River stage gauge by deep learning image recognition

Gerardo Villegas Contreras, Paola Naomi García Reyes, Renata Uribe Sánchez, and Ivan Emmanuel Gutiérrez Y López

Instituto Tecnológico y de Estudios Superiores de Monterrey, Campus Guadalajara

Abstract. The focus of this study is the image recognition with deep learning methods that combine feature extraction and convolutional neural networks in order to predict the river stage in different seasons of the year. The inherent advantage of having an algorithm capable of predicting the river stage lies in the circumstances where the gauging sensors could fail, so these gaps are filled with the imagery information. Therefore, an initial machine learning approach was implemented by applying a statistical analysis over the extracted features data set in an attempt to understand the yearly behavior of the stage measurements. In order to address this issue, two convolutional neural networks (CNN) were developed: a VGG16 and a second one accomplished by our research group. In both approaches, the images were previously filtered with a CNN classification algorithm to identify if the picture was suitable for the model prediction. As an additional study, there was also incorporated an hydrology expert chat-bot to get familiarized with some water terms relevant within the study and a user interface.

Keywords: Machine learning · Image processing · Deep Learning · CNN · Hydrology · River Stage · Resnet model · VGG model

1 Introduction

The era of accelerationism in which we live, leads to the inevitable scarce of resources, specifically water reserves, which can result in various negative scenarios for humanity such as famine, cold wars, economic crises and water shortages. As this persistent financial crisis continues to progress, the governments adopt an austerity policy, privatization of public services, an unemployment wave but also an unstoppable capitalism production and consumerism. [5]

Hence the importance of managing and monitoring water bodies that provide resources to populations, that, if applied correctly, could prevent both floods and future water crises. In addition to these important factors, the proper management of water bodies is capable of generating intelligent supply systems and thus avoiding water shortages, a phenomenon that is already a reality in various parts of the world.

That is the main reason on going deep into hydrology phenomena and get familiar with some water related terms such as river flows, floods and measure techniques to analyze the main characteristics of a river: its stage and discharge, the main subjects of this study. Consequently, one of the most significant interests to hydrologists is to gauge the amount of water within a stream because of what it represents: the pure end-product of all the previous processes in a hydrological cycle. [2]

However, it is key to start defining these two main concepts of hydrology which in fact are strongly related. Based on the stream elemental concept as a body of flowing water always in a natural channel, its primary characteristic, the stream discharge or stream flow is defined as the volume of water that runs through a specific location at a specific time, normally expressed in cubic meters per second. Even though the discharge could be measured continuously, it is rather calculated as an averaged flow over a time period on a particular river. On the other hand, the stream stage refers to the water level above some arbitrary point in a stream. [3]

As mentioned previously, these two concepts are highly related, and it is the usefulness of the so-called rating curve where multiple discharge gauges can lead to interlocking both characteristics. In addition, the inherent advantage of this rating curve is the allowance of continuous stream stage measurements that can lead to the future discovery of the real discharge. Thus, the stage-discharge relationship is obtained by making a series of velocity and area evaluations at a given location while recording the level with a stilling well. [2]

It is through deep learning algorithms and some concepts from an hydrology domain expertise that this study provides an approach focused on water level prediction based on image recognition and subsequent feature extraction of a stream.

The University of Nebraska has conducted an initial case study and it is our intention to continue its development(see *Camera-based Water Stage and Discharge Prediction with Machine Learning. Hydrology and Earth System Sciences Discussions.* [1]) In the initial case study, the authors used the same imagery dataset from the North Platte River over a six-year period. Feature extraction was implemented in two different ways: by computational algorithms and manually by hydrology experts. Image features extracted by the algorithms include pixel intensity, image entropy, saturation, etc. In the other hand, the domain expertise (hydrology) features were hand-crafted recognized by and hydrologist: and they were streams characteristics such as shape and texture of white foam, and the color of the water above and below the weir.

2 Methodology

Before addressing solutions for predicting the river stage based on image recognition, a work plan was implemented following most of the important aspects of the CRISP-DM methodology as a standard process for data mining. This workflow is exemplified in Fig. 1 [4]. In addition to following up on previously extracted features (see *Camera-based Water Stage and Discharge Prediction with Machine Learning. Hydrology and Earth System Sciences Discussions* [1]), image processing was implemented from scratch to extract new features from the images in a different approach. Subsequently, deep learning models were implemented through convolutional networks after constructing a supervised classification algorithm to identify the most relevant images that are supposed to be usable, without fog and snow and removing the darker ones because of the dawn and dusk instants. For all image and data processing we used a *Colab Notebook* with an AWS instance called *AWS Deep Learning Notebook*. This will permit a faster analysis as there is a larger capacity in infrastructure.

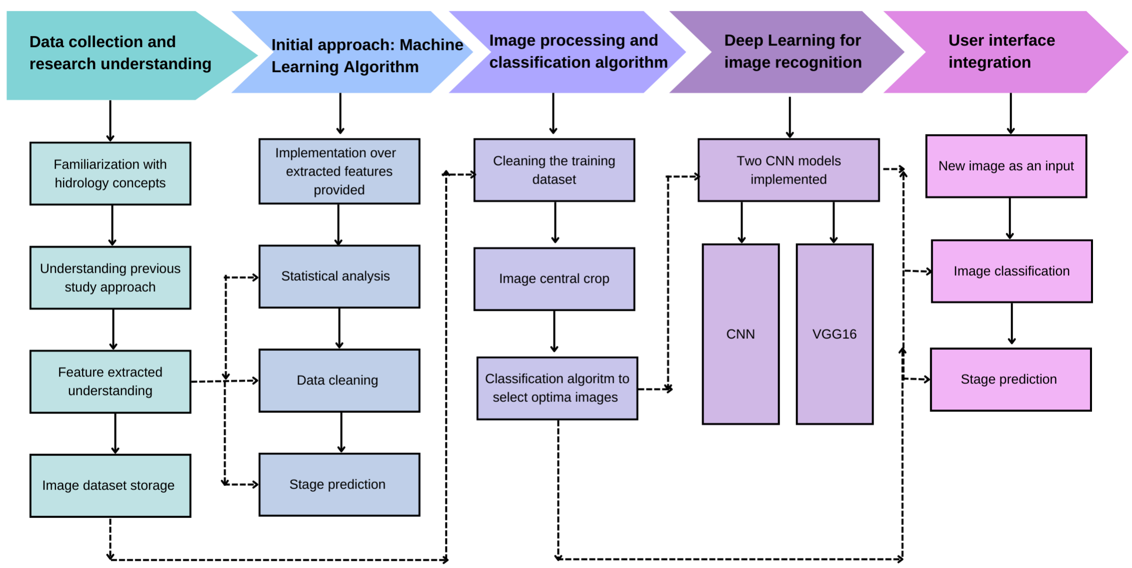

Fig. 1: Study Workflow

2.1 Data collection and research understanding

As mentioned above, the data set used in this research was gathered by the University of Nebraska as a previous study approach of the river discharge and stage. Thus, the primary study extracted features from the images, counting 55 variables including the dependent variables (stage and discharge) and predictors, such as exposure, gray scale features, width and height.For more detailed variable description see Appendix section, Table 7. Although there are also embraced some hand-crafted features created by the hydrologist experts like the measurement of the discharge-stage relationship as the rating curve, the shape and texture of the white water and the colour of the water above and below the weir. For a more detailed variable description see Appendix section, Table 7.

The whole raw imagery data was taken by a single angle corresponding to the North Platte River over a six year period (2012, 2013, 2016, 2017, 2018, 2019) through different seasons and hours, including unclear images (darker ones) from dusk/dawn moments and fog/snow (extremely white images). A sample of a suitable and clear imagery is shown in Fig. 2.

Fig. 2: North Platte River imagery sample

2.2 Initial Approach: Machine Learning Algorithm

In order to start a machine learning approach to predict both stage and discharge, given the extracted features data set, there was a statistical analysis performed initially to filter the variables with more relevance within the model. After computing a matrix correlation heat map it was confirmed the expected strong relationship between the river stage and discharge. The resulting correlation coefficient correspond to 0.97, therefore, the decision to focus only on the prediction of the river stage was made. Due to the stage-discharge correlation, all the analysis and models presented below focuses exclusively on the river stage.

As a machine learning model, an implementation with the whole data set was computed in order to identify the insignificant variables. After applying a standard scaler, the predictors that presented a p-value lower than the significance level of 0.05 and the empty columns were removed. At the end of this procedure there were 50 remaining predictors and there were deleted approximately 2000 outliers through an influence technique for each data set observation as well. As a standard statistic practice, there was applied a deeper data cleaning by identifying and subsequently removing the higher points on leverage, cook distance, betas and DFFIT. Since there was no significant cook distance, betas and DFFIT values, only outliers and higher leverage points were removed. At the same time, the preceding models implemented by the University of Nebraska were replicated to analyse their results for ourselves.

Within the statistical analysis, a calendar visualization was made in order to graphically represent the annual fluctuations of the stage in this particular stream, being its minimum representation 2.87 m during winter and the higher point 6.49 m corresponding to summer season. The calendar is shown in Fig. 3, each square represents a day where the rows are the respective years of the study. It is clearly visible an increase of the stage levels during summer, mainly represented by the months from may to september with minimum variations. This tool was usefully for understanding the seasonality, the missing data (see purple squares in Fig. 3) and perhaps the increasing global warming.

2.3 Image Preprocessing and classification algorithm

As there is interest in the exclusive use of images where enough information can be extracted in order to predict the desirable measurements, the entirety of the data samples can't be used as there are images where there's not enough light to distinguish any shape (Fig. 4a) or images that were too blurry due to weather conditions (dirt, snow, fog, rain, etc.) (Fig. 4b). Taking this into account, and given the vast amount of images, an efficient way to distinguish between useful and not useful images needed to be found. Due to the fact that the given feature extracted data set had already been cleaned by the initial study, it was concluded that a classification algorithm could be implemented to identify the optimal images, the clear ones, without blurry and dark pixels.

Besides, we explored an image pre-processing to focus mainly on the area of interest in the image, where the discharge is located, ignoring external elements such as trees and the sky. In order to accomplish this, we applied a central crop to the whole imagery data set (see a crop example in Fig. 5), a pixel reduction to 150 x 150 and a class balancing to equally train the model.

Therefore, we defined the training data set of the algorithm as the preceding extracted features observations, considering them all as optimal images and detecting those that had not been included in the cleanup to be considered unsuitable for the model. Some of the parameters considered in this section include binary cross entropy as the loss function, the *Adam* optimizer and a total of 10

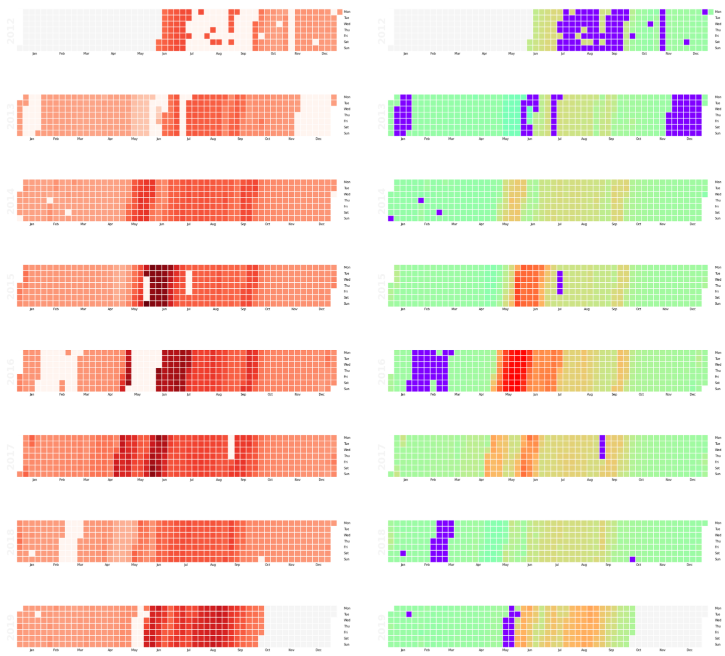

Fig. 3: Stage fluctuations calendar from 2012 to 2019 and calendar missing data

epochs for training. Our CNN (Convolutional Neural Network) implemented returns 1 if the image is suitable to predict stage and 0 if it is not. This algorithm methodology is explained in the Fig. 6

2.4 Deep Learning for image recognition

In this work, we choose three different CNN algorithms to be implemented: the Resnet, VGG16 and a third model that was structured by ourselves.

As a CNN that contains 16 convolutional layers, VGG16 is a suitable model for the current work involving image feature extraction by deep image analysis just like the Resnet for the deep feedback capability even with a large number of layers. [6]

Firstly, the three models were implemented without prior image processing where only 1140 images were used for training. Subsequently, an attempt was made to implement the three models but the Resnet failed due to computational power issues, however the other two models performed very well, perhaps due to the 26,918 images used for training.

In Fig. 7 and Fig.8 it is briefly presented as a diagram, the architecture of the CNN implemented by our research group and the VGG16, both models developed after image pre-processing. It is important to mention that the models

(a) Dark Image

(b) Blurry Image

Fig. 4: Images where useful information can't be extracted

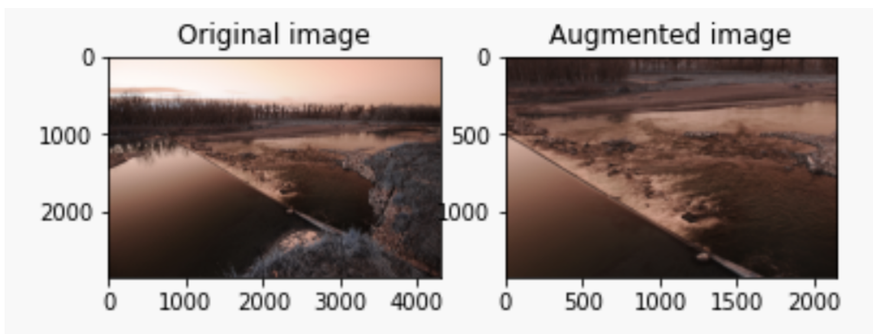

Fig. 5: Centralized crop explained

were trained with 30 epochs using the MSE as loss function and R^2 as the evaluation metric but an early stopping was applied to reduce resources when the model was no longer improving, so both algorithms stopped at epoch 20.

2.5 User interface integration

In an attempt to facilitate the use of our algorithm, i.e. generate stage rate predictions from new images, we conducted a user interface through the streamlit framework for ML and data science to easily create web apps open-source app framework for Machine Learning and Data Science teams. Within the web platform the user most upload an image that is previously filtered by the classification algorithm and then the stage is predicted. If there is an unsuitable picture for the model, a message is displayed with the option of introducing the desirable date if known. On the other hand, if the image is optimal, then it is introduced in the CNN for the stage prediction. On the screen there are three different stage levels: the real one, and the ones calculated with both models (VGG16 and our developed model). (This methodology is shown in Fig. 9)

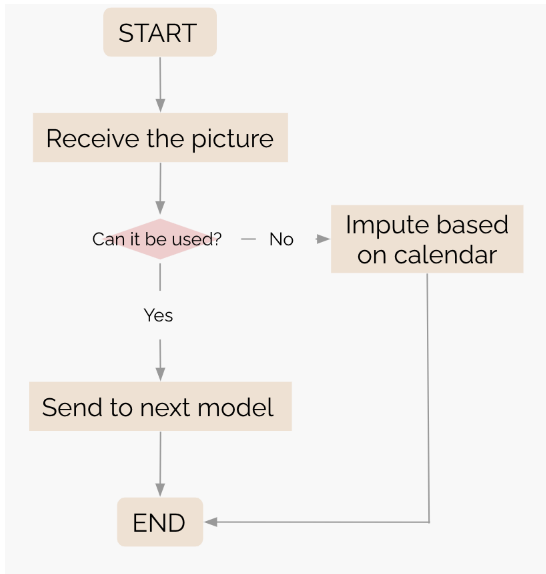

Fig. 6: Classification algorithm methodology

2.6 Water chatbot

Starting from a premise of hydrological content in text form (the excerpt was extracted from the book Fundamentals of Hydrology, 2008)[2], the following natural language processing methodology (see Fig. 10) has been followed including a first tokenization, text cleaning (removing punctuation and , frequency diagrams as word clouds and the subsequent implementation of a transformer: RoBERTA to give answers to possible questions raised in relation to the topic.

3 Results

3.1 Initial Approach Machine Learning Algorithms

As a further comparison experiment, we replicate the algorithms (with no data cleaning included) in the previous study of the University of Nebraska [1]to better understand the problem and try to improve the results. In Table 1 we show the corresponding coefficient of determination of the tree regression models to predict stage: MLP(Multi Layer Perceptron), SVM(Support Vector Machine) and RFR(Random Forest Regression). As can be seen, the MLP and RFR models show a very poor performance since the R^2 value is beneath zero. Regarding to the SVM model, it is shown a better performance reaching a 0.637 coefficient but it is not the optimal.

As mentioned earlier, the intuitively strong correlation between stage and discharge led us to predict exclusively a stage measurement. A preliminary model was computed with no data cleaning resulting in a 0.665 R^2.

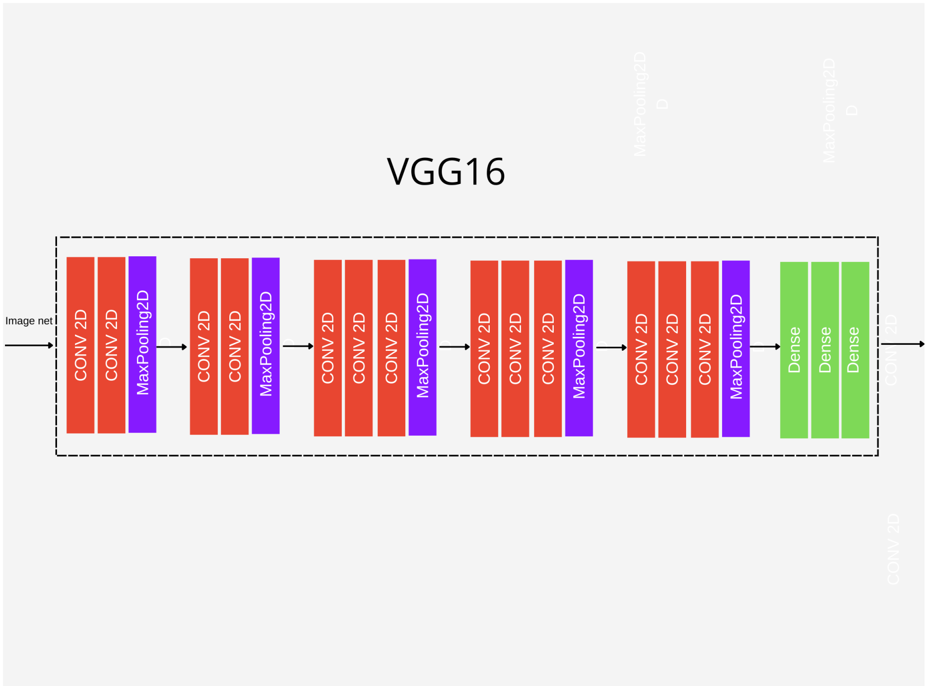

Fig. 7: VGG16 architecture

University of Nebraska stage predictio

Model	R^2
Multi Layer Perceptron	-1169.12
Support Vector Machine	0.637
Random Forest Regression	-0.0068

Table 1: University of Nebraska initial study R^2 stage prediction

After the data cleaning (removal of 2,000 records of outliers, columns filled with zeros and predictors statistically non significant) there were implemented two models: the simplest linear regression and a MLP. A high leverage points was attached to the data cleaning slightly improving the R^2 of the regression model from 78.4 to 78.6, as shown in Table 2

As can be shown in Tables 1 and 2, there was a notable improvement between the coefficient of determination of the initial algorithms in contrast to the ones we implemented due to data cleaning.

However, two different models could be considered for each season, since in summer there are enough higher flood records.

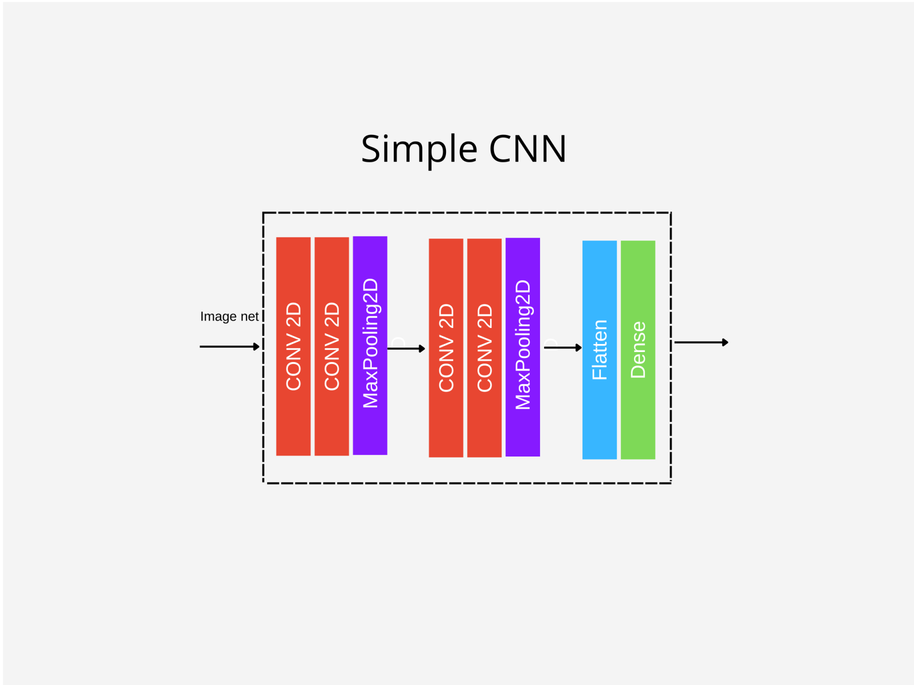

Fig. 8: Simple CNN architecture

Machine learning models	
Model	R^2
LR without outliers	0.784
MLP without outliers	0.918
LR without outliers and high leverage points	0.784
MLP without outliers and high leverage points	0.928

Table 2: Machine Learning R^2 results after data cleaning

3.2 Image classification algorithm

A deep learning convolutional neural network was selected as a filtering step to process images before entering in the stage prediction algorithm. The model is focused in recovering clear images, without extremely white and dark pixels. Due to the outstanding good performance of this classifier after only 10 epochs (see Table 3 and Fig. 11) nearly reaching a 100% on accuracy (98%) and an extremely low value of the loss function as binary cross entropy (0.004) we did not hesitate on implementing it over the main prediction algorithm.

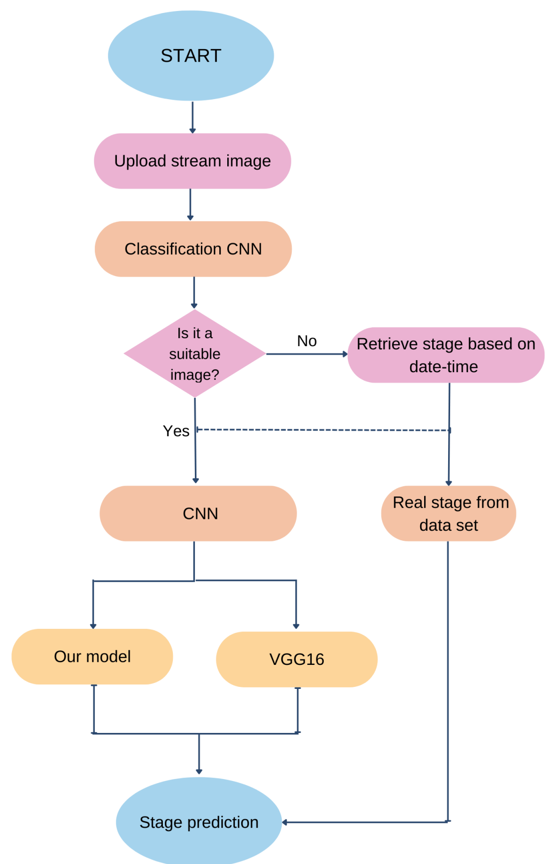

Fig. 9: User interface flow

CNN classifier results	
Accuracy	Loss Function value
0.98	0.004

Table 3: CNN classifier accuracy and loss function

As part of the results, the confusion matrix related to the classification model is also included (see Fig. 12), where the high performance of the algorithm can be analyzed.

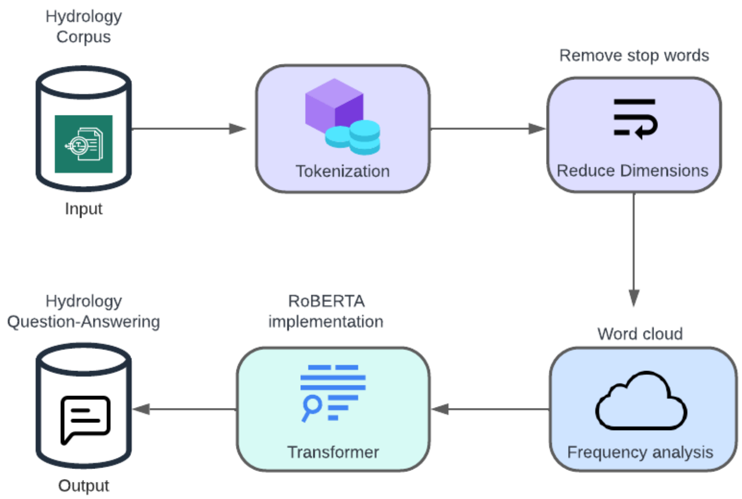

Fig. 10: Chatbot implementation methodology

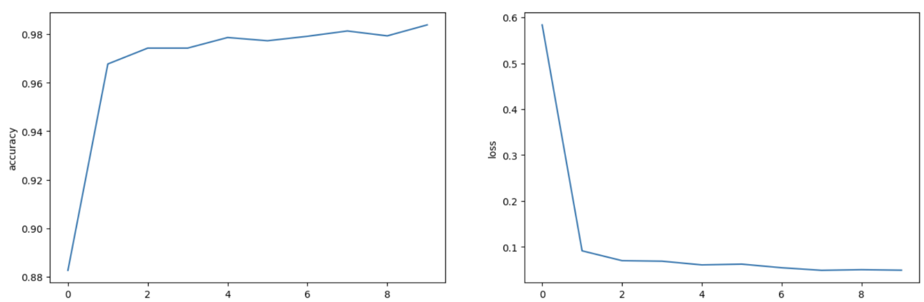

Fig. 11: Accuracy and loss function curves for classification model

3.3 Deep Learning and Convolutional Neural Networks

In a first attempt to implement the convolutional neural network, even without the processed images, we developed the following models: simple VGG16, Resnet and a third model created by our own layers. The results of this initial approach on CNN are shown on Table 4. As can be seen, the metrics are very poor, being the best the simple CNN model.

Due to the poor performance of the models, the images had to be pre-processed. Afterwards, the results of both CNNs can be seen on Table 5 in terms of R^2 and MSE, also including the learning curves of the loss function and the accuracy of both models on Fig. 13.

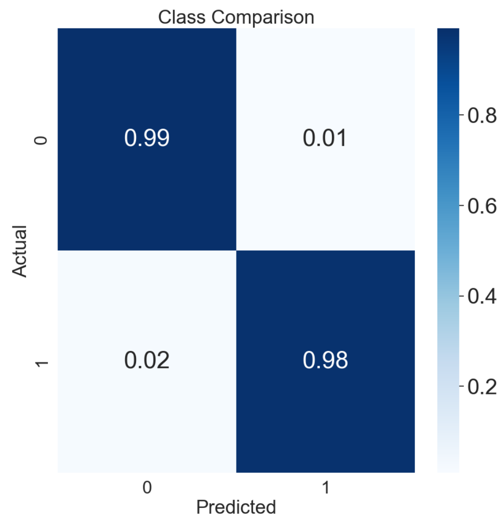

Fig. 12: Confusion matrix for classification model

CNN R^2 and MSE before image processing			
	Simple VGG16	Simple CNN	Resnet
R^2	0.370	0.456	0.325
MSE	0.5006	0.3167	0.4481

Table 4: CNN model results before image processing

CNN R^2 and MSE after image processing		
	Simple VGG16	Simple CNN
R^2	0.985	0.975
MSE	0.0079	0.0141

Table 5: CNN model results after image processing

Taking into account the incorporation of both algorithms (classification and stage prediction) the best achieved combination that reaches more than 0.98 for both algorithms is, of course, with image pre-processing where classification reaches 0.98 of accuracy and stage prediction 0.985 R^2 using the best convolutional neural network for this study: VGG16.

The Fig. 14 shows the contrast between the stage prediction of the VGG16 model and the real value, highlighting the good performance of the study.

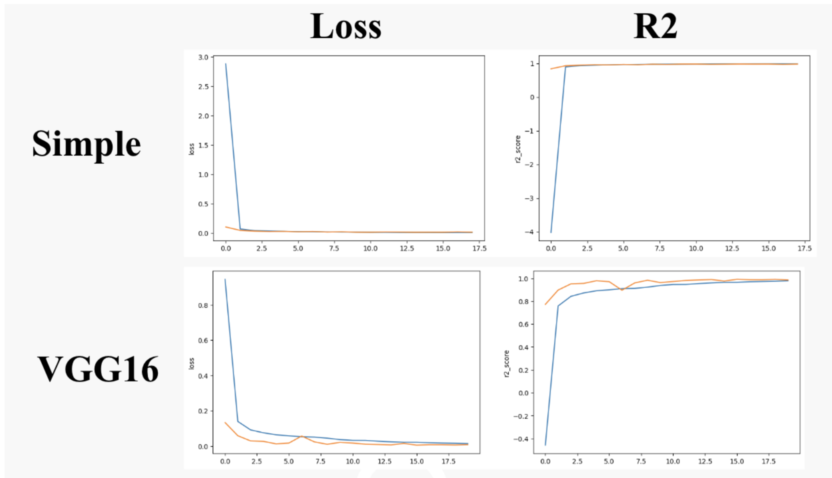

Fig. 13: CNN loss function and accuracy curves

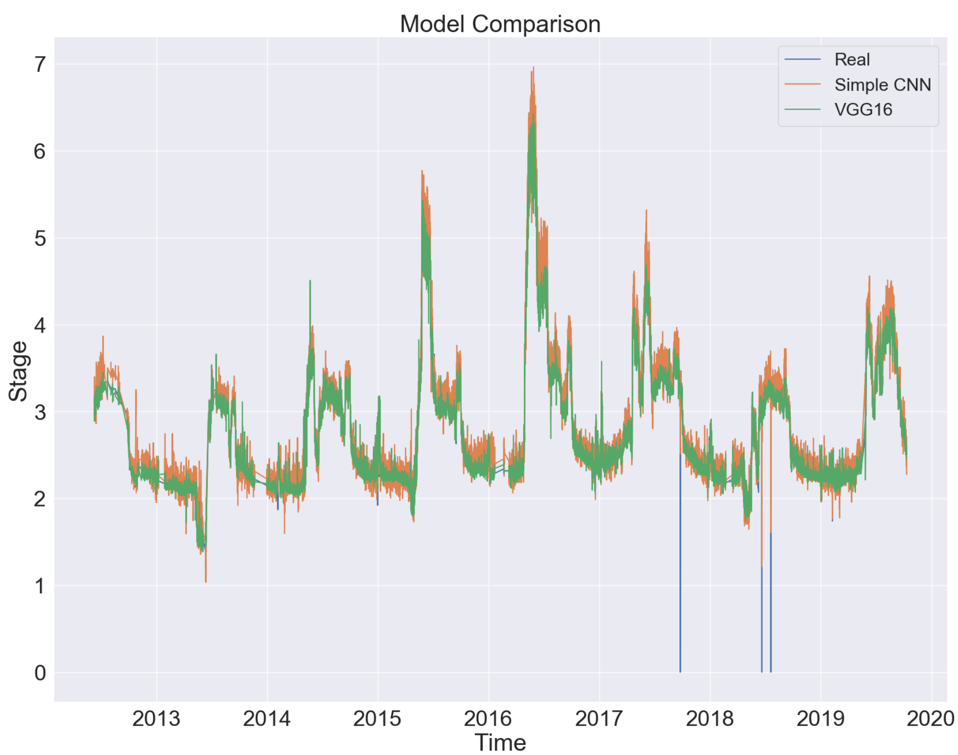

Fig. 14: Real stage vs Simple CNN and VGG16 prediction

4 Conclusions

In terms of the stage-discharge relationship, one of the most notable difficulties is that the frequent measurement leads to many records of low and medium weir, that is normally, throughout winter and other average weather periods unlike the few data recovery on potential summer floods. This scarce of stream overflow records is generally because of the infrequency of the phenomena and the danger of measuring the weir discharge during the flood.

Although, the model we developed can be used only on the specific river that it was trained for (North Platte River) the pipeline of this study can be replicated in other streams or water bodies. However, the filtering classification model could be used on other rivers, to show the image utility on a later CNN application to predict the stage. This work had shown a significant enhancement against the previous study, reaching suitable results of the coefficient of determination (R^2) and implementing additional resources as the hydrology chatbot and the user interface to facilitate the application of the CNN models.

5 Further steps: a chatbot hydrology expert and the user interface

5.1 Chatbot

As mentioned above, our models might be useful only for this specific stream, but a further research could implement pre-trained new models based on this work.

On the other side, as a complement to the study, a chatbot expert in hydrology was implemented, which is to be incorporated into the user interface to answer questions about hydrology when there are no experts to interpret results.

Before the corpus analysis a word cloud was computed (Fig. 15), derived from its frequency of occurrence in the text, the chatbot is now an expert in hydrologic terms such as discharge and stage.

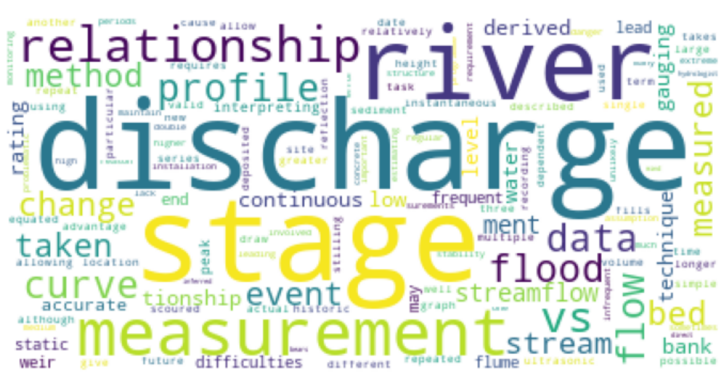

Fig. 15: Word cloud derived from the training corpus

After feeding the transformer with the text corpus mentioned above, it was asked a few questions to evaluate its performance. (See chatbot example questions in Fig.16 where the blue square represents the chatbot and the gray one an average user)

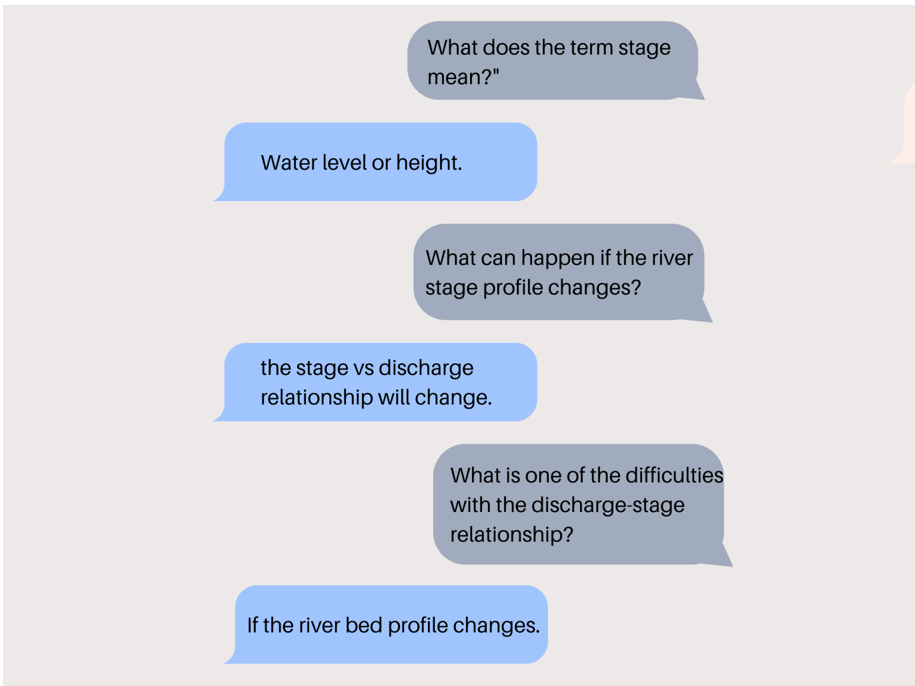

Fig. 16: Chatbot conversation example

Although it is true that the transformer did not always get it right, his answers, even if there were redundant and brief, managed to make sense within the context. Through this simple implementation it is feasible to identify the different possibilities and applications that this model could mean for the familiarization of hydrological concepts within and outside the purposes of this study.

5.2 User interface

This section explains in detail, with screenshots, how the interface works.

After browsing a river image from the local files (see Fig. 17), if it is appropriate to calculate the stage, the following appears on display (Fig. 18) showing three different numbers: the real stage, and the predicted with our model and the VGG16.

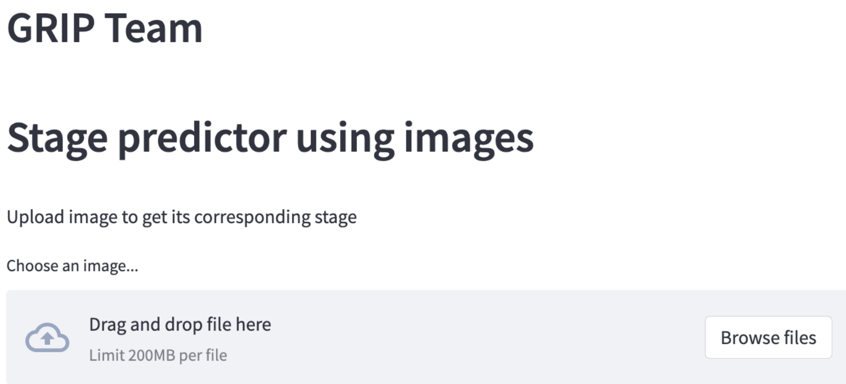

Fig. 17: Initial browse from interface (input)

If the selected image is dark or blurry, then the following message is displayed with the possibility, if known, of selecting the desirable stage measurement date-time (shown on Fig. 19). After selecting this information, the exact, or the closest date-time stage measurement is returned as in Fig. 20.

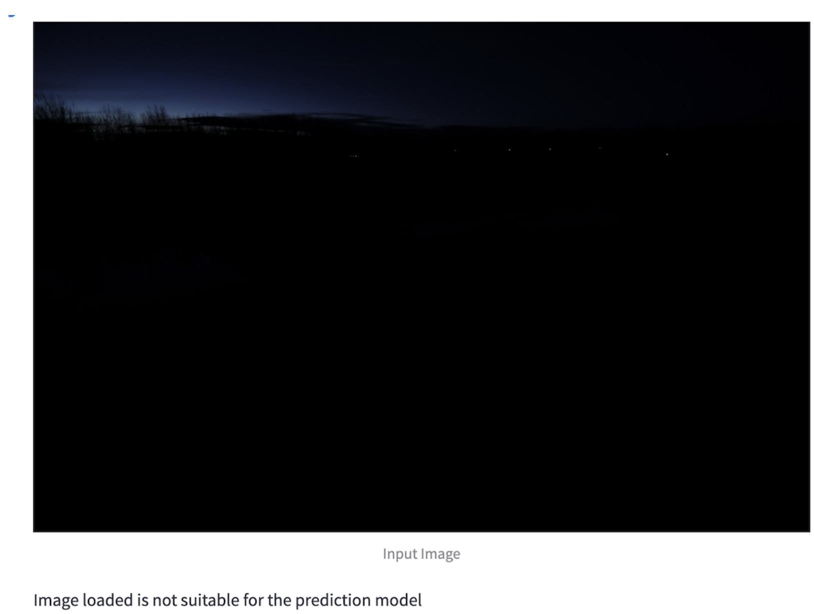

Fig. 19: Dark image failing example

Sample Image

Predicted Stage:

Real: `3.0`

Our model: `3.02`

VGG: `3.08`

Fig. 18: Optimal image selection and stage prediction

Please select the date of the Stage to load

2016/05/25

Select the time for that date

02:00

Select

The closest Stage to your date is: `6.35` which corresponds to: `2016-05-25 04:00:00`

Fig. 20: Stage prediction based on the closest date-time introduced

6 Contributions

In this section, the team member contributions are briefly detailed:

- **Gerardo Villegas Contreras** : Calendar, interface and model development.
- **Renata Uribe Sánchez**: Chatbot, paper writing, data processing.
- **Ivan Emmanuel Gutiérrez Y López**: Model adjustment, development and data pre-processing.
- **Paola Naomi García Reyes** : Model development, pipeline, cloud adjustment.

7 Appendix

7.1 Resource code

Github repository, including all the model implementations is available at `https://github.com/PaolaGarci/IA-pt2.git`

7.2 Interface

The user interface demo can be found on `https://grip-stage-predictor.streamlit.app/`

7.3 Feature extracted variables

References

1. Chapman, K. W., Gilmore, T. E., Chapman, C. D., Mehrubeoglu, M., Mittelstet, A. R. (2020). Camera-based Water Stage and Discharge Prediction with Machine Learning. Hydrology and Earth System Sciences Discussions, 1-28.
2. Davie, Tim. (2008). Fundamentals of Hydrology. Abingdon, Oxon: Routledge.
3. Water Science School. (2018). Dictionary of Water Terms. USGS, science for a changing world
4. Chapman, P., Clinton, J., Kerber, R., Khabaza, T., Reinartz, T., Shearer, C. and Wirth, R. (2000). CRISP-DM 1.0 Step-by-step data mining guide. The CRISP-DM consortium .
5. edited Robin Mackay and Armen Avanessian. (2014). Accelerate : the accelerationist reader. Falmouth, United Kingdom : Berlin :Urbanomic Media Ltd. ; in association with Merve
6. Xue, Minglong & Shivakumara, Palaiahnakote & Wu, Xuerong & Lu, Tong & Pal, Umapada & Blumenstein, Michael & Lopresti, Daniel. (2020). Deep invariant texture features for water image classification. SN Applied Sciences. 2. 10.1007/s42452-020-03882-w.

Table 6: Feature extracted variables

Variable	**Description**
SensorTime	Sensor time and date
CaptureTime	Capture time and data
Filename	File name, corresponding to time and date
Agency	Capturing agency
SiteNumber	ID of the site where the capture was made
TimeZone	Three letters, time zone
Stage	Water level of the stream, measured in meters
Discharge	Volume of water that runs through a specific location at a specific time, normally expressed in cubic meters per second
CalcTimestamp	Capture date/time
width	nNumber of pixels across the width of the capture
height	Number of pixels across the height of the captures
exposure	Amount of light in the image
fNumber	Focal ratio of the lens
isoSpeed	Camera's sensibility to light
shutterSpeed	Time the lens was open
grayMean	Average of pixel intensities after conversion to grayscale
graySigma	Sum of pixel intensities after converting them to grayscale
entropyMean	Average Shannon entropy of pixels in grayscale
entropySigma	Shannon entropy sum of the grayscale pixels
hMean	Hue average taking each pixel in HSV format
hSigma	Hue summation taking each pixel in HSV format
sMean	Saturation average taking each pixel in HSV format
sSigma	Saturation summation taking each pixel in HSV format
vMean	Value averaging taking each pixel in HSV format
vSigma	Sum of value by taking each pixel in HSV format
grayMean 0	Average intensity of the pixels in grayscale, above the waterfall
graySigma 0	Sum of intensity of grayscale pixels above the waterfall
entropyMean 0	Average entropy of grayscale pixels after waterfall
entropySigma 0	Entropy sum of grayscale pixels after water fall off
hMean 0	Average hue taking each pixel in HSV format, after water fall-off
hSigma 0	Hue summation taking each pixel in HSV format, after the waterfall
sMean 0	Saturation average taking each pixel in HSV format, after water fall off
sSigma 0	Saturation summation taking each pixel in HSV format, after waterfall
vMean 0	Average value taking each pixel in HSV format, after waterfall
vSigma 0	Sum of value taking each pixel in HSV format, after the waterfall
grayMean 0	Average intensity of pixels in grayscale, before water drop
graySigma 0	Sum of intensity of pixels in gray scale, before water drop
entropyMean 0	Average entropy of grayscale pixels, before water fall off
entropySigma 0	Entropy sum of grayscale pixels, before water fall-off
hMean 0	Average hue taking each pixel in HSV format, before water fall-off
hSigma 0	Hue summation taking each pixel in HSV format, before water fall-off
sMean 0	Average saturation taking each pixel in HSV format, before waterfall
sSigma 0	Saturation summation taking each pixel in HSV format, before water drop

Table 7: Feature extracted variables

Variable	**Description**
vMean 0	Average value taking each pixel in HSV format, before the waterfall
vSigma 0	Sum of value taking each pixel in HSV format, before the waterfall
WeirAngle	Angle where the waterfall is located
WeirPt1X	Far point for waterfall calculation, X axis
WeirPt1Y	Far point for water fall calculation, Y-axis
WeirPt2X	Near point for water fall calculation, X-axis
WeirPt2Y	Near point for water fall calculation, Y-axis
WwRawLineMin	Minimum distance between water fall and white water
WwRawLineMax	Maximum distance between water fall and white water
WwRawLineMean	Average distance between the water fall and the white water
WwRawLineSigma	Sum of distances between water fall and white water
WwCurveLineMin	Minimum distance between the water fall and the white water curve
WwCurveLineMax	Maximum distance between the water fall and the white water curve
WwCurveLineMean	Average distance between water fall and white water curve
WwCurveLineSigma	Sum of distances between water fall and white water curve
WwCurveLineMean	Sum of distances between water fall and white water curve