchisescu, C., Weiss, Y. (Eds.), Computer Vision – ECCV 2018. Springer International Publishing, Cham, pp. 833–851.

Davidsen, S., Löwe, R., Ravn, N.H., Jensen, L.N., Arnbjerg-Nielsen, K., 2017a. Initial conditions of urban permeable surfaces in rainfall-runoff models using Horton's infiltration. Water Sci. Technol. 77, 662–669. https://doi.org/10.2166/wst.2017.580.

Davidsen, S., Löwe, R., Thrysøe, C., Arnbjerg-Nielsen, K., 2017b. Simplification of one-dimensional hydraulic networks by automated processes evaluated on 1D/2D deterministic flood models. J. Hydroinformatics 19, 686–699. https://doi.org/10.2166/hydro.2017.152.

Deltares, 2017. SOBEK Suite.

DHI, 2016. MIKE 21 Flow Model & MIKE 21 Flood Screening Tool - Hydrodynamic Module - Scientific Documentation. Hørsholm, Denmark.

Dodge, Y., 2008. The concise encyclopedia of statistics: with 247 tables. Springer, Concise Encyclopedia of Statistics.

Eriksen, J.M., Dichmann, L.E.N., 2019. Varslingssystem imod regnbetingede oversvømmelser. Aalborg University. https://projekter.aau.dk/projekter/files/306662030/Afgangsprojekt.pdf.

GDAL Development Team, 2020. GDAL – Geospatial Data Abstraction Library, Version 3.2.0.

Guidolin, M., Chen, A.S., Ghimire, B., Keedwell, E.C., Djordjevic, S., Savić, D., 2016. A weighted cellular automata 2D inundation model for rapid flood analysis. Environ. Model. Softw. 84, 378–394. https://doi.org/10.1016/j.envsoft.2016.07.008.

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

He, K., Zhang, X., Ren, S., Sun, J., 2016. Deep residual learning for image recognition. Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. 770–778 https://doi.org/10.1109/CVPR.2016.90.

Hofmann, J., Schüttrumpf, H., 2019. Risk-based early warning system for pluvial flash floods: approaches and foundations. Geosciences 9 (3), 127. https://doi.org/10.3390/geosciences9030127.

Höhlein, K., Kern, M., Hewson, T., Westermann, R., 2020. A comparative study of convolutional neural network models for wind field downscaling. Meteorol. Appl. 27 (6) https://doi.org/10.1002/met.v27.610.1002/met.1961.

Innovyze, 2020. Infoworks ICM.

Isola, P., Zhu, J., Zhou, T., Efros, A.A., 2017. Image-to-Image Translation with Conditional Adversarial Networks, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Honolulu, Hawaii, pp. 1125–1134.

Jamali, B., Bach, P.M., Cunningham, L., Deletic, A., 2019. A cellular automata fast flood evaluation (CA-ffé) model. Water Resour. Res. 55, 4936–4953. https://doi.org/10.1029/2018WR023679.

Jamali, B., Löwe, R., Bach, P.M., Urich, C., Arnbjerg-Nielsen, K., Deletic, A., 2018. A rapid urban flood inundation and damage assessment model. J. Hydrol. 564, 1085–1098. https://doi.org/10.1016/j.jhydrol.2018.07.064.

Jean, M.È., Duchesne, S., Pelletier, G., Pleau, M., 2018. Selection of rainfall information as input data for the design of combined sewer overflow solutions. J. Hydrol. 565, 559–569. https://doi.org/10.1016/j.jhydrol.2018.08.064.

Kabir, S., Patidar, S., Xia, X., Liang, Q., Neal, J., Pender, G., 2020. 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.

Kaspersen, P.S., Høegh Ravn, N., Arnbjerg-Nielsen, K., Madsen, H., Drews, M., 2017. Comparison of the impacts of urban development and climate change on exposing European cities to pluvial flooding. Hydrol. Earth Syst. Sci. 21 (8), 4131–4147. https://doi.org/10.5194/hess-21-4131-2017.

Kratzert, F., Klotz, D., Herrnegger, M., Sampson, A.K., Hochreiter, S., Nearing, G.S., 2019. Toward improved predictions in ungauged basins: exploiting the power of machine learning. Water Resour. Res. 55 (12), 11344–11354. https://doi.org/10.1029/2019WR026065.

Li, H., Xu, Z., Taylor, G., Studer, C., Goldstein, T., 2018. Visualizing the loss landscape of neural nets, in: Advances in Neural Information Processing Systems (NIPS 2018).

Li, X., Willems, P., 2020. A hybrid model for fast and probabilistic urban pluvial flood prediction. Water Resour. Res. 56, 1–26. https://doi.org/10.1029/2019WR025128.

Lin, Q., Leandro, J., Wu, W., Bhola, P., Disse, M., 2020. Prediction of maximum flood inundation extents with resilient backpropagation neural network: case study of Kulmbach. Front. Earth Sci. 8, 1–8. https://doi.org/10.3389/feart.2020.00332.

Löwe, R., 2021. U-FLOOD - computer code associated with the article "U-FLOOD – topographic deep learning for predicting urban pluvial flood water depth." Technical University of Denmark. URL https://doi.org/10.11583/DTU.14206838.v1.

Löwe, R., Arnbjerg-Nielsen, K., 2020. Urban pluvial flood risk assessment – data resolution and spatial scale when developing screening approaches on the microscale. Nat. Hazards Earth Syst. Sci. 20 (4), 981–997. https://doi.org/10.5194/nhess-20-981-2020.

Löwe, R., Mair, M., Pedersen, A.N., Kleidorfer, M., Rauch, W., Arnbjerg-Nielsen, K., 2020. Impacts of urban development on urban water management – limits of predictability. Comput. Environ. Urban Syst. 84, 101546. https://doi.org/10.1016/j.compenvurbsys.2020.101546.

Löwe, R., Urich, C., Sto. Domingo, N., Mark, O., Deletic, A., Arnbjerg-Nielsen, K., 2017. Assessment of urban pluvial flood risk and efficiency of adaptation options through simulations – a new generation of urban planning tools. J. Hydrol. 550, 355–367. https://doi.org/10.1016/j.jhydrol.2017.05.009.

Maas, A.L., Hannun, A.Y., Ng, A.Y., 2013. Rectifier nonlinearities improve neural network acoustic models. ICML Work. Deep Learn. Audio, Speech Lang. Process. 28.

Madsen, H., 2008. Time series analysis, Chapman & Hall/CRC texts in statistical science series. Chapman and Hall/CRC, Boca Raton, FL, United States.

Meneses, E.J., Löwe, R., Brødbæk, D., Courdent, V., Petersen, S.O., 2015. SURFF – Operational Flood Warnings for Cities Based on Hydraulic 1D-2D Simulations and NWP, in: Proceedings of the 10th International Conference on Urban Drainage Modelling (UDM). Québec, Canada.

Müller, T., Schütze, M., Bárdossy, A., 2017. Temporal asymmetry in precipitation time series and its influence on flow simulations in combined sewer systems. Adv. Water Resour. 107, 56–64. https://doi.org/10.1016/j.advwatres.2017.06.010.

Nah, S., Hyun Kim, T., Mu Lee, K., 2017. Deep Multi-Scale Convolutional Neural Network for Dynamic Scene Deblurring, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).

Pardoe, I., Simon, L., Young, D., 2020. STAT 501 Regression Methods [WWW Document]. URL https://online.stat.psu.edu/stat501/lesson/welcome-stat-501 (accessed 1.20.21).

Pham, B.T., Luu, C., Phong, T.V., Trinh, P.T., Shirzadi, A., Renoud, S., Asadi, S., Le, H.V., von Meding, J., Clague, J.J., 2020. 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.

Rawat, W., Wang, Z., 2017. Deep convolutional neural networks for image classification: a comprehensive review. Neural Comput. 29 (9), 2352–2449. https://doi.org/10.1162/neco_a_00990.

Reichstein, M., Camps-Valls, G., Stevens, B., Jung, M., Denzler, J., Carvalhais, N., Prabhat, 2019. Deep learning and process understanding for data-driven Earth system science. Nature 566, 195–204. https://doi.org/10.1038/s41586-019-0912-1.

Ronneberger, O., Fischer, P., Brox, T., 2015. U-Net: convolutional networks for biomedical image segmentation. In: MICCAI 2015: Medical Image Computing and Computer-Assisted Intervention – MICCAI 2015. Lima, Peru, pp. 234–241.

SCALGO, 2020. SCALGO Live. https://scalgo.com/en-US/live-flood-risk.

Smith, L.N., 2017. Cyclical learning rates for training neural networks, in: Proceedings - 2017 IEEE Winter Conference on Applications of Computer Vision, WACV 2017. pp. 464–472. https://doi.org/10.1109/WACV.2017.58.

Thrysøe, C., Balstrøm, T., Borup, M., Löwe, R., Jamali, B., Arnbjerg-Nielsen, K., 2021. FloodStroem: a fast dynamic GIS-based urban flood and damage model. J. Hydrol. 600, 126521. https://doi.org/10.1016/j.jhydrol.2021.126521.

Voinov, A., Kolagani, N., McCall, M.K., Glynn, P.D., Kragt, M.E., Ostermann, F.O., Pierce, S.A., Ramu, P., 2016. Modelling with stakeholders – next generation. Environ. Model. Softw. 77, 196–220. https://doi.org/10.1016/j.envsoft.2015.11.016.

Wartalska, K., Kaźmierczak, B., Nowakowska, M., Kotowski, A., 2020. Analysis of hyetographs for drainage system modeling. Water 12 (1), 149. https://doi.org/10.3390/w12010149.

Webber, J., 2019. Reliable and Resilient Surface Water Management through Rapid Scenario Screening. Ph.D. thesis. University of Exeter.

Webber, J.L., Fu, G., Butler, D., 2019. Comparing cost-effectiveness of surface water flood management interventions in a UK catchment. J. Flood Risk Manag. 12, 1–12. https://doi.org/10.1111/jfr3.12523.

Zahura, F.T., Goodall, J.L., Sadler, J.M., Shen, Y., Morsy, M.M., Behl, M., 2020. Training machine learning surrogate models from a high-fidelity physics-based model: Application for real-time street-scale flood prediction in an urban coastal community. Water Resour. Res. 56, e2019WR027038. https://doi.org/10.1029/2019WR027038.

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

Zhu, Q., Member, S., Chen, J., Shi, D., Zhu, L., 2020. Learning temporal and spatial correlations jointly: a unified framework for wind speed prediction. IEEE Trans. Sustain. Energy 11, 509–523. https://doi.org/10.1109/TSTE.2019.2897136.