Coupling stochastic rainfall images and data-driven flood emulation for fast urban flood mapping

Tabea Cache[1]

Abstract

Urban environments are becoming extremely prone to destructive floods due to city's expansions, the increase of impervious surfaces, dense infrastructures and intensification of rainfall events as a result of climate change. Various hydrological models have been developed and growing attention has been given to data-driven model such as convolution neural networks. While the former are not adapted for studies such as risk analysis due to the high computational time, the latter have not yet explored rainfall spatial variability. This study aims at exploring the added value of using spatial distributed images of rainfall events as multi-channel feature inputs to the deep convolutional network U-Net. The model's accuracy does not provide satisfactory results yet, but future improvements are suggested.

1. Introduction

The World Meteorological Organisation (2021) identified floods as the natural disaster causing the second highest economic losses and third highest number of reported deaths globally since 1970. Urban environments, with dense infrastructures and populations, are particularly vulnerable. The impact of urban floods might be exacerbated by socio-economical and climate changes (IPCC, 2022; UN-DESA, 2018). Cities affect the proportion of impervious surfaces which reduces rainfall infiltration and increases the frequency and magnitude of pluvial floods (USGS, 2003; Cutter et al., 2018). Moreover, short duration rainfall events are intensifying under climate change (Fowler et al., 2021). The foreseen urbanisation and the impact of climate change encourages to develop resilient cities (Rosenzweig, 2018). In that scope, rapid flood models are crucial to understand the uncertainty and risks linked with flood in cities. These can also be used as flood warning tools to provide high resolution information on the flood depth and extent are crucial tools (Hammond et al., 2019; Apel et al., 2022).

Urban pluvial flood models can be classified into three main categories: physically based, non-physically based and data driven. Various physically based models have been developed (Al-Suhili et al., 2019). However, these models rely on the availability of high-resolution data such as the storm drainage system and require long computational time to simulate large areas with high spatial resolution necessary to capture urban features (Leitao et al., 2009; Guerreiro et al., 2019). Non-physically based models can use parallel computing which reduces computational time. The main shortcoming of these models is that their accuracy depends on the spatial resolution and simulation time step (Guo et al., 2020). Additionally, small grid cell sizes lead to computational time and power that are too important for uncertainty analysis of urban pluvial floods (Guidolin et al., 2016; Al-Suhili et al., 2019). Lastly, data-driven models have been given growing attention to emulate physical processes. Machine learning algorithms such as Artificial Neural Networks (ANN), Support Vector Machines (SVM), Random Forest (RF), Recurrent Neural Networks (RNN) and Convolutional Neural Networks (CNN) have been used to model runoff (Aziz et al., 2014; Tehrany et al., 2015; Wang et al., 2015; Han et al., 2021; Guo et al., 2020; Li et al., 2022). Recently, Guo et al. (2020) have successfully developed a data-driven urban flood emulator using only 0.5% of the time of a physically based model and using hyetograph data.

Recent studies have shown that the variability in spatial distribution of rainfall can explain 26% of the flow variability in a hydrodynamic urban flood model (Peleg et al., 2017). In order to have a thorough estimation of the risks and uncertainties of floods in urban areas, it is therefore necessary to include variability in rainfall spatial distribution. This can be achieved by usind stochastic weather generators that are capable of simulating synthetic spatially distributed meteorological variables for present or future climate (Fatichi et al., 2011; Peleg et al., 2019). Multiple realisations generated by these models can be used for risk analysis in hydrological studies (Waheed et al., 2020; Peleg et al., 2015).

Ronneberger et al. (2015) have shown that their deep con-

[1]University of Lausanne, Lausanne, Switzerland. Correspondence to: Tabea Cache <tabea.cache@unil.ch>.

Final Report. Spring Semester 2022, Machine Learning for Environment and Earth Sciences.

volutional network U-Net enables precise localisation in image processing. In this study, we investigate how this architecture can be applied to emulate fast urban flood maps based on spatially distributed rainfall images simulated using a parametrisation of a stochastic weather generator. The precipitation data have a spatial and temporal resolution of respectively 15 m and 10 min. The machine learning model is trained using flood maps generated by a non-physically based hydrological model and corresponding to the peak water depths for each rainfall event. We discuss the advantages of coupling a stochastic weather generator parametrisation with a deep convolutional network. Last, we identify future research directions to support risk analysis in urban environments.

2. Material and Methods

2.1. Case study

This study explores the use of the deep convolutional network U-Net to emulate maximum water depth images in an urban environment. To do so, a flood map dataset consisting of hydrological simulation results for a design storm was used as input. Design storms are commonly used to assess flood impacts and design resilient infrastructures (Sun et al., 2011). The most intense short rain event recorded in Switzerland was chosen as the design storm for this study. Peleg et al. (NP) simulated multiple realisations of this storm with the STREAP model (Paschalis et al., 2013). The STREAP model (Space Time Realisation of Areal Precipitation) is a stochastic rainfall generator simulating synthetic rainfall events. The statistical properties of these events are derived from observations and reproduced for the simulated event. Weather generators like the STREAP model have been extensively used in hydrological studies for various purposes such as extending short-record time series, exploring climate natural variability, filling incomplete time series with missing data or analysing extreme rainfall (Paschalis et al., 2014; Fatichi et al., 2016; Schuol and Abbaspour, 2006; Peleg et al., 2017). Here, the rainfall generator is used to produce rainfall images that have the same temporal structure but varying spatial distribution. Data from a convective permitting model (CPM) were then used to modify the design storm using three methods: a spatial quantile mapping, a uniform quantile mapping and a rainfall-temperature relationship. The multiple realisations of the design storm time series were fed into the cellular automata CADDIES flood model (Guidolin et al., 2016). Only the maximum water depth simulated for the city of Zurich are used as input to our U-Net model here. Using results from hydrological models as input to train machine learning algorithm has been shown to be a valid framework (Guo et al., 2020). In total, 30 rainfall realisations with 25 images each of size 512x512 and their corresponding flood map of size 512x512 were used to train, validate and test the model. 10% of the dataset was used to perform the validation and 10% for the testing of the model. The allocation of data to training, validation or testing was performed randomly. The rainfall time series used as input have a 10 min temporal resolution and both rainfall and flood images have 15 m grid cell size.

2.2. Machine learning algorithm

U-Net was first presented to process biomedical images where the task requires localisation of the desired output (Ronneberger et al., 2015). The architecture presented in Figure 1 can be summarised to consist of a contraction and an expansion path. The contraction path arises from the encoder that is similar to standard convolution networks. The expansion path from the decoder along with the concatenation of features from the contracting path with those of the expansion path enables to build high-resolution maps. The encoder is a series of two convolution layers followed by a maximum pooling layer. The series is repeated four times. The decoder is a series of deconvolutions, a concatenation and two convolutions. Similarly to the encoder, the series is repeated four times. The kernel sizes are 3x3 for the convolution layers and 2x2 for the max pooling and deconvolution layers.

The architecture was implemented in Python using Tensorflow 2.8.0 (Abadi et al., 2015). All computations were performed using Graphical Processing Units (GPU) to allow for shorter running times by performing parallel computations. The spatially varying precipitation events were randomly labelled as training, validation and testing sets. The training set was used to optimise the model's parameters and the validation set enabled to optimise the models hyperparameters and detect possible overfitting of the model. The test set consisting of data that are not included in the training and validation set allows to investigate the generalisability of our model.

The original images size are 1500x1824. These were resized to 512x512 by coarsening the images with a scale of 3, zero padding in the latitudinal direction and cropping in the longitudinal direction. The precipitation images were preprocessed using min-max normalisation or standardisation depending on the hidden activation function used. Additionally, the float precision was reduced from 32 bits to 16 bits due to memory limitations. Finally, the Adam optimiser was chosen for its high convergence speed and quality (Géron, 2019). The maximum number of epochs was fixed to 100 and an Early Stopping callback was implemented to avoid overfitting. In fact, the latter regularises the deep convolution network by stopping training when the minimum validation error is reached.

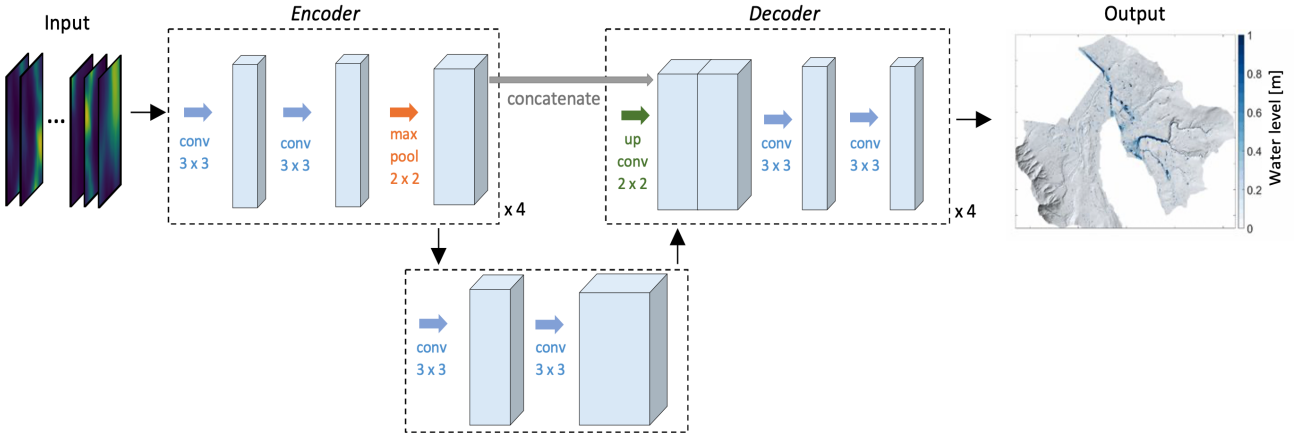

Figure 1. Architecture of the deep convolutional neural network U-Net. The arrows represent various operiations: blue for convolution, orange for maximum pooling, grey for concatenation and green for deconvolution.

2.3. Hyperparameters

The hyperparameter optimised during validation of the model are summarised in Table 1. The first hyperparameter characterises the filters which are the convolution kernels that slide across the input image of the layer. The value attributed to the filters defines the number of output filters in the convolution. The second hyperparameter is the activation function for the hidden layers. The ReLU and SELU functions in Table 1 stand respectively for Rectified Linear Unit function and Scaled Exponential Linear Unit function. The advantage of the SELU function is that it solves the vanishing or exploding gradients problem (Hochreiter et al., 1998). However, SELU does not outperform the default ReLU function in nonsequential architectures like the one of U-Net (Géron, 2019). In fact, self-normalisation that solves the vanishing or exploding gradients problem is not guaranteed due to the concatenation skip connection. Another activation function that was optimised is the output activation. Both output activation functions evaluated are activation functions for regression with positive outputs considering that the desired output is a flood map with only positive water depths. Finally, two loss functions were evaluated: the Root Mean Squared Error (RMSE) and the Mean Absolute Error (MAE). These are expressed as follow:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|(Y_i - \hat{Y}_i)^2|$$

where Y_i are the 'true' water depths computed by CADDIES and $\hat{Y}_i$ are the simulated values emulated by U-Net. The obtained performance metrics for training, validation and testing of the model are summarised in Table 2.

Table 1. U-Net Hyperparameter search range and selection.

HYPERPARAMETERS	SEARCH RANGE	SELECTED VALUE
FILTERS	8, 16, 32	16
HIDDEN ACTIVATION	RELU, SELU	RELU
OUTPUT ACTIVATION	RELU, SOFTPLUS	SOFTPLUS
LOSS FUNCTION	MSE, MAE	MSE

3. Results

Training the model was limited either by memory or computing power. The configuration presented here was limited by memory capacity while training the model with higher resolution images, using a scale factor smaller than 3, was limited by computing power. The model's performance is summarised in Table 2 and Figure 2 for two metrics and the different datasets. Training and testing of the model took respectively 0.88 minutes and 1 second.

The performance of the model is not satisfactory. While the performance metrics reported in Table 2 suggest that the error is very low, this is not representative as most of the simulated maps are not flooded. Non flooded grid cells have a value of zero and the flood depths are normalised which leads to performance metrics close to zero. The empirical cumulative distribution function in Figure 2 shows the failure of the U-Net model to satisfactorily emulate water depths. This failure holds true for low as well as for high water depths. This is rational considering the high number

Table 2. Performance metrics RMSE and MAE evaluated over the training, validation and testing datasets.

Performance Metrics	Training Set	Validation Set	Testing Set
RMSE	0.0576	0.0529	0.0505
MAE	0.0426	0.0424	0.0387

of trainable parameters and the relatively little training data. In fact, the model consists of 241,745 trainable parameters but is trained on only 27 precipitation events.

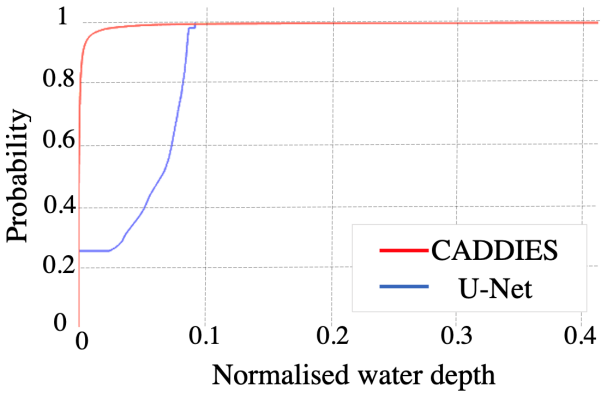

Figure 2. Empirical cumulative distribution function of the normalised water depths in Zürich for the design storm. The water depths are modelled by the hydrological model CADDIES and the deep convolutional network U-Net.

4. Discussion and Conclusion

At this stage, the proposed urban flood emulator does not enable to draw conclusions on added value of coupling spatially varying precipitation maps with deep convolution network. One of the main reasons is the limited training, validation and testing of the model with only 30 precipitation events that were upscaled by a factor of 3 due to memory and GPU limitations. Additionally, the model has been applied only to one city. The generalisation of the model would require to evaluate its performance for more urban areas. The next steps therefore involve:

- Training the model with all data available (120 realisations),

- Using data augmentation to further train the model, if necessary,

- Using transfer learning from a model trained for a similar problem.

Previous urban flood emulators have been trained using extracted information from DEM (Guo et al., 2020). Adding information to the multi-channel input such as the DEM and the slope could be investigated to increase the accuracy of the deep convolutional network. In that scope, the robustness of the model to changes in the city's organisation could also be evaluated. As the physical aspect of cities are changing due to expansion and greater density of infrastructures, an efficient model should be able to emulate flood maps with high accuracy despite modifications to the DEM.

Software and Data

The data used can be accessed under: Click here to access the data files.

The code used can be accessed under: Click here to access the GitHub code.

Acknowledgements

This study was completed with the advices and help from Tom Beucler and Milton Gomez Delgadillo.

References

Abadi, M., Barham, P., Chen, J., Chen, Z., Davis, A., Dean, J., ... Kudlur, M. (2016). Tensorflow: A system for large-scale machine learning. Paper presented at 12th symposium on operating systems design and implementation. USENIX, Savannah, pp. 265–283

Al-Suhili, R., Cullen, C., Khanbilvardi, R. (2019). An urban flash flood alert tool for megacities-Application for Manhattan, New York City, USA. Hydrology, 6(2). https://doi.org/10.3390/HYDROLOGY6020056

Apel, H., Vorogushyn, S., Merz, B. (n.d.). Brief communication-Impact Forecasting Could Substantially Improve the Emergency Management of Deadly Floods: Case Study July 2021 floods in Germany. https://doi.org/10.5194/nhess-2022-33

Aziz, K., Rahman, A., Fang, G., Shrestha, S. (2014). Application of artificial neural networks in regional flood frequency analysis: A case study for Australia. Stochastic Environmental Research and Risk Assessment, 28(3), 541–554. https://doi.org/10.1007/s00477-013-0771-5

Cutter, S.L, Emrich, C.T., Gall, M., Reeves, R. (2018). Flash flood risk and the paradox of urban development. Nat. Hazards Rev., 19(1), 05017005.

C40 (2021). How to reduce flood risk in your city. C40 Cities Climate Leadership Group, C40 Knowledge Hub.

IPCC (2022). Climate Change 2021 Working Group I contribution to the Sixth Assessment Report of the Intergovern-

mental Panel on Climate Change Summary for Policymakers. (n.d.).

Fatichi, S., Ivanov, V. Y., Caporali, E. (2011). Simulation of future climate scenarios with a weather generator. Advances in Water Resources, 34(4), 448–467. https://doi.org/10.1016/j.advwatres.2010.12.013

Fatichi, S., Ivanov, V. Y., Paschalis, A., Peleg, N., Molnar, P., Rimkus, S., Kim, J., Burlando, P. and Caporali, E. (2016). Uncertainty partition challenges the predictability of vital details of climate change, Earth's Future, 4(5), 240–251, doi:10.1002/2015EF000336.

Fowler, H. J., Wasko, C., Prein, A. F. (2021). Intensification of short-duration rainfall extremes and implications for flood risk: Current state of the art and future directions. In Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences (Vol. 379, Issue 2195). Royal Society Publishing. https://doi.org/10.1098/rsta.2019.0541

Geron, A. (n.d.). Hands-on machine learning with Scikit-Learn, Keras, and TensorFlow: concepts, tools, and techniques to build intelligent systems.

Guerreiro, S. B., Glenis, V., Dawson, R. J., Kilsby, C. (2017). Pluvial flooding in European cities-A continental approach to urban flood modelling. Water (Switzerland), 9(4). https://doi.org/10.3390/w9040296

Guidolin, M., Duncan, A., Gibson, M. (2012). CADDIES: a new framework for rapid development of parallel cellular automata algorithms for flood simulation EU-CIRCLE (A pan-European framework for strengthening critical infrastructure resilience) View project Saraswati View project. https://www.researchgate.net/publication/257199887

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. Journal of Flood Risk Management, 14(1). https://doi.org/10.1111/jfr3.12684

Hammond, M. J., Chen, A. S., Djordjević, S., Butler, D., Mark, O. (2015). Urban flood impact assessment: A state-of-the-art review. Urban Water Journal, 12(1), 14–29. https://doi.org/10.1080/1573062X.2013.857421

Han, H., Choi, C., Jung, J., Kim, H. S. (2021). Deep learning with long short term memory based sequence-to-sequence model for rainfall-runoff simulation. Water (Switzerland), 13(4). https://doi.org/10.3390/w13040437

Hochreiter, S. (1998). The vanishing gradient problem during learning recurrent neural nets and problem solutions. International Journal of Uncertainty, Fuzziness and Knowlege-Based Systems, 6(2), 107–116. https://doi.org/10.1142/S0218488598000094

Leitão, J. P., Boonya-aroonnet, S., Prodanović, D., and Maksimović, Č. (2009). The Influence of Digital Elevation Model Resolution on Overland Flow Networks for Modelling Urban Pluvial Flooding. Water Sci. Technol. 60 (12), 3137–3149. doi:10.2166/wst.2009.754

Li, P., Zhang, J., Krebs, P. (2022). Prediction of Flow Based on a CNN-LSTM Combined Deep Learning Approach. Water, 14(6), 993. https://doi.org/10.3390/w14060993

Paschalis, A., Molnar, P., Fatichi, S., Burlando, P. (2013). A stochastic model for high-resolution space-time precipitation simulation. Water Resources Research, 49(12), 8400–8417. https://doi.org/10.1002/2013WR014437

Peleg, N., Ban, N., Gibson, M. J., Chen, A. S., Paschalis, A., Burlando, P., Leitão, J. P. (n.d.). Graphical Abstract Mapping storm spatial profiles for flood impact assessments.

Peleg, N., Blumensaat, F., Molnar, P., Fatichi, S., Burlando, P. (2017). Partitioning the impacts of spatial and climatological rainfall variability in urban drainage modeling. Hydrology and Earth System Sciences, 21(3), 1559–1572. https://doi.org/10.5194/hess-21-1559-2017

Peleg, N., Molnar, P., Burlando, P., Fatichi, S. (2019). Exploring stochastic climate uncertainty in space and time using a gridded hourly weather generator. Journal of Hydrology, 571, 627–641. https://doi.org/10.1016/j.jhydrol.2019.02.010

Peleg, N., Shamir, E., Georgakakos, K. P., Morin, E. (2015). A framework for assessing hydrological regime sensitivity to climate change in a convective rainfall environment: A case study of two medium-sized eastern Mediterranean catchments, Israel. Hydrology and Earth System Sciences, 19(1), 567–581. https://doi.org/10.5194/hess-19-567-2015

Ronneberger, O., Fischer, P., Brox, T. (2015). U-Net: Convolutional Networks for Biomedical Image Segmentation. http://arxiv.org/abs/1505.04597 Rosenzweig, B. R., McPhillips, L., Chang, H., Cheng, C., Welty, C., Matsler, M., Iwaniec, D., Davidson, C. I. (2018). Pluvial flood risk and opportunities for resilience. In Wiley Interdisciplinary Reviews: Water (Vol. 5, Issue 6). John Wiley and Sons Inc. https://doi.org/10.1002/wat2.1302

Schuol, J., and Abbaspour, K. (2007). Using monthly weather statistics to generate daily data in a SWAT model application to West Africa, Ecol. Modell., 201(34), 301–311, doi:10.1016/j.ecolmodel.2006.09.028.

Tehrany, M. S., Pradhan, B., Mansor, S., Ahmad, N. (2015). Flood susceptibility assessment using GIS-based support vector machine model with different kernel types. Catena, 125, 91–101. https://doi.org/10.1016/j.catena.2014.10.017

UN-DESA: World Urbanization Prospects: The 2018 Revision, United Nations, Department of Economic and Social

Affairs, Population Division, New York, 22, 2018.

USGS (2003). Effect of Urban Development on Floods.

Waheed, S. Q., Grigg, N. S., Ramirez, J. A. (2020). Development of a parametric regional multivariate statistical weather generator for risk assessment studies in areas with limited data availability. Climate, 8(8). https://doi.org/10.3390/CLI8080093

Wang, Z., Lai, C., Chen, X., Yang, B., Zhao, S., Bai, X. (2015). Flood hazard risk assessment model based on random forest. Journal of Hydrology, 527, 1130–1141. https://doi.org/10.1016/j.jhydrol.2015.06.008

WMO (2021). Atlas of mortality and economic losses from weather, climate and water extremes (1970-2019).