Co-UDlabs

Data Storage Report

CoUDlabs_WP8_T811_UDC_002

Application of imaging velocimetry techniques for urban drainage applications

This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement N°101008626

DATASET DETAILS

Project acronym	**Co-UDlabs (www.co-udlabs.eu)**
Project title	Building Collaborative Urban Drainage research labs communities
Call identifier	H2020-INFRAIA-2020-1
Grant Agreement No	101008626

Dataset Information	
Activity	Joint Research Activities
Dataset ID	CoUDlabs_WP8_T811_UDC_002
Dataset title	Application of imaging velocimetry techniques for urban drainage applications
Data sources	Data from laboratory experimental campaign where the water surface velocities were measured from LSPIV and low-cost cameras.
Content	Raw and processed images of surface runoff and surface velocity maps resulted.
Formats	jpeg, txt,
Volume	9 Gb

AUTHORS

Name	**Institution**	**Email**	**ORCID ID**
Juan Naves	UDC	juan.naves@udc.es	0000-0001-9031-756X
Daniel Carreres	UDC	daniel.carreres@udc.es	0000-0002-1852-1053
Jose Anta	UDC	jose.anta@udc.es	0000-0003-2002-0618

LICENSE

Abstract

This document describes the dataset 'Identifying sediment deposits from temperature signals' which is a result from the Joint Research Activity 3 (WP8, Improving Resilience and Sustainability in Urban Drainage solutions), Task 8.1.1. (Development of Scalable Hydrodynamic Performance Protocols.) within Co-UDlabs project, funded under the European Union's Horizon 2020 research and innovation programme under grant agreement No 101008626.

This dataset includes the data obtained during the installation of a camera system and development of a LSPIV methodology to determine the velocities of runoff generated in the BLOCK rainfall simulator located at the Universidade da Coruña. The dataset consists of raw and processed images and surface velocity maps obtained during the assessment of the full process. This data will help to improve and assess imaging techniques in urban environments with the presence of realistic rain, and test new preprocessing and postprocessing methods to optimize the different processes.

1. Experimental setup

Raspberry Pi units have been employed in the context of video recording and processing. Beyond their affordability, these compact, single-board computers are recognized for their open architecture and flexibility. They are equipped with a Camera Serial Interface port for camera module attachment through a ribbon cable and a range of general-purpose input/output (GPIO) pins. Furthermore, they offer the convenience of remote control via their integrated Wi-Fi connectivity. Here, two different configurations were used: i) four Raspberry Pi 4B paired with an HQ Camera Module and ii) four Raspberry Pi Zero W paired with a v2 Camera Module. A total of 4 installation points (A, B, C and D in Figure 4.1) have been stablished to cover the full facility. Each point consists of 2 LED lamps and 2 UV lamps connected through GPIO to a Raspberry Pi 4, which has a HQ Camera. Additionally, a Raspberry Pi Zero W paired with a v2 Camera Module is also installed at each point to have a wider frame of the surface. All the raspberries were connected through WIFI to a central computer that can control the image acquisition and lighting of the installation.

Figure 1. Installation of cameras based on Raspberry Pi and lamps in BLOCK. Four positions distributed along the facility were selected to hold two different image acquisition systems and LED and UV lamps.

2. LSPIV methodology

Once each camera was pointed to the measurement areas, a rectification of the image is needed. This procedure was carried out by means of a chess pattern carpet. The uniformity of the sizes of the squares means that the coordinates of the squares are known and allows the rectification of the image projection produced by the oblique focus of the cameras to be carried out. This strategy also allows the results obtained in each measurement area to be positioned in a global reference system for the positioning of the results. For this purpose, images are taken from the cameras and the real and pixel coordinates of four points chosen to cover the four corners of the images obtained are determined. These coordinates are used to rectify the obtained frames using the Python library 'OpenCV' (Bradski, 2000) as seen in Figure 4.2. The methodology is similar to that followed in previous works described in Naves et al. (2021, 2020, 2019).

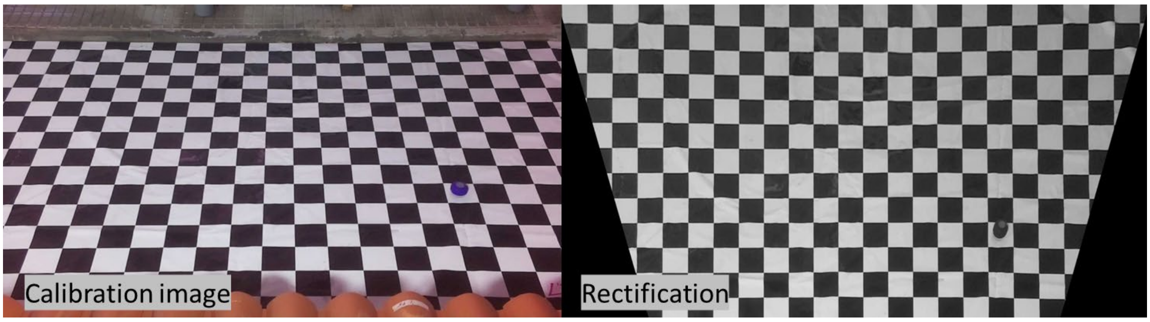

Figure 2. Calibration image with the chess pattern carpet disposed over the surface to get the real coordinates to rectify the frames (left), and result of the rectification over the calibration image (right).

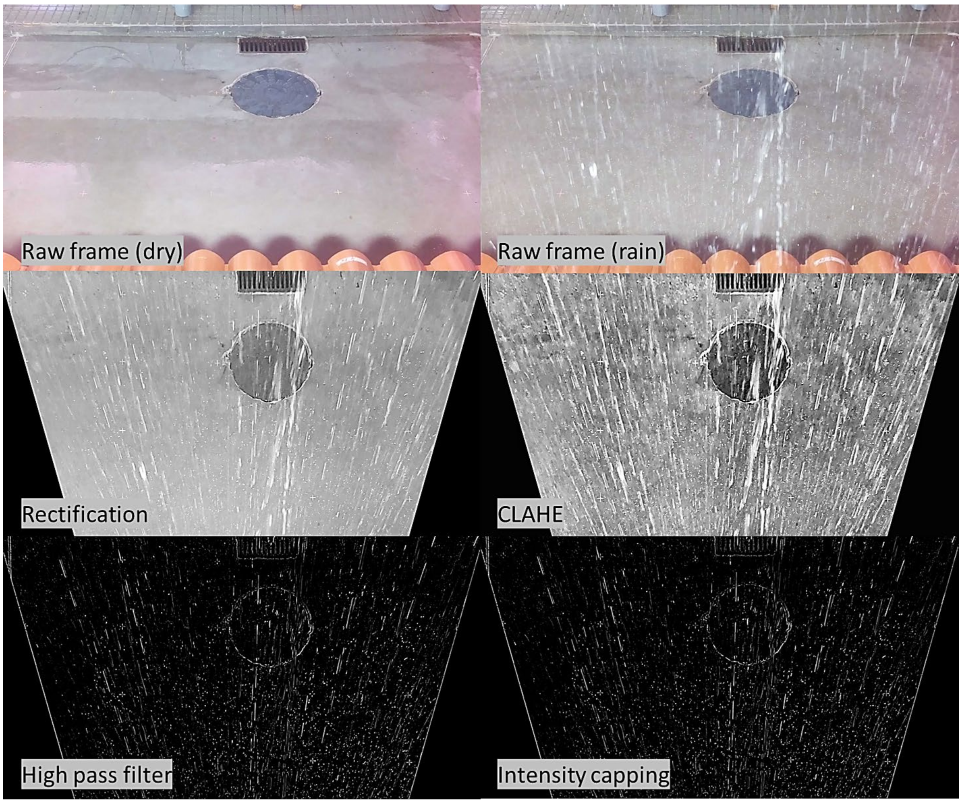

Figure 3: Example of the treatment of the frames including rectification of the recording projection, processing and filtering of the images recorded in order to highlight tracers and improve PIV cross-correlation results. The frame corresponds to the camera equipped with a picamera v2 at position A and the H3 experiment.

The rectified images are then filtered using the Contrast Limited Adaptive Histogram Equalization (CLAHE) image processing technique, a high pass filter and the intensity capping procedure to highlight the particles that will be used as tracers to estimate the surface velocities of the flow. In the tests carried out in this experimental campaign, the tracers used are air bubbles originating naturally from the impact of raindrops on the surface. Figure 4.2 shows an example of original, rectified and filtered frame.

The open access software OpenPIV (Liberzon et al., 2022) was applied to the rectified and filtered frames recorded. The raw velocity results were filtered to remove erroneous vectors using the noise ratio from cross-correlation function, a median spatial filter and a minmax filter. The mean result of 20 consecutive pairs of frames was then obtained considering a median temporal filter. The results are finally scaled, and the reference system is established. Figure 4.3 includes an example of the postprocessing procedure applied where erroneous vectors were removed from raw results.

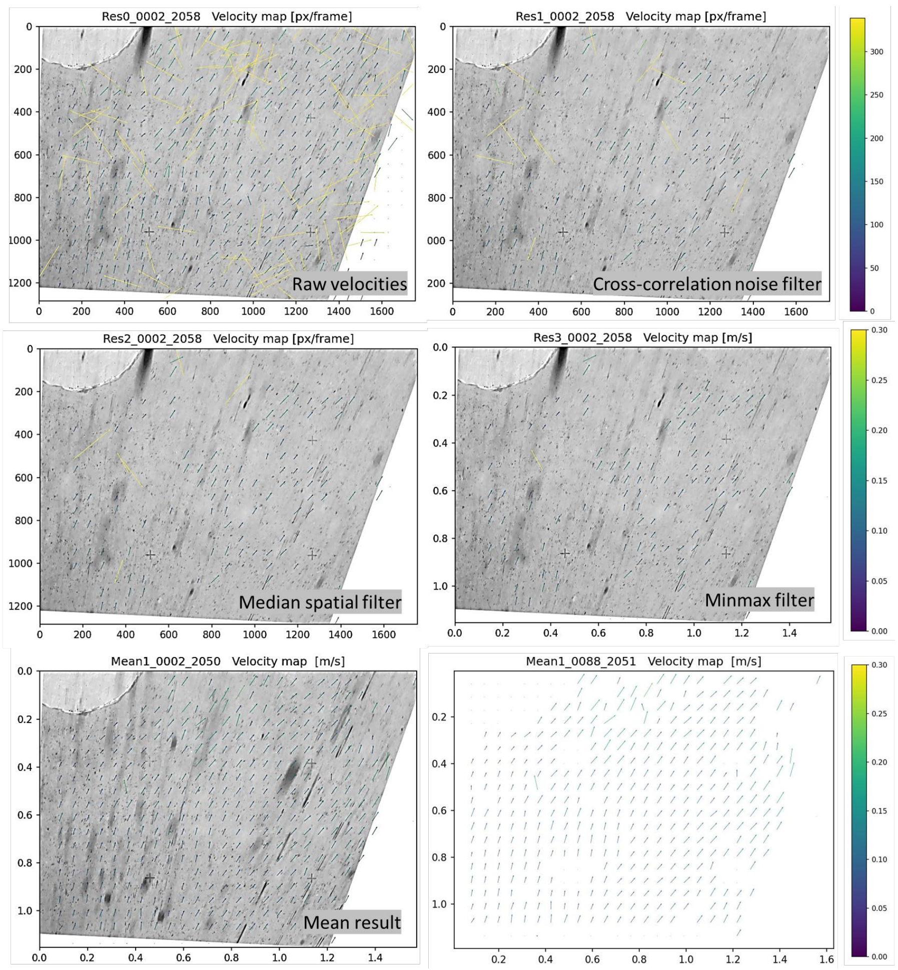

Figure 4: Example of postprocessing of raw vectors for H3 test using a raspberry equipped with the HQ camera placed on position A.

3. Test procedure

Six different tests were carried out to measure 2D velocities on the surface of the facility (Figure 4.4). First, three constant rain intensities of 30, 50 and 80 mm/h were simulated during 240 s (H1, H2 and H3). This corresponds to steady conditions tests where the three rain intensities that the rainfall simulator can generate were simulated. Then, two intermittent experiments were carried out following Figure 4.4 (H4 and H5) where short periods of rain were generated between gaps of 45 s without rain. During the last experiment the rain intensity was increased and then decreased each 30 s between the available rain intensities (H6). These last experiments pretend to analyse the performance of the methodology for transitory conditions.

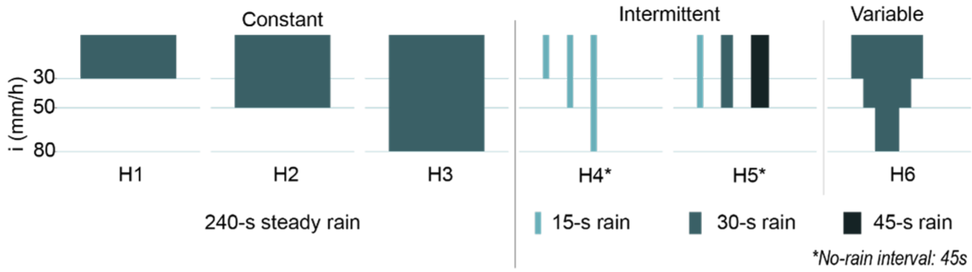

Figure 5: Test carried out during the experimental campaign.

All the cameras were recording frames during the experiments. The acquisition frequency was 10 Hz. A record was kept of the time taken to record each frame, so that possible delays that sometimes occur with the raspberry Zero due to its low processing capacity can be taken into account in the calculation of the velocities. Calibration, preprocessing and postprocessing parameters were optimized and kept common to all the experiments. Finally, each result is a mean of 20 pair of frames, which is equivalent to 2 s with the 10Hz frequency.

4. Data and result files organization

The dataset is structured following first the six experiments performed (H1 to H6). One folder for each of the eight cameras installed is contained inside each experiment. Finally, for each camera folder, the data storage is made following the following folder structure:

Test name:

Camera:

- 0_Frames: Original frames numbered, including also the timestamp with the time when each frame was obtained.
- 1_Rectification: Rectified frames. Only saved for some frames. Example in Figure 3.
- 2_FilterCLAHE: Filtered rectified frames using CLAHE filter. Only saved for some frames. Example in Figure 3.
- 2_FilterHP: Filtered frame adding a high pass filter. Only saved for some frames. Example in Figure 3.
- 2_FilterIC: Filtered frames adding an intesity capping filter. Only saved for some frames. Example in Figure 3.
- 3_Results: Partial and final results saved:
  - Pair_0: Raw instantaneous results (pixel/s). Only saved for some frames.

- Pair_1: Algorithm noise filtered results (pixel/s). Only saved for some frames.
- Pair_2: Instantaneous postprocessed results (pixel/s). Only saved for some frames.
- Pair_3: Instantaneous postprocessed results (mm/s). Only saved for some frames.
- Mean_0: Mean results without additional filtering (mm/s). Only saved for some frames
- Mean_1: Mean results applying a temporal median filter (mm/s).

Each result folder contains: Plot images with vectors (jpg), 'Res' txt files with coordinates and velocity components (x,y,u,v) for each measured velocity vector and a 'Timestamp.txt' file with the time corresponding for each result obtained.

The rest of folders corresponding to image processing contain raw and processed images (jpg) and timestamp files (txt).

The join plots when the results are represented with a global reference system for all the facility surface were included in the folder 'JoinResults' placed at the same level as the experiments folder. An example of the joined results is included in Figure 6. Examples of timeseries for all the tests are also included in the folder 'timeseries'.

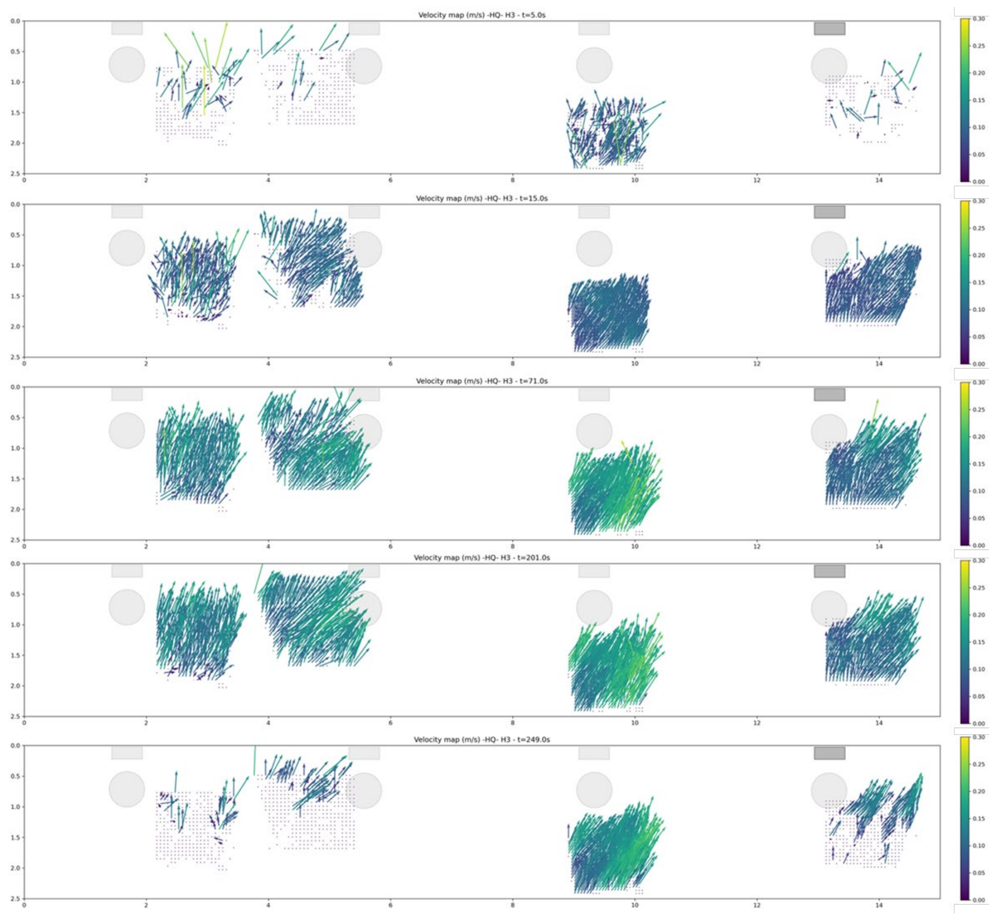

Figure 6. Surface velocity maps obtained from HQ cameras at different times of test H3 corresponding with 5, 15, 71, 201 and 249 s from the start of a constant 80 mm/h 240 s rain.

5. REFERENCES

Bradski, G. (2000). The openCV library. Dr. Dobb's Journal: Software Tools for the Professional Programmer, 25(11), 120-123.

Liberzon, A., Lasagna, D., Aubert, M., Bachant, P., Käufer, T., Bauer, A., Vodenicharski, B., Dallas, C., Borg, J., tomerast & ranleu. (2020). OpenPIV/openpiv-python: OpenPIV - Python (v0.22.2) with a new extended search PIV grid option (0.22.2). Zenodo. https://doi.org/10.5281/zenodo.3930343

Naves, J., García, J. T., Puertas, J., & Anta, J. (2021). Assessing different imaging velocimetry techniques to measure shallow runoff velocities during rain events using an urban drainage physical model. Hydrology and Earth System Sciences, 25(2), 885-900.

Naves, J., Anta, J., Suárez, J., & Puertas, J. (2020). Hydraulic, wash-off and sediment transport experiments in a full-scale urban drainage physical model. *Scientific data*, *7*(1), 44.

Naves, J., Anta, J., Puertas, J., Regueiro-Picallo, M., & Suárez, J. (2019). Using a 2D shallow water model to assess Large-Scale Particle Image Velocimetry (LSPIV) and Structure from Motion (SfM) techniques in a street-scale urban drainage physical model. *Journal of Hydrology*, *575*, 54-65.