Master's thesis

NTNU
Norwegian University of Science and Technology
Faculty of Engineering
Department of Civil and Environmental Engineering

Anders N. Løfald

Using Optical Velocimetry to Detect Illicit Inflow to Sewers:

A Sensor Strategy

Master's thesis in Civil and Environmental Engineering
Supervisor: Francois Henri Leon Clemens-Meyer
Co-supervisor: Franz Tscheikner-Gratl
June 2023

NTNU
Norwegian University of
Science and Technology

Anders N. Løfald

Using Optical Velocimetry to Detect Illicit Inflow to Sewers:

A Sensor Strategy

Master's thesis in Civil and Environmental Engineering
Supervisor: Francois Henri Leon Clemens-Meyer
Co-supervisor: Franz Tscheikner-Gratl
June 2023

Norwegian University of Science and Technology
Faculty of Engineering
Department of Civil and Environmental Engineering

NTNU
Norwegian University of
Science and Technology

Using Optical Velocimetry to Detect Illicit Inflow to Sewers: A Sensor Strategy

Anders N. Løfald

08/06/2023

Abstract

The illicit inflow of water to urban drainage systems refers to the unauthorized entry of water, through improper disposal or wrongly made connections to drainage systems causing inflow of sewer to stormwater systems and stormwater to sewer systems. This has the potential to increase operating costs and could cause an exceedance of the capacity of the systems. The predominant strategies for detecting illicit connections in sewer systems are inefficient because the most effective methods, which are also the most time-consuming, must be applied throughout the entire system. Illicit connections often occur when storm drains from houses and streets are incorrectly connected, leading to an influx of excess water into the sewer systems. This causes a temporary increase in discharge and alters the water composition. Detection can be achieved through intrusive techniques such as sampling, flow measurement, or Distributed Temperature Sensing (DTS). However, these methods suffer from being either too expensive, too inaccurate, or too time-consuming. A non-intrusive sensor strategy is being researched in this master thesis project, involving the use of cameras, Computer Vision (CV) and Optical Velocimetry (OV) algorithms to observe increased flow during periods following rain events in order to detect illicit connections to the system. This sensor strategy is being applied in a real sewer system in the Municipality of Tilburg, NL. A detailed account of the process and concepts will be provided in this master thesis.

Summary

The illicit inflow of water to urban drainage systems refers to the unauthorized entry of water, through improper disposal or wrongly made connections to drainage systems causing inflow of sewer to stormwater systems and stormwater to sewer systems. This has the potential to increase operating costs and could cause an exceedance of the capacity of the systems. Mitigating illicit inflow in sewer systems reduces the potential for environmental pollution and decreases the cost of operation by removing excess water. A cost-effective and efficient method for identifying and locating illicit connections is needed since present methods are time-consuming and may not always provide reliable results. Illicit inflow can be detected by correlating rain data with increased velocity observed in the system. Capturing videos with a camera and processing them with Optical Velocimetry (OV) algorithms to produce a series of velocity estimates could be a good non-intrusive alternative to other velocity measurement methods.

Three different OV algorithms were shown capable of estimating the flow velocity in the context of sewers when processing footage from both high-quality and lower-quality videos. However, the success of the different algorithms is very dependent on image quality, wastewater characteristics, and particle characteristics. The detection sensitivity of the algorithms and the variation in velocities produced in a conducted experiment showed that Optical Flow (OF) and Particle Tracking Velocimetry (PTV) are less susceptible to image noise when compared to Particle Image Velocimetry (PIV), making them better suited for distinguishing the presence of flow from no flow. PTV also produced the series with the least variation between velocities estimated from each frame of the videos. The experiment also proved that increasing the time between evaluations of displacement effectively reduces uncertainty and increases sensitivity. Furthermore, the experiment showed that using ultraviolet light in combination with visible light could be beneficial compared to using only visible light when recording footage for velocimetry purposes in sewers.

The results from the "in-field" implementation of the OV camera sensor were hindered by camera problems. The limited amount of data collected from this implementation is inadequate to definitively prove that the sensor strategy, implementing hydraulic simulation in sensor placement, verification of results, and source detection, works. Nevertheless, the field implementation demonstrated how the camera sensor can be placed, and the data showed, despite the lower

quality footage obtained in the field study, that all three algorithms are able to distinguish between the presence of low or no flow and periods with higher flow when averaging frame velocities over a 30-second sequence. Therefore, it would be possible to detect illicit inflows as deviations occurring during periods when the system runs dry. The biggest limitation to the success of the method is the video quality. PTV also suffers from insufficient image quality, but it also showed, in the field implementation, the ability to accurately represent the widest range of velocities, even from poor-quality footage.

Acknowledgements

The research project, which is reflected in parts of this master's thesis, was made possible by the sewer inspection and renovation company vandervalk+degroot, NL. They provided the necessary resources, including cameras and equipment, to test the method during a three-month-long internship at the company. Special gratitude goes to Ronald van de Waerdt for his support and trust in the project, as well as to Rene van Vliet for letting me clutter his office with cameras, mounts, and duct tape.

The contributions and support of master thesis supervisor Prof. Francois Clemens throughout this project are greatly appreciated. His unwavering belief in the project and availability to answer questions and provide assistance at short notice were truly invaluable. Additionally, sincere thanks are extended to Ph.D. Antonio Moreno Rodenas for making himself available to provide the particle tracking velocimetry algorithm.

Contents

Chapter 1

Introduction

Sewer systems are essential components of modern urban infrastructure as they provide a safe and efficient means of collecting and disposing of wastewater generated by residential, commercial, and industrial activities. Butler and Davies (2000, p. 18) categorizes conventional sewer systems into two types: Combined Sewer Systems (CSS) and Sanitary Sewer Systems (SSS). CSS collects both wastewater and stormwater in a single pipe network, while SSS collects only wastewater from residential, commercial, and industrial sources. Improper connections to SSS, such as illegal discharges from industrial facilities or direct connections with storm drains, can compromise the effectiveness and sustainability of these systems. These illicit connections can lead to increased pollutant loads in natural water bodies, exceed system capacity, increase water treatment costs (more energy and chemicals), and potentially decrease treatment effectiveness.

Identification of illicit connections in sewer systems is a complex and challenging task due to several factors. Sewer networks are often extensive and intricate, and the identification of an illicit connection within them can be time-consuming, requiring a detailed understanding of the system's layout and operation. Furthermore, the detection of illicit connections can be costly, necessitating specialized equipment and techniques for accurate identification. In some countries, it may only be legal to enter private property with permission, making detection techniques rely on indirect methods. The process can also be hazardous, posing health risks to workers involved in the investigation. The methods currently used to identify illicit connections are time-consuming and may not always provide reliable results in sewers compared to stormwater systems. Further research is needed to develop cost-effective and efficient methods for identifying and locating illicit connections, to ensure that existing infrastructure can cope with the increasing pressure on drainage systems caused by population growth and more extreme weather events.

A comprehensive overview of various Sewer Inspection Methods (SIMs) employed in practice is presented by Sadeghikhah et al. (2022), categorizing them into three distinct tiers. These methods range from assessing the general structural health of the system, involving spatial mapping and modelling, to more specific methods for investigating the temporal behaviour of the system and pinpointing problem locations. Beheshti, Sægrov and Ugarelli (2015) distinguishes between quantitative methods for assessing magnitude, volume, and discharge, and qualitative methods for source detection of Infiltration/Inflow (I/I) in drainage systems. The qualitative methods discussed include smoke testing, dye testing, Distributed Temperature Sensing (DTS), and Closed Circuit Television (CCTV). The article also reports on the limitations of these different methods. DTS and CCTV

both require expensive equipment and rely on the occurrence of inflow, which can make them time-consuming in order to avoid false negatives. Dye testing, on the other hand, is only possible under certain flow conditions, is labour-intensive, and requires access to private property. While smoke testing is less accurate and is deemed ineffective by Beheshti and Sægrov (2019) in finding all types of infiltration/inflow sources in sewer networks on its own, suggesting it is more effective when combined with other methods.

By strategically employing high-resolution cameras within the sewer network, flow patterns can be observed, enabling the identification of anomalous flow patterns after precipitation events. The rationale behind this approach is that stormwater entering through illicit connections leads to rapid changes in discharge volume and an almost proportional increase in velocity. Analyzing captured imagery for variations in flow velocity can, as a result, reveal the presence of illicit connections. By incorporating this into a smart sensor strategy that utilizes hydraulic simulation for sensor placement, validation of results, and source location, a non-invasive and efficient means of detecting illicit connections in urban sewer systems is established.

The goal of the work presented here is to investigate the ability of optical velocimetry techniques to measure flow velocity in sewer networks and identify illicit inflow of stormwater by capturing sequences of videos at specific intervals and durations. Patterns in the observed velocity variation are then correlated with the occurrence of storm events to provide a strong indication of the presence of illicit connections. Several optical velocimetry algorithms were tested to determine their effectiveness under different settings, conditions, and physical setups. The detection level of the method is assessed by analysing the uncertainty of the measurements, conducting hydraulic simulations, and applying the method in a field study. A sensor strategy is also proposed to identify suitable locations for camera placement and to verify the results based on hydraulic simulations of the modeled system.

Chapter 2

Background

Using digital image analysis to produce real-world measurements has emerged as a promising method for monitoring water for a range of different purposes. Moreno-Rodenas et al. (2021) employed it for monitoring the accumulation of fat, oil, and grease in the sumps of wastewater pumping stations, while Legleiter and Kinzel (2021) estimated surface flow velocities from satellite video, and Kim et al. (2014) estimated flood runoff from CCTV footage. Zeng et al. (2021) also showed that a low-cost digital camera colorimetry setup can be used to investigate quantitative relationships between watercolour and concentrations of dissolved and particulate materials. Especially, the advantages of low power and low energy consumption were found by Ufuoma et al. (2021) to make camera sensors viable and recommended as an alternative technology for flood monitoring when assessing the accuracy of camera vision (CV) for immediate water level monitoring.

In the context of sewers, Meier et al. (2022) demonstrated how footage from cheap, consumer-grade cameras can be used in a real-time flow sensor without relying on trackable particles by training a Convolutional Neural Network (CNN) to extract surface velocity and water level, which are then used to approximate the flow in real-time. Jeanbourquin et al. (2011) concluded their investigation into using automatic flow velocity algorithms in combined sewer overflow monitoring by stating it as a feasible means to measure and better understand sewer flows and hydraulics. The capability of Optical Velocimetry (OV) was proven by Naves et al. (2021) by showing that both techniques with and without the addition of seeding particles provided satisfactory velocity distributions in cases of unidirectional flows when assessing different imaging velocimetry techniques to measure shallow runoff velocities during rain events.

The possibilities that lie within digital image sensors also interest vandervalk+degroot (V+G), a sewer inspection and renovation company located in the Netherlands and Belgium. The company brands itself by "innovating wherever it is possible" and has already been involved in projects using optical measurements for sewer purposes and wants to investigate the possibility of using cameras in illicit inflow detection. V+G has been appointed to search for illicit connections in drainage and sewer systems in suspected areas throughout the Municipality of Tilburg, Netherlands. In the project outline, they proposed the possibility of using cameras placed in manholes over a certain period to measure an increase in inflow in relation to rain events with 5-10 mm/hr intensity. Sewer Robotics, a manufacturer and innovator of trenchless technology based in the Netherlands, already manufactures cameras suited for mounting on jetting hoses for recording while cleaning the network, combining cleaning and inspection. These cameras

are waterproof, robust, and have a built-in light source that could make them suitable for use also in stationary surveying within the sewer. This is a win-win situation for both companies as multipurpose equipment defends the investment, and Sewer Robotics can further develop an existing product. A three-month-long internship was conducted by the thesis author at V+G to research the proposed method and to help further develop the product.

Chapter 3

Method

3.1 Camera Vision To Detect Illicit Inflow

Increased inflow in open channel conduits leads to both an increase in flow velocity and water depth. The exact relationship between inflow, velocity, and depth is not unambiguous as it depends on various factors such as channel geometry, roughness, and fluid viscosity. To determine the exact discharge, the channel geometry, depth, and velocity must be known. However, when the open channel is long enough and the flow rate remains constant for an extended period of time, the flow in the channel will approach uniform conditions. This means that the depth, known as the normal depth under such conditions, is the same at all points along the channel. The normal depth can then be used as an approximation of the actual depth of flow within a channel segment, according to Brunner (1997, p. 6-16), and this depth is proportionate to the discharge. This implies that the velocity also increases proportionally. However, the assumption of normal depth is not always true under transient conditions, but the velocity can still be used as a measurement proxy for the discharge by assuming that depth and velocity increase in a somewhat proportionate manner.

If the discharge of sewer is measured at a high enough frequency, certain regularities are likely to emerge. A pattern showing daily fluctuations, differing between working weekdays and weekends, is to be expected when the flow rate becomes the average of enough people's daily routines, and contributions from industrial processes occur at regular intervals. When patterns are observable, detecting illicit inflow becomes a matter of measuring deviations from this pattern. However, when the number of contributors upstream is low, each upstream connection's contribution becomes relatively larger. This could result in more rapid fluctuations of discharge and the absence of a typical observable diurnal pattern. In such cases, it is still possible to distinguish values that deviate by being greater in magnitude from the normally observed velocity values. However, it is not possible to detect anomalies based on deviations from a pattern as the "noise" is greater than the "signal" of deviation. Fortunately, there is one pattern that (almost) always is present in wastewater flow, which is the pattern of a small discharge (infiltrating groundwater) or no flow during the night. If flow is measured during time windows of the day when the system normally runs dry and the increase corresponds somewhat in time and duration with a rain event, this is a strong indication that an illicit storm drain or roof drainage is connected somewhere upstream from the measuring location.

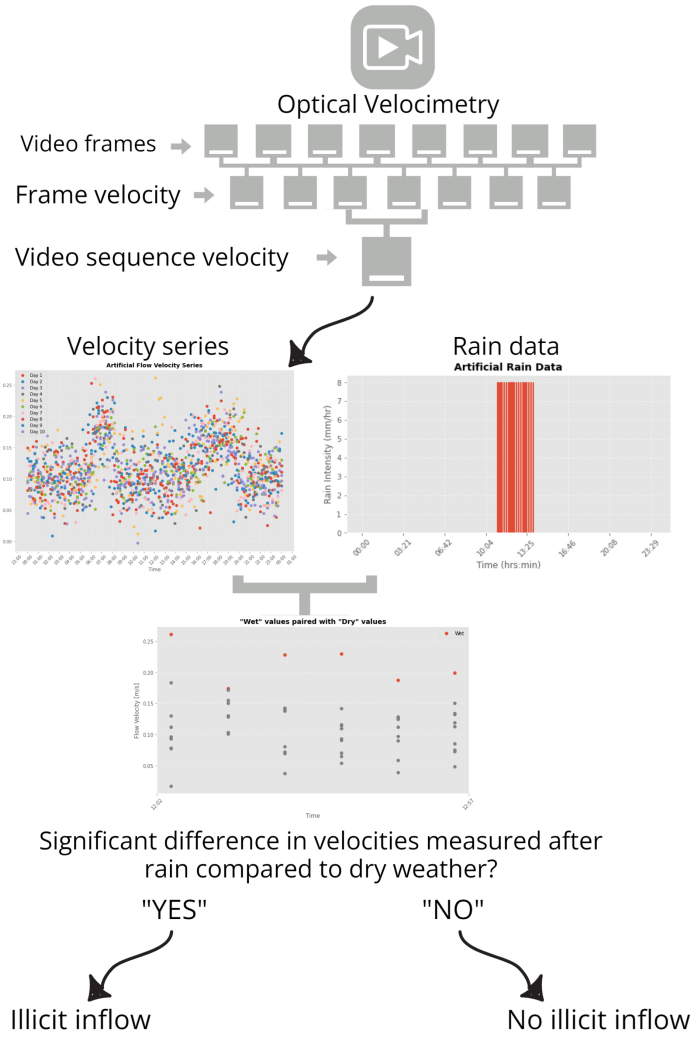


Figure 3.1: Evaluation of pairs of consecutive frames from a video is producing frame velocities that are averaged over all frames to produce a velocity representative for the time the video was captured. A velocity time series is produced from videos captured at intervals throughout the day that can be correlated with rain data to observe if there is a significant deviation in velocities produced during and after storm events to detect illicit inflow.

Establishing a time series of the flow is done by capturing sequences of videos at certain intervals throughout the day. The sequences are then processed by Optical Velocimetry (OV) algorithms to produce a representative velocity value for that time interval by averaging the velocities estimated from measuring displacement between frames in the video. The averaged value is less influenced by random errors caused by the algorithms and takes into account the possible variation

in flow-related features tracked within the video sequence. Averaging velocities over time only makes sense if the velocities are steady during the duration of the captured sequence. This is based on the assumption that the timescale of change is high due to the system's believed ability to absorb peaks. This ability arises from the fact that conduits are designed to accommodate higher flow volumes while maintaining stable flow velocities, that there is some storage capacity in the system enabling the storage of excess flow, and that the interaction of different sources of flow smooths out short-term fluctuations.

The velocity estimated from each video sequence contributes to a time series. When several time series are created, which are known to not be influenced by rain-induced inflow, the 'normal' pattern can be established. Values produced after the onset of storm events can then be compared to the "normal" pattern using statistical tests to determine if they deviate significantly enough to indicate that they are caused by the inflow of rainwater. This is illustrated in Fig. 3.1.

3.1.1 Velocity Measurement Using Optical Velocimetry

Optical Velocimetry (OV) can be used to visualize and estimate velocity in fluids by tracking distinguishable features interpreted from differences in pixel intensities between frames, which result from the movement of objects within the water body or on the water surface. This is achieved by tracking particle clusters for Particle Image Velocimetry (PIV), distinct particle features for Particle Tracking Velocimetry (PTV), and by tracking pixel intensity gradients over time for Optical Flow (OF). A very brief and simplified explanation of the three algorithms will be given in the following sections. The formulas and mathematics involved in PIV and OF are thoroughly explained in the book "Particle Image Velocimetry" by Adrian and Westerweel (2011) and the article "Determining Optical Flow" by Horn and Schunck (1981).

For the computer vision algorithms to track and analyse the motion of particles and structures accurately, the flow must exhibit a sufficient level of contrast and distinguishable features. The objects' velocities are used as an estimate of the water flow velocity and rely on the implicit assumption that the objects move with the same velocity as the water. The truth of this assumption depends on the objects present and general flow characteristics and will not always hold. The particles' slip velocity, as defined by Deshpande and Tallapragada (2018), is the relative velocity of the particle with respect to the undisturbed fluid velocity, and different particles will have different slip velocities. Thus, particles can have different velocities when passing the camera at the same flow rate. This could be taken into account by estimating the slip velocity and adding it to the estimated particle velocity or by extracting only the highest velocities assuming that they are representative of the particles that travel with velocities closest to the flow velocity.

The analysis with the OV algorithms is limited to representing two-dimensional flow when using a one-camera setup. This neglects the possibility of any out of plane component to the motion of the objects and does not take into

consideration the varying velocity profile in the vertical cross-section. The varying velocity profile has the potential to produce greater variation in the measurements if the particles present are not uniformly distributed in the profile. Particles on the surface will give higher velocity estimates than particles travelling along the bottom. The contribution from this is though believed to be small relative to other error sources. Any out-of-plane motion will also introduce error as the actual displacement of the particle is longer than what is seen in the images. The effect depends on the direction and magnitude of the out-of-plane component relative to the in-plane motion of the objects, but it is assumed to be small within the short time window in which the particles are observed.

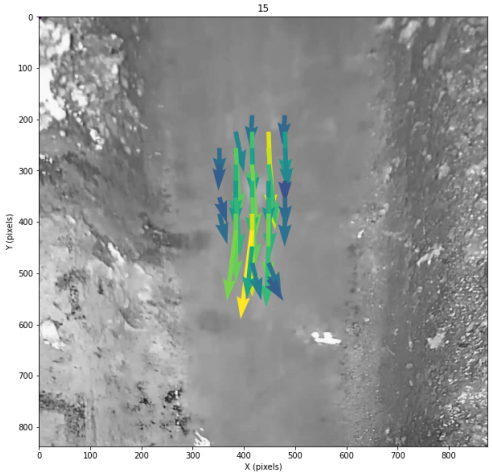
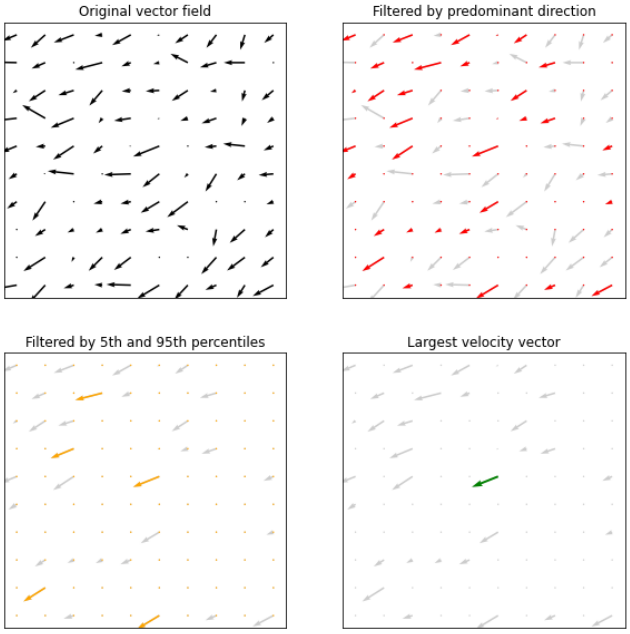


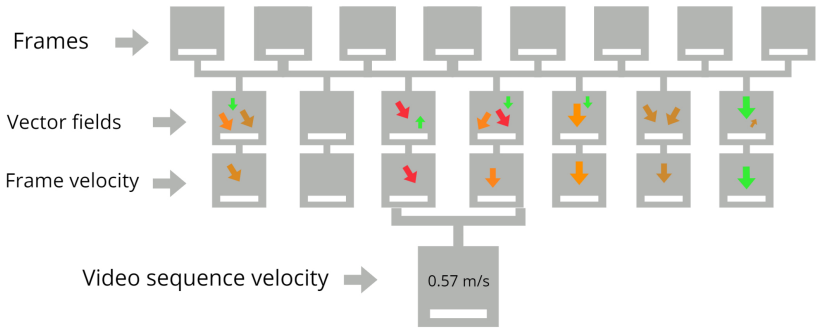
Figure 3.2: A field of vectors is produced for each frame representing the displacement found when evaluating pairs of frames in the video.

The algorithms provide a vector field of different resolutions representing velocities for each pair of consecutive frames from the video (Fig. 3.2). The vectors represent what the algorithms interpret as displacement between frames. Some of these vectors can be used to estimate flow velocity, while others are spurious and caused by image noise, falling dust, or insects. A filtering process is consequently applied to the vectors to remove those that deviate from the predominant direction and those with magnitudes higher and lower than certain percentile values, depending on the assumed initial accuracy of the algorithms. The remaining velocities are then used in the estimation of a frame velocity representing each frame of the video. The method of estimating the velocity for each sequence may differ between implementations. This is particularly related to the amount of particles present in the videos. Ideally, there is a continuous stream of particles to track in the videos, but that is likely not the case most of the time. Therefore, the computation of the velocity for each sequence should somehow reflect the frames in which the algorithm has particles to track, while also being the value that is most consistently represented in all the video sequences to ensure that a difference between measurements represents an actual change in flow and not just a result

of there being fewer moving objects to track. The process of filtering velocities and retrieving velocity from the video sequence is visualized in Fig. 3.3.



(a) Velocities found from each pair of frames evaluated in the video sequence are filtered based on predominant direction and magnitude.



(b) A vector field is produced for each pair of evaluated frames from which a frame velocity is extracted and then averaged to produce the video sequence velocity.

Figure 3.3: The process of retrieving velocities from videos.

3.1.2 Optical Velocimetry Algorithms

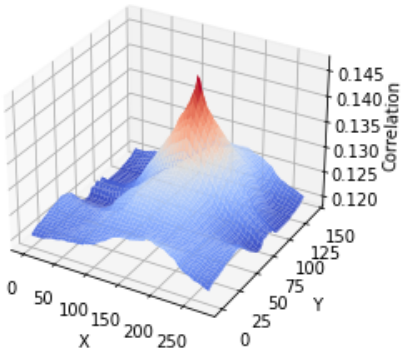
The general workflow for the OV algorithms is to read a series of images/frames, possibly from a video, and compare the present frame to a reference frame, which

is the previously read frame. The frames are first reduced to 2D matrices of pixel intensity values by converting the frame to black and white. Different methods are then applied to compare the frames and estimate the movement of pixel intensities, ultimately computing the velocity based on the time difference between frames. In the search for a suitable algorithm for interpreting velocity from surface flow in sewers, three algorithms that use different techniques for estimating movement are attempted.

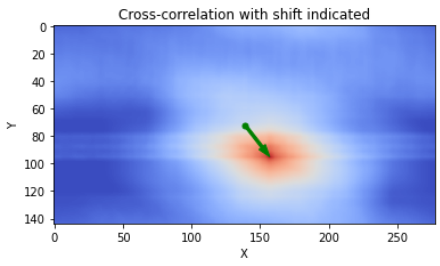
Particle Image Velocimetry

Particle Image Velocimetry (PIV) uses an interrogation scheme by dividing frames into smaller windows and estimates the displacement by finding the content from the interrogation windows in the present frame in the search windows in the reference frame. Several approaches exist, but one way of doing this is by computing cross correlation between the windows in each frame. This can be done through direct cross correlation between pixel intensities in the spatial domain, the cross correlation between the image signal transformed by discrete Fourier transformation, as described by Raffel et al. (2007), or by first decomposing the image into its abstract features.

Cross correlation between pixel intensities is used to determine the similarity between the images at different displacement values, with the peak in the cross correlation function representing the displacement that produces the highest similarity as shown in Fig. 3.4.



(a) Normalized cross correlation 3D plot between images.



(b) Cross correlation 2D intensity plot showing displacement.

Figure 3.4: Cross correlation between pixel intensities is used to determine the similarity between the images at different displacement values, with the peak in the cross correlation function representing the displacement that produces the highest similarity in intensity. Here it is computed between an image and the same image shifted (to the right and down). The cross correlation shows the highest similarity when the second image is shifted back (up and to the left) by the same distance as indicated by the green arrow.

By Fourier transforming the image "signal" into the frequency domain, the vari-

ation between frames can be interpreted based on the changes in pixel intensity along a distance in the image. As described in "Particle image velocimetry: a practical guide" by Raffel et al. (2007, p. 121), this takes advantage of the correlation theorem, which states that the cross correlation of two functions is equivalent to a complex conjugate multiplication of their Fourier transforms. Correlation analysis can then be done more efficiently in the frequency domain by simply multiplying the Fourier transforms of the signals and then transforming the result back to the time domain when needed using the inverse Fourier transform. This Fourier transformed image can be described as the spatial frequency of the intensity. If pixel intensities are slowly varying, such as in smooth areas, the frequency is low. Higher spatial frequencies correspond to fast-changing pixel intensities, such as the edges of particles. By interpreting the image as spatial frequency, the image content can be represented by the magnitude and phases of the complex Fourier coefficients, which describe the contribution of each frequency component in the image. Displacement can then be deduced from the phase shifts of different frequencies in the image as intensities move from frame to frame. Treating the image in the frequency domain also makes it easier to avoid noise by discarding what is assumed to be frequencies related to noise in determining displacement.

"Multipass" and "window deformation" implementations of PIV can further enhance the quality by shifting the search window towards the cross correlation peak in several iterations while also deforming it to recompute a new cross correlation in attempts to test to see if a better fit can be found for another slightly different displacement and shape of the interrogation window. As described by Boutelier et al. (2019) citing Scarano (2001) the initial vector grid is interpolated onto the new smaller grid in the next iteration and the search window is shifted and deformed before the next cross correlation computation to correct the vectors found in the first pass. This also allows to capture translation and rotation of particle clusters to better grasp the "real" movement of the particles.

The PIV algorithm implemented for processing videos in the project makes use of the "simple_multipass" function from the Python library OpenPIV (Liberzon et al., 2020). It applies multiple passes with iteratively smaller windows and overlaps between windows to investigate cross correlation between the Fourier transformed images. It also incorporates the deformation of the search window and utilizes subpixel interpolation by interpolating between correlation peaks found within the windows. This approach enables subpixel accuracy in determining displacement, as opposed to pixel accuracy obtained by considering only the largest cross correlation peak.

Optical flow

Optical Flow (OF), as described and presented in the paper on "Determining Optical Flow" by Horn and Schunck (1981), represents the distribution of movement within the image by tracking brightness patterns between consecutive frames in the image sequence. The main assumption in OF algorithms is that the brightness

of pixels related to objects remains constant over time, which is used to justify the computation of motion from pixel displacement between frames. Errors are introduced if the brightness consistency assumption is invalid for the processed footage. Such a case is illustrated in Fig. 3.5.

Optical flow algorithms make use of different iterative mathematical schemes to solve sets of equations that describe the motion in the image, as further elaborated in the article by Horn and Schunck (1981). These algorithms can either estimate the motion for all pixels in dense optical flow or a subset of pixels in sparse optical flow. The estimation is based on the requirement of smoothness in velocity flow, meaning that neighbouring pixels have similar motion, and image brightness consistency between corresponding pixels in two consecutive frames, as the motion and change in intensity are assumed to be small.

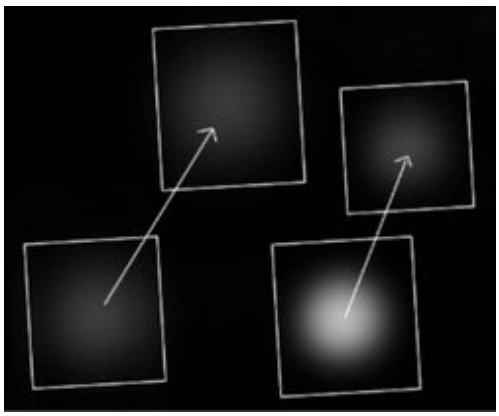


Figure 3.5: The assumption that OF relies on, of constant pixel brightness, only holds true for the displacement to the left. Estimating displacement for a moving object to the right using OF introduces errors in the velocity measurement.

The OF algorithm used to process the videos was provided by Prof. Francois Clemens. It incorporates the Farneback optical flow algorithm to compute dense optical flow between consecutive frames of a video. The Farneback algorithm, implemented using the "cv.calcOpticalFlowFarneback" function from OpenCV (OpenCV manual, 2022e), calculates dense optical flow by dividing the frames into smaller regions and searching for corresponding regions in the second frame through the analysis of pixel intensity patterns. By using polynomial expansion, the algorithm models the motion between frames by approximating pixel intensities in terms of a polynomial function. By solving a system of equations, it determines the polynomial coefficients, which enable it to calculate motion vectors for each region. These motion vectors represent the displacement of each pixel from the first frame to the second frame, resulting in a dense optical flow field. The previously computed flow field is also used as an initial guess for the new flow field of the next frame, introducing a temporal aspect to the algorithm by guiding it in the direction of flow coherency between frames.

Particle tracking velocimetry

Particle Tracking Velocimetry (PTV) involves tracking the motion of individual particles between successive images and computing the displacement between their positions. PTV can potentially provide a more detailed and accurate representation of the velocity field compared to other OV algorithms by tracking the same particles over multiple frames. Various methods, as described in the book "Particle Tracking Velocimetry" by Dabiri and Pecora (2019, ch. 5), can be used to achieve this. One common method is the multi-frame approach, which utilizes multiple image pairs to estimate the velocity field. The main challenge in this approach is to ensure that the displacement vectors from each image pair are consistent and can be combined to estimate the velocity field over the entire time interval. Another technique for particle matching in two-frame PTV analysis is the cross correlation method, which exploits the principle that fluid particles and their closest neighbouring particles typically undergo quasi rigid-body motion, meaning they move as a whole in translation and rotation. This knowledge is used to determine the probability of a candidate particle in the first frame being a match with a candidate particle in the second frame.

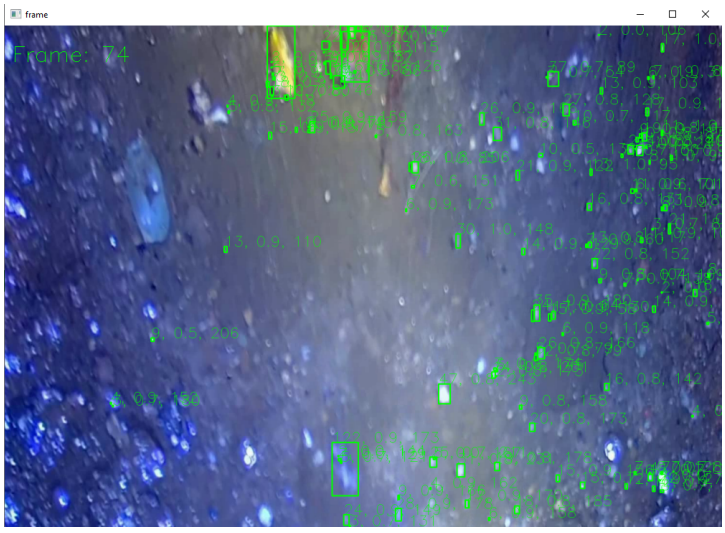


Figure 3.6: Screenshot from the particle identification procedure in the PTV algorithm in footage in ultraviolet/visible light. Data on the particles are displayed with the particle outline. Current frame particles are compared with particles detected in previous frames to obtain track their trajectory.

The success of the PTV algorithm depends on its ability to detect particles that accurately represent flow velocity and to distinguish between different particles to avoid producing erroneous displacement vectors. This involves two steps, as also described in the book by Dabiri and Pecora (2019, ch. 3). Firstly, the algorithm must determine which pixels constitute an image of a particle, and secondly, it needs to accurately locate the particle image in the present frame. The book

also discusses various particle masking techniques and particle identification algorithms. Viable options include neural network segmentation approaches, feature extraction from optical flow, and methods to enhance particle visibility, such as image binarization.

The PTV algorithm used for processing videos in this project, provided by Ph.D. Antonio Moreno Rodenas, follows a specific workflow. First, each frame of the video is processed to identify moving particles using background subtraction and contour-finding algorithms from OpenCV (OpenCV manual, 2022d; OpenCV manual, 2022g). This is depicted in Fig. 3.6. For each detected particle, properties such as centroid, size, and eccentricity are computed and stored. To link particles across frames and ensure the uniqueness of particle IDs, the code utilizes a combination of spatial and property distances. The spatial distance measures the physical distance between the centroids of particles, while the property distance compares the similarity of particle properties. By comparing the current particle with the particles detected in previous frames within a specified memory range, potential matches are identified based on both distances. If a match is found, the code assigns the ID of the matched particle to the current particle.

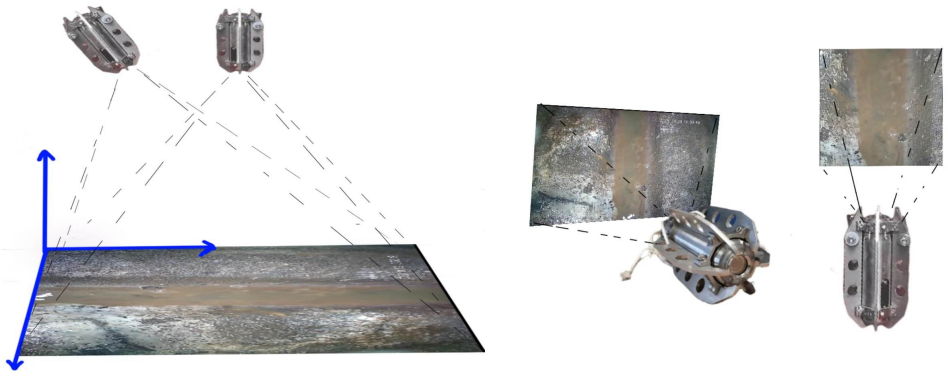
To refine the results, the code filters out particles that are tracked spuriously by removing tracks with fewer consecutive detections than a specified threshold. This filtering process helps eliminate unreliable or noise-induced particle trajectories. Finally, the velocities are calculated by measuring the displacement of particles over time. The code determines the change in position between the first and last positions of a particle track and divides it by the time difference to obtain the velocity.

3.1.3 Image and Video Processing

When projecting the 3D world onto the 2D image plane of a camera, a transformation from a three-dimensional coordinate system to a two-dimensional coordinate system occurs. It is important to consider this transformation when performing physical measurements in the images, as it can introduce systematic measurement errors. Since the camera captures only two dimensions, it is necessary to align the camera's image plane coordinate system with the plane in which the physical displacement occurs. Failure to do so will result in an out-of-plane component in the displacement that is not accounted for in the measurements.

Ideally, the camera would be positioned perpendicular to the water surface. However, in practice, this may not always be possible due to factors such as light reflections or the restrictive geometry of the manhole. Fig 3.7 illustrates the effect of having the camera at an angle towards the water surface.

The camera lens may introduce errors due to radial and tangential lens distortions, which cause non-linear displacements of image points. These distortions can affect the measured velocities, particularly near the image edges. To compensate for these lens induced deformations, undistortion techniques such as radial and tangential distortion correction can be applied to rectify the captured frames. To



(a) **(b)**

Figure 3.7: When the leftmost camera is at an angle towards the flow, the flow motion occurs in a plane that the camera is unable to accurately represent. As a result, displacement further from the camera is interpreted as larger, and parallel lines in the real world, such as the conduit embankment edge, appear to converge to a point at infinity in the image.

address the issues related to perspective imaging and ensure more accurate spatial measurements, perspective transformation can be applied to rectify the video frames captured at an angle. This transformation aligns the coordinate systems, allowing for a better interpretation/quantification of displacement in the images.

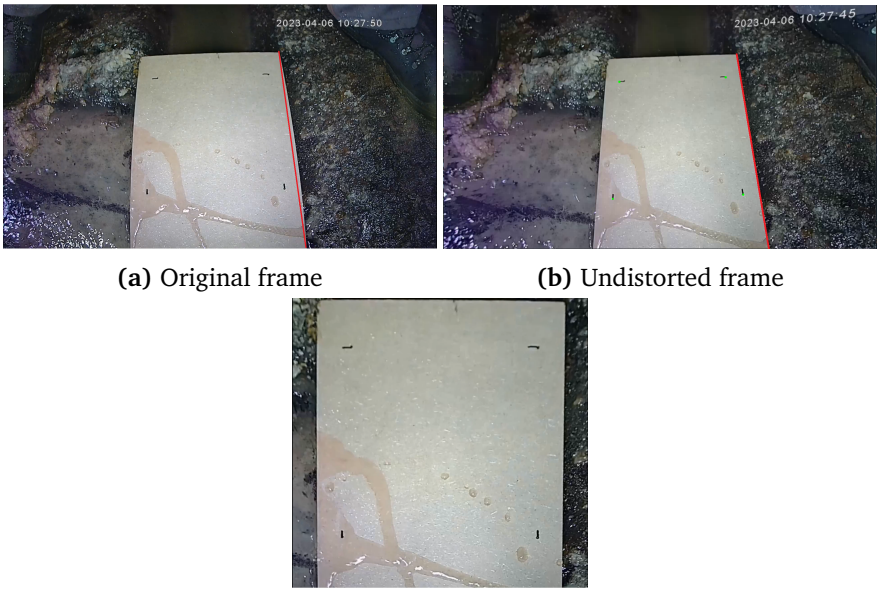
The transformation that occurs when the 3D world is captured in the 2D image plane can be mathematically described by a camera matrix, as explained in "Multiple view geometry in computer vision" by Hartley and Zisserman (2004, ch. 2). The camera matrix is a 3x4 matrix that relates the 3D world coordinates to the 2D image coordinates and contains information about the camera's intrinsic and extrinsic parameters. The intrinsic parameters characterize the internal properties of the camera, such as the focal length, principal point, and distortion coefficients. These parameters influence how the camera lens captures and projects the light rays onto the image sensor, thus affecting the projection of the 3D world onto the 2D image. The extrinsic parameters determine the position and orientation of the camera in the world coordinate system, affecting the viewpoint and perspective from which the scene is captured.

A calibration procedure has to be performed to estimate the camera's intrinsic parameters. This involves mapping an object with a known geometry (such as a checkerboard pattern or matrix of dots) to corresponding points in the image plane while minimizing the reprojection error, as described in the OpenCV manual (2022b). By using the found parameters to transpose the image the effects of lens distortion are minimised. The impact of lens distortion correction can be observed in Fig. 3.8a and 3.8b, where the edge of the wooden plate straightens and aligns with the red straight line after distortion correction.

When capturing images under perspective imaging, the shape of objects be-

comes distorted. However, this distortion can be corrected by computing an inverse transformation and applying it to the image. As also explained by Hartley and Zisserman (2004, p. 34), by mapping a real-world quadrilateral with known dimensions, which appears distorted in the image, to four corresponding points in the image plane, the transformation matrix can be estimated. This matrix can then be used to undistort the object and restore its correct geometric shape in the image. Fig. 3.8c shows an example of a calibration board used for the purpose and the transformed frame.

When performing perspective transformation and undistortion, new pixels are introduced and interpolated to fill gaps as part of the image is stretched or distorted. This has the potential to increase the measurement error as this interpolation process introduces an element of approximation. The accuracy of the measurements relies on the assumption that the interpolated pixels accurately represent the true values that would have been captured by the camera.



(a) Original frame **(b)** Undistorted frame

(c) Transformed frame

Figure 3.8: The procedure of obtaining a top-down view from the original footage by first correcting for lens distortion and next correcting for perspective distortion. In the latter step parts of the image are lost. This can be counteracted but at the cost of a loss of resolution.

Depending on the characteristics and visibility of particles within the camera's field of view, as well as the presence of any disturbing elements that affect velocity estimation, such as reflections, it may be necessary to pre-process the videos using various techniques to modify pixel intensities. Fig. 3.9 and 3.10 illustrate different pre-processing techniques used.

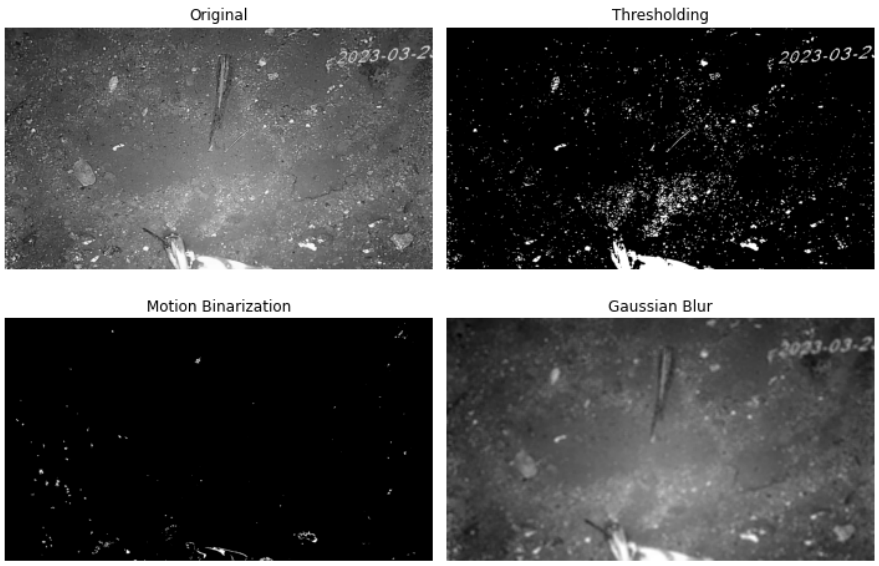


Figure 3.9: Thresholding involves assigning pixel intensities as black or white based on a predefined threshold value. Motion binarization sets pixel intensities as black or white depending on the difference between frames, emphasizing areas of movement. Gaussian blurring reduces the noise and sharpness of edges.

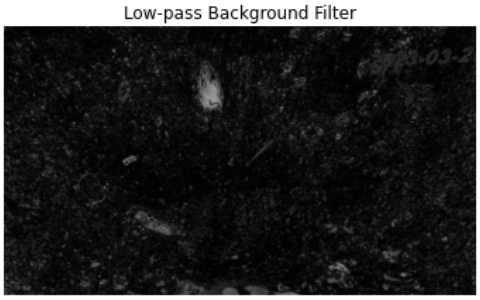


Figure 3.10: Low-pass background filtering uses a filter on all image frames to reduce high-frequency noise. Subtracting the filtered image from the original isolates high-frequency components not part of the background in the overall sequence.

The pre-processing of images and other video-related tasks are performed using the OpenCV library for Python, which is an open-source computer vision and machine learning software library. OpenCV provides a wide range of algorithms and functions for various image and video processing tasks, including checkerboard calibration, undistortion, estimation of projective matrices, perspective transformation, image thresholding, low-pass filter subtraction, Gaussian filtering, and the ability to read and write frames to and from videos. These functionalities are described in the online OpenCV-Python tutorial/manual documented in OpenCV manual (2022a), OpenCV manual (2022c), OpenCV manual (2022f),

OpenCV manual (2022h), OpenCV manual (2022i) and OpenCV manual (2022j). A general introduction to the library and its capabilities is given in the book "Learning OpenCV: Computer Vision with the OpenCV Library" by Bradski (2008).

3.2 Optical Velocimetry Experiments

Experiments were conducted to test two hypotheses. The first hypothesis aimed to determine if one algorithm would be superior to the other algorithms in terms of lower variability and detection limit. The second hypothesis explored whether capturing videos in a combination of visible and ultraviolet light is more beneficial for Optical Velocimetry (OV) purposes compared to using visible light only. It is mainly based on the potential of there being organic compounds present in the water that are fluorescent, meaning that they are able to absorb the radiated energy from UV, and emit visible light, making them potentially more distinguishable in the footage. The fluorescence characteristics could be present due to numerous types of fluorescent substances, including humic and fulvic acids, lignin-derived substances, as well as varying amounts of steroids, phenols, nonvolatile acids, oils, and trace quantities of surface-active agents according to Baker et al. (2014, p. 102).

The experiments were conducted by positioning a camera in a section of the sewer system where there otherwise was minimal flow present to capture video while the flow of water passing the camera was controlled by pouring water into an upstream manhole from a tank. The experimental setup considered different combinations of algorithms, lighting conditions (including ultraviolet and visible light - UV/VIS or visible light only - VIS), and various flow scenarios (both with water flowing and without flow). The experimental matrix is presented in Table 3.1.

	VIS	UV/VIS	PTV	OF	PIV	FLOW	NO FLOW
Run 1		x	x	x	x		x
Run 2		x	x	x	x	x	
Run 3	x		x	x	x		x
Run 4	x		x	x	x	x	

Table 3.1: Experimental matrix

The focus of analysis in the "no flow" condition was to evaluate the algorithms' performance in the absence of any actual flow. This assessment aimed to determine the level of measurement noise present and establish the algorithms' threshold for detecting movement. If the algorithms exhibit a tendency to produce non-zero velocities in the absence of flow, it becomes impossible to differentiate actual flow velocities below this threshold from the measurement noise.

With flow present in the footage, the algorithms were evaluated by examining the standard uncertainty associated with the velocities they produced. This

assessment was performed by calculating the standard deviation of the velocities produced for each frame after filtering those that deviated significantly in direction or magnitude. Additionally, the algorithms' performance was visually evaluated by plotting the displacement vectors and comparing them with the observed flow motion in the videos. The processing time of the algorithms was also noted, as computational effort is an important parameter in practical applications. The algorithms were given the best chance at performing well, with beneficial preprocessing of the videos and adjusting algorithm parameters to suit the wastewater characteristics in the footage. While some standard deviation is expected due to local variations in flow velocities captured by the camera, comparing the standard deviations among the algorithms helps to identify the algorithms that yield the widest range of velocities for the same footage. Higher standard deviations may indicate over- or underestimation of velocities by certain algorithms.

3.2.1 Real Accuracy Of The Measurements

The primary focus of the method of using OV to detect illicit inflow is to observe the relative change in velocity rather than providing an absolute measurement of the real flow velocity. However, having an estimate of the real flow velocity can be useful for adjusting algorithm parameters and optimizing the filtering process to obtain velocity values that best represent the actual flow velocity at each frame.

To obtain an estimate of the real flow velocity, the same videos processed in the experiment can be analysed using video tracking software such as the open-source Tracker software. This software enables tracking and motion analysis of objects in videos, providing an estimation of velocity. It is important to note that the velocity estimate obtained from video tracking software, similar to the optical vision algorithms, is subject to uncertainties associated with performing real-world physical measurements from digital images. Nevertheless, this estimation can serve as a reference for calibrating and refining the optical velocimetry algorithms to enhance their accuracy and reliability.

3.3 Reducing Uncertainty

Bigger variability in the estimated velocities makes it more difficult to detect deviations to reveal illicit inflow, as it becomes challenging to discern whether a particular estimated velocity represents a genuine deviation or simply reflects the inherent variability in the measurements. This variability influences the detection sensitivity, which refers to how well the algorithms are able to distinguish between different flow velocities. The variability in frame velocities could stem from actual differences in velocities within the image frame, as well as from the variability in the algorithm's ability to estimate these velocities and definite wrong interpretation by the algorithm resulting in spurious vectors. In the implementation of the optical velocimetry method of detecting illicit inflow, the frame velocity is averaged to produce measurement points (video sequence velocity) in the time

series which reduces the impact of uncertainty in the frame velocity towards the sequence velocity, but it is nevertheless important to reduce it.

The measured velocity is determined by the estimated displacement, the time interval between frames, and the uncertainties associated with these quantities. The uncertainty arises from the uncertainties in the measured positions (initial and final positions, x_1 and x_2) and the uncertainties in the measured times (initial and final times, t_1 and t_2) used to calculate the velocity. According to the law of propagation of uncertainty, as described by Ku (1966), the overall uncertainty of the velocity calculation can be obtained by combining the uncertainties of the involved variables. If we disregard the possibility of covariance between variables, eq. (3.1) represents the resulting uncertainty.

$$\sigma_v^2 = \left\{\frac{\partial v}{\partial t_2}\right\}^2 \sigma_{t_2}^2 + \left\{\frac{\partial v}{\partial t_1}\right\}^2 \sigma_{t_2}^2 + \left\{\frac{\partial v}{\partial x_2}\right\}^2 \sigma_{x_2}^2 + \left\{\frac{\partial v}{\partial x_1}\right\}^2 \sigma_{x_1}^2 \tag{3.1}$$

Since the standard uncertainty is the same for both the initial and final times, the time variables contribute to the combined uncertainty as $2\frac{\Delta x}{\Delta t^2}\sigma_t$. This reveals that evaluating velocity at larger time intervals (Δt) reduces the uncertainty in the measurements, as the denominator increases. This can be achieved by skipping frames in the video. As described in the article by Feng et al. (2011), the uncertainty in particle position is due to random image noise and the finite number of pixels used to represent a particle. This uncertainty remains constant at each evaluation and has a larger relative contribution when evaluated more frequently. Similarly, uncertainties related to the time variable, such as variable frame rates, shutter speed, and image processing are expected to be reduced when evaluated less frequently, as the outcomes of repeated processes are averaged. However, it's important to note that reducing the uncertainty by using longer time intervals may be counteracted by increased uncertainty in the actual displacement, as more time between frames allows for changes in particle trajectory and acceleration to occur. The overall effect can be assessed by examining how the standard uncertainty of velocities retrieved from each frame changes when more frames are skipped.

An attempt to estimate the theoretical uncertainty in the measurements can be made by calculating the standard uncertainty based on certain assumptions with the law of propagation of uncertainty. By comparing the calculated uncertainty with the measured uncertainty, we can assess whether our experimental uncertainties are consistent with the theoretical predictions and evaluate the performance of the algorithms. If the measured uncertainty is larger than the calculated uncertainty, it could indicate the presence of error sources in our experimental setup that are not accounted for, or that the assumptions made in our calculations are not valid. Conversely, if the measured uncertainty is smaller than the calculated uncertainty, it could indicate that we have overestimated the uncertainty in our measurements or that our experimental techniques are more precise than initially thought. However, it's important to note that the result will only be an approximate estimate, and the accuracy will depend on the quality of the estimates

of the uncertainties in the variables. When the covariance terms are unknown, we may assume the variables to be uncorrelated or very small compared to the other terms. However, this assumption may not be valid and can lead to an underestimation or overestimation of the uncertainty in the calculated values compared to the measured values.

3.4 Inflow Detection Using Significance Testing

Ideally, a large number of measurements would be conducted during dry weather to establish a baseline of flow values for each discrete moment throughout the day (hydraulic fingerprint). Measurements outside this range could indicate the presence of illicit inflow, but here it is also important to consider that unlikely events can occur even during dry weather conditions. However, if a series of outliers is measured consistently over time, it could indicate 1) the presence of illicit inflow, 2) faulty measurements, or 3) changes in the system's geometry due to maintenance activities, or 4) an unknown facility has started discharging intermittently, this may e.g. be the result of some illegal activity. Each possibility can be evaluated by cross correlation with other information sources, such as getting reports from the system operators, regular checks of the sensor status and generally being aware and documenting any nuisance. Further discussion is limited to the former possibility. The possibility of there being faulty measurements needs to be verified under all conditions.

In situations where the number of measurements is considerably smaller, such as when the camera is deployed for a few days to a couple of weeks, it becomes more challenging to draw definitive conclusions about the parent population to get the hydraulic fingerprint as the samples may not be fully representative. Statistical significance testing will then state with some level of confidence based on the number of samples whether the measured difference in flow is statistically significant and represents a real difference in flow, or if it could be due to measurement uncertainty or chance occurrences.

A common approach, as outlined by Løvås (1999), is to use a t-test to compare quantitative data between two groups. In this case, a paired t-test can be employed to assess whether the mean difference between paired measurements taken before and after an intervention (in this case, storm events) is significantly different from zero. The paired t-test assumes that the differences between the pairs are normally distributed and that the observation pairs are stochastically independent. By evaluating the test statistics obtained from the paired t-tests for all possible combinations of "dry" and "wet" data, it is possible to determine if they significantly differ from each other.

The paired t-test as described by Løvås (1999) is performed by first defining a null and alternative hypothesis with the null hypothesis being that there is no significant difference between the means of the two paired samples, while the alternative hypothesis assumes that there is a significant difference. The data is collected in pairs, in this case, related by being conducted at the same time during

the day measured in both "dry" and "wet" weather. The first condition's measurements are denoted as $X_1, X_2, \ldots, X_n$, and the second condition's measurements as $Y_1, Y_2, \ldots, Y_n$, where n is the number of paired observations. The differences between the paired observations are found by subtracting the first condition's measurement from the second condition's measurement for each pair: $D_i = Y_i - X_i$, where D_i is the i-th difference. The sample mean of the differences denoted as $\bar{D}$, is then calculated using eq. (3.2), and the standard deviation of the differences from eq. (3.3). The test statistics, denoted as t, using the formula in eq. (3.4). To evaluate the test results, the test statistic is compared to the critical value determined based on the chosen significance level. If the absolute value of the test statistic is greater than the critical value, the null hypothesis is rejected, indicating a significant difference between the means of the paired samples.

$$\bar{D} = \frac{1}{n} \sum_{i=1}^{n} D_i \tag{3.2}$$

$$S_D = \sqrt{\frac{1}{n-1} \sum_{i=1}^{n} (D_i - \bar{D})^2} \tag{3.3}$$

$$t = \frac{\bar{D}}{S_D / \sqrt{n}} \tag{3.4}$$

3.5 Sensor strategy

The sensor strategy comprises three primary components: identifying suitable sensor locations, verifying the results through rain-inflow event simulations, and attempting to recreate measured inflow scenarios to locate the source. All hydraulic simulations for this purpose were performed using the SWMM software, with the assistance of Python packages PySWMM (McDonnell et al., 2020) and swmmio (Erispaha, 2020) for simulation, manipulation, and result retrieval.

Currently, the best practice for locating illicit connections involves preliminary investigations followed by the use of Distributed Temperature Sensing (DTS) if the preliminary investigations do not yield definitive results or if there are indications of inflow. The preliminary investigations may include system cleaning and CCTV inspections to identify suspicious debris or signs, or the placement of a sieving mechanism at the outflow to observe any accumulation of debris caused by illicit connections. Based on the evidence gathered during the preliminary investigations, a decision is made to either deem the investigated area healthy or to proceed with further inspections. However, conducting effective preliminary investigations in sewer systems, particularly for detecting stormwater connections, remains a challenge as there are few specific visual indicators distinguishing stormwater from sewer water.

Although one part of the sensor strategy promises to locate the source of inflow (further discussed in subsection 3.5.3), limitations in accurately modeling the system and uncertainties related to runoff generating processes may pose challenges

to this approach. The method of using camera sensors viability as a preliminary investigation technique remains, even if it cannot precisely pinpoint the exact location.

DTS must be applied throughout the system to ensure reliability because the temperature change observed by the fiber optic cable diminishes with increasing distance from the source of illicit inflow, as the inflow becomes diluted with the existing water. To achieve reliable results, the cable must remain in place for a duration sufficient to capture any potential illicit connections. In the case of illicit connections to sewer systems, this involves waiting for rain events and relying on the inflow temperature being distinguishable, considering the influence of air and surface temperatures. A waiting period (about two weeks) is typically required until it can be confidently stated that any illicit inflow should have already been observed. A more sophisticated method, although complex in the case of DTS, would involve simulating the possible inflows starting from the first rain event after sensor placement to determine if the event would have been detectable as a temperature increases in the case of illicit connections. This approach could potentially reduce the measurement campaign duration from weeks to days. Similarly, the optical velocimetry method could adopt a similar approach, which would be less complex to model, to both shorten the measurement campaign and provide more reliable and concrete results in terms of reliability. Several camera sensors can be mounted at different locations throughout the system to quickly assess the presence of illicit inflows.

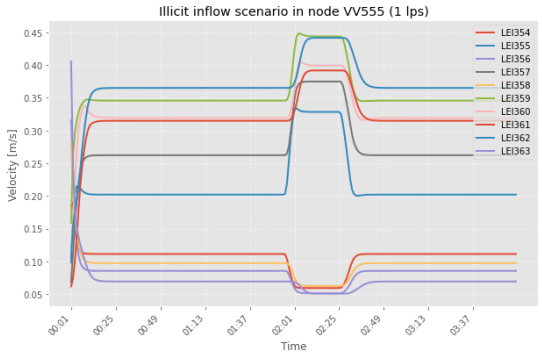
The decision on camera sensor locations needs to take into account the hydraulic considerations related to observing inflow with the camera and also practical considerations related to using the sensors in conjunction with DTS. For that purpose, it is proposed to use a genetic algorithm (GA) to solve what becomes a multi-objective optimization problem.

3.5.1 Sensor Placement

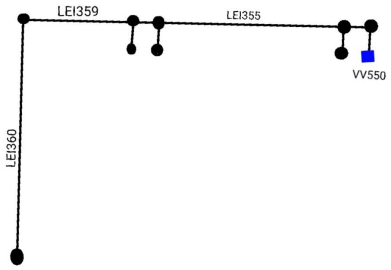
By first assessing the sensitivity of different sensor locations towards detecting illicit inflow through hydraulic simulation of various inflow scenarios, the most suitable locations for sensor placement are revealed. Simulating the full range of possible inflow scenarios also provides an impression of how well the camera sensor strategy can cover the system since the simulations could reveal inflow scenarios that may never be detected by any sensor placement.

The hydraulic model of the sewer system consists of model nodes (manholes) and links between the nodes (conduits). In the illicit inflow scenario simulation, additional inflow is introduced in each node one at a time on top of a baseline flow scenario with the same low inflow in all nodes of the system. The added inflow resembles what could be seen as a realistic case of a rain event combined with a wrongly connected roof drain with a catchment (roof area) based on the housing demographics in the area. Each potential sensor location is then investigated by examining the increase in simulated velocity in the conduits for the different scen-

arios. Depending on the uncertainty in the estimation of the velocity by OV there must be a certain increase in velocity for it to reliably detect as a difference and not just random fluctuations in measurements. A detection limit of 10% is here used as the minimum increase in velocity for the inflow scenario to be considered detected, which was the maximum error found in PTV algorithms (from a reference velocity) by Jeanbourquin et al. (2011) when comparing PTV algorithms accuracy. The results from one inflow scenario from an arbitrarily selected node ("VV550") are presented as a time series showing the temporary increase in velocity for some selected conduits ("LEI" = conduit), along with a drawing of node "VV550" and parts of the system close to it in Fig. 3.11. The conduits in the main axis observe an increase in velocity, while the velocity in smaller connecting pipes is being slowed down.



(a) Increase in velocity in conduits ("LEI") in the system with illicit inflow at node "VV550".



(b) Node "VV550" and nearest conduits.

Figure 3.11: Time series of velocities observed in different conduits for one of the simulated inflow scenarios. After simulating all possible inflow scenarios each conduit has an associated list of nodes (parts of the system) from which it is able to detect inflow from.

When all scenarios have been simulated, each conduit will have an associated list of nodes from which it is able to detect inflow. By creating the system as a directed network using the Python package NetworkX (NetworkX documentation, 2023), which allows for manipulation and analysis of complex networks, data such as lists of upstream conduits and manholes, and lengths of the upstream system can be easily retrieved. It also enables solving network optimization problems such as Shortest Path Problems (SPP). By comparing the list of upstream manholes with the list of scenario detections made by the conduit, a detection likelihood can be computed, serving as a measure of their ability to detect all expected scenarios. A detection likelihood score of one indicates that the conduit detects all upstream inflow scenarios. Having a likelihood lower than one should not necessarily exclude the manhole from being a potential sensor location. However, placements at these locations will require additional investigations of areas where the camera is unable to detect inflow.

When each conduit has its list of nodes from which it can detect inflow, it becomes interesting to determine which combinations of sensor locations will cover most of the network. However, this cannot be assessed on its own as DTS might need to be implemented after the camera sensor placement to further narrow down the location. DTS cables are fixed in lengths and need to be laid in loops to cover dead-ends in the system. They also require connection to a processing computer on one end, necessitating appropriate infrastructure placement. The fixed and limited cable length necessitates optimization to utilize most of the cable in each campaign, efficiently covering the entire system and avoiding leaving small portions uninvestigated. Therefore, employing a GA approach to explore numerous solutions and converge towards the best one can be beneficial.

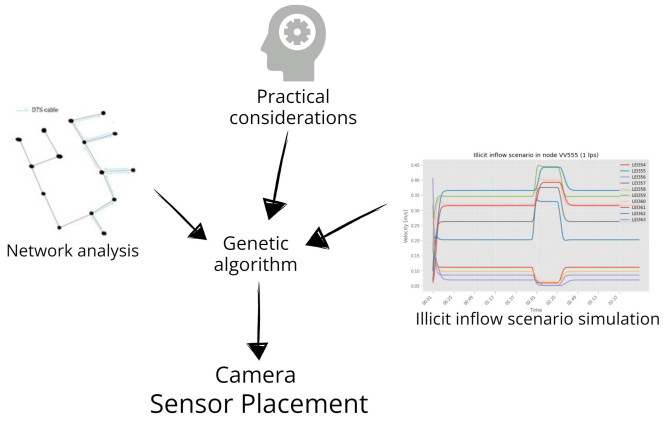


Figure 3.12: The genetic algorithm converges towards the best solution taking into account the practical considerations regarding placement of the DTS cable, network analysis and the results from hydraulic simulations.

Genetic algorithm

The concept of genetic algorithms (GA) and the non-dominated sorting genetic algorithm (NSGA), as described in the specialization project by Løfald (2022), is reproduced here to justify its use for sensor location optimization. The GA simulates the process of genetic evolution, where the possible solutions to an optimization problem are represented as chromosomes (candidate solutions) composed of genes (nodes) representing the variables. Selection is made among the chromosomes for mating and recombination to produce a new generation of chromosomes, as described by Mishra et al. (2022, p. 11).

The non-dominated sorting genetic algorithm (NSGA), described by Deb et al. (2002), has been used as a method for optimization in sensor strategy-related problems, particularly in the context of the Battle of the Water Sensors Network (BWSN) reviewed by Ostfeld et al. (2008). In NSGA, solutions are compared to determine dominance. The domination count for each solution, represents the number of solutions that dominate it, and the set of solutions it dominates. Solutions

with a domination count of zero form the first front and the solutions they dominate have their domination counts reduced. This process continues to identify all fronts as new solutions have their domination count reduced to zero.

NSGA maintains solution diversity by estimating the density using the crowding distance. The crowding distance is the average distance between neighbouring points along all objective functions. It is calculated by normalizing the differences in function values using eq. (3.5) and summing the distances for each objective. Smaller distances indicate crowding by similar solutions. Solutions with lower non-domination ranks and less crowding are preferred when selecting and recombining for the next generation. Elitism is introduced in the algorithm by comparing previously found best solutions to the current population and retaining the best ones as described by Deb et al. (2002).

$$i_d = \frac{dist(i+1) - dist(i-1)}{obj(max) - obj(min)} \tag{3.5}$$

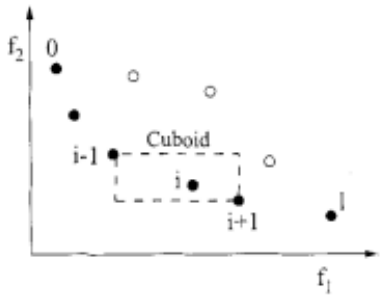


Figure 3.13: Crowding-distance calculation illustrated Deb et al. (2002).

Various criteria can be used to evaluate the "fitness" of the subset of sensor locations in the GA. Firstly, value should be given to the effectiveness of placement in terms of its ability to cover a significant portion of the network. This is achieved by optimizing for the length of the upstream network covered by the sensors while considering the limitation of the fixed length of the DTS cable. If the upstream area exceeds the coverage capacity of the DTS cable, a "penalty" could be applied. This evaluation utilizes features from NetworkX, along with additional code to handle issues typically not encountered in directed network problems, such as having a directed graph (flow-direction) that can be traversed in any direction in the case of DTS implementation.

The second criterion focuses on the detection capability of the camera sensors as the previously mentioned detection likelihood is used to score each manhole in the fitness function. The two first criteria could make the GA select subsets that have a high detection likelihood but make few detections or subsets that have a long upstream network without many nodes where illicit inflow could occur. The third criterion fulfils the purpose of the other two criteria by assigning value to the manhole based on its ability to detect inflow from a high number of upstream

nodes. This is illustrated in Fig. 3.14. A better measure of the possible "activity" in the upstream network, as nodes/manholes not necessarily are synonymous with connections with inflow, would be to have mapped the actual number and locations of connections to the system in order to use this in the simulation. A subjective evaluation of different placements could also be implemented by assigning a "quality" score to each manhole that considers the suitability of the locations.

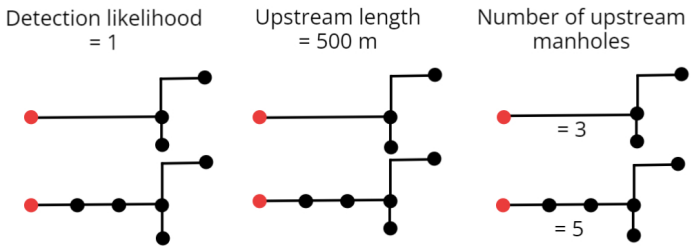


Figure 3.14: Evaluating the solutions based on the number of upstream nodes avoids making the GA select subsets that have a high detection likelihood but make few detections or subsets that have a long upstream network without many nodes where illicit inflow could occur.

3.5.2 Result Verification

Once the camera sensors and optical velocimetry have produced several velocity time series, a flow pattern can be established. This flow pattern can be used to model node inflow in the same node in the hydraulic model as where the camera was placed. By setting the measured dry weather flow as a baseline inflow in the model, different rain scenarios can be simulated to observe which rain events are necessary to see a change in velocity higher than the detection limit. The rain data collected throughout the measurement campaign can also be used in scenario simulations to determine if a storm event observed should have produced a significant increase if there was illicit inflow upstream from the camera placement during the campaign. How the measured velocity and rain data can be used in the verification of the results is shown in Fig. 3.15.

3.5.3 Illicit Inflow Source Detection

When a time series retrieved from the camera sensors and optical velocimetry have measured an illicit inflow as a significant deviation in velocity from the baseline this series becomes the illicit inflows fingerprint. The baseline flow pattern can then be used in conjunction with the observed rain event that caused the inflow to simulate scenarios of inflow at different nodes to achieve a similar fingerprint in the model as in the real measurement series. The nodes that produce the most similar fingerprints can then reveal information about the location of the source. However, there are limitations to this approach due to the difficulty in obtaining

an accurate model of the system and uncertainties and unknowns related to the rain-runoff process, such as the amount of rainfall that becomes runoff, the runoff storage capacity of roofs, and the travel time from rainfall in the most distant parts of the roof until it enters the system. Despite these uncertainties, there is a possibility of excluding areas in distant parts or in close proximity just simply from the time lag between the onset of rain and the measured increase in inflow. How the measured velocity and rain data can be used in source localization is shown in Fig. 3.15.

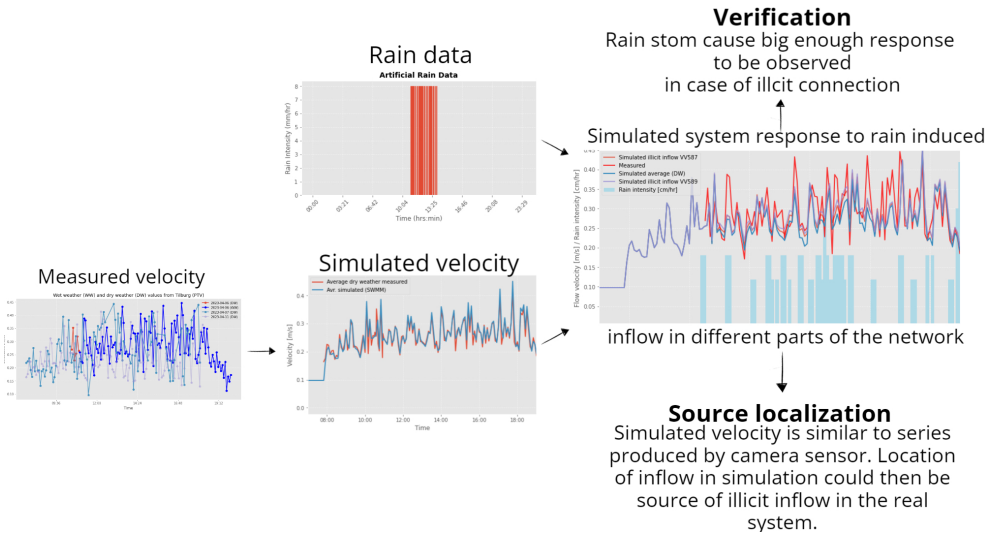


Figure 3.15: The measured velocities in dry weather is used to generate inflow in the hydraulic model to recreate the observed velocity series in the model. By using rain data and simulating inflow events it can be determined if the rain event would have produced a big enough velocity change from the simulated in case of an illicit inflow to be detected. There is also the possibility of locating the illicit inflow source if a simulated scenario produces a similar series as a real measured series that is known to be influenced by rain/inflow.

3.6 Field Study

Vandervalk+degroot, an inspection and renovation company in the Netherlands, has been assigned by the Tilburg Municipality to locate illicit connections in their stormwater and sewer systems. This task is primarily addressed through various strategies of preliminary investigation followed by Distributed Temperature Sensing (DTS). One of the areas suspected of having illicit connections is "Tradepark Noord", a highly industrialized area with a significant proportion of shops, logistics companies, and industries, all accompanied by extensive impervious surfaces. The project description agreed upon by V+G and Tilburg outlines a process that involves using cameras for a preliminary investigation to create a flow fingerprint of the sewer system. Rain data from a weather station is then correlated with

the flow patterns that deviate from the fingerprint, indicating areas that require further investigation with DTS.

3.6.1 Sewer System Model

The sewer model was created using data provided in the SUF-HYD format, which is a Dutch format for storing data on sewer models. The process of reading the format was automated in Python to generate an input file suitable for SWMM modeling. Some of the information in the file was found to be incorrect, and minor adjustments were made to ensure that the water flows in the intended direction toward the system outlets. The system under consideration is a sewer only network that transports wastewater from an industrialized area, primarily consisting of shops and warehouses, to a pumping station. The entire system ("Tradepark Noord") is drawn out in Fig. 3.16.

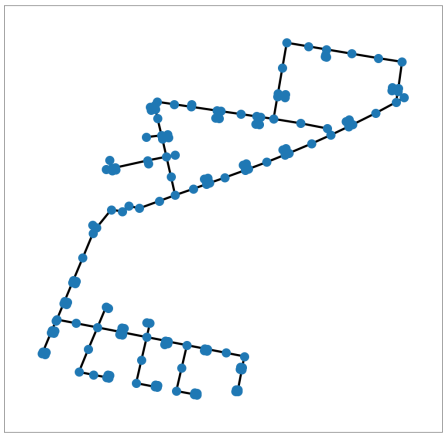


Figure 3.16: The sewer system ("Tradepark Noord") as NetworkX graph.

3.6.2 Camera

The cameras used in the field implementation were two modified versions of the C70 Sewer Jetting Video Nozzles provided by Sewer Robotics. These cameras allow for automated, well-lit, self levelling recording and storage until the footage is retrieved in a sewage-proof casing. Both cameras were similar in build, with the only physical difference being the light rings surrounding the lens. One camera had visible light Light Emitting Diodes (LED) only, while the other had a combination of visible and ultraviolet LED. There were also some differences in the software, as one camera allowed for automatic switching of the light while recording and went into a low-power state between recordings to save battery. These were modifications made to the camera for the purpose of stationary recording, and the cameras were considered prototypes under development. Due to a limitation in battery capacity, both cameras had to be used alternately. The unused camera

was charged and synchronized to retrieve the videos while the other camera was capturing footage.

The camera was mounted in the manhole by sliding it into a metal holster resting on two expandable horizontal bars spanning between the manhole walls, as depicted in Fig. 3.17. This setup allowed for hoisting the camera in and out of the same position when necessary for recharging or downloading videos. It reduced the time spent in sewers and ensured a standardized and consistent installation so that the perspective transformation only needed to take into account rotational differences (as the self levelling did not work perfectly) between measurement sequences when the camera was replaced. During the initial measurement campaign, a sheet with known distances between corner points was placed in front of the camera and then removed. This allowed for marking four points in a plane aligned with the water surface, which was used in the perspective transformation. It also serves the purpose of a calibration object for the conversion between distance in pixels in the image to real world distance to derive velocities in meters instead of pixels.

Figure 3.17: The camera was mounted in the manhole by sliding it into a metal holster resting on two expandable horizontal bars spanning between the manhole walls.

The video sequences captured by the camera were processed using all three algorithms. The PIV and PTV algorithms were applied to the original footage, while the footage for OF was prepared by performing motion binarization. To reduce uncertainty, frame skipping was used in the PIV and OF algorithms, with two frames skipped between each evaluation of the displacement with an initial frame

rate of 20 frames per second. No frame skipping was done for PTV, as processing all frames gives the algorithm a better chance of tracking particles over enough consecutive frames.

3.6.3 Weather Data

The Weather Observations Website (https://wow.metoffice.gov.uk/), a service provided by the UK Met Office that allows anyone to report and share weather observations, was used to retrieve rain data for evaluating results and hydraulic modeling purposes. Three different weather reporting sites were located within a 4 km radius of the camera placements, providing the opportunity to verify the quality of rainfall data by comparing data from neighbouring stations.

Chapter 4

Results

4.1 Optical Velocimetry Experiment

The optical velocimetry (OV) algorithms produce velocity vectors for each pair of evaluated frames in the experiment videos. These vectors can be visually inspected to investigate whether the algorithms accurately represent the flow in the footage. Representative frames from the processed videos from the experiment with flow present are presented in Fig. 4.6 and 4.8. Additionally, frames representing the typical noise in the videos from the "no flow" experiment are shown in Fig. 4.2 and 4.4. The pre-processing procedures used were Gaussian blurring (gb) and lowpass background filtering (lpbf) for PIV, thresholding (th) and motion binarization (mb) for OF, and no preprocessing for PTV.

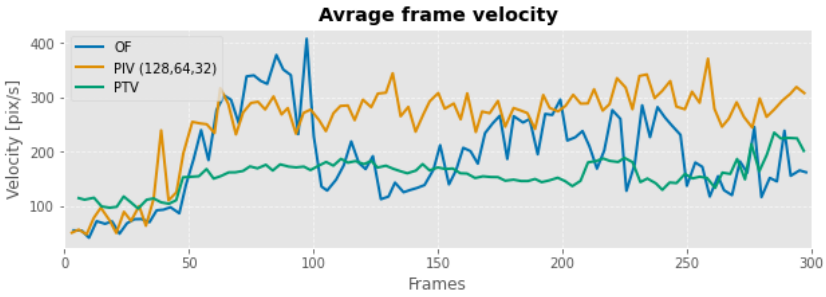
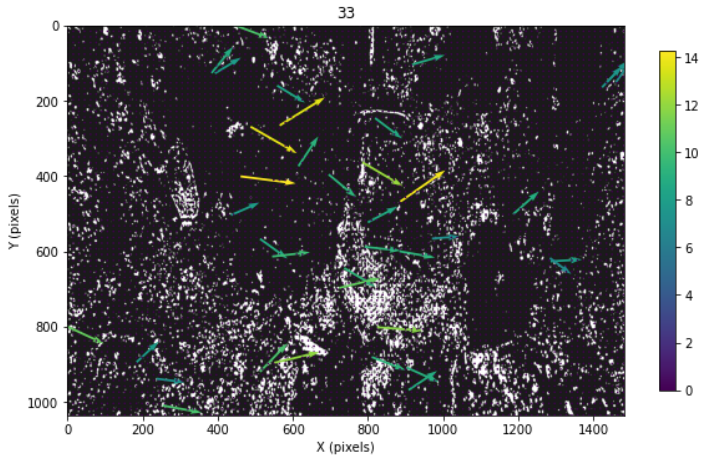


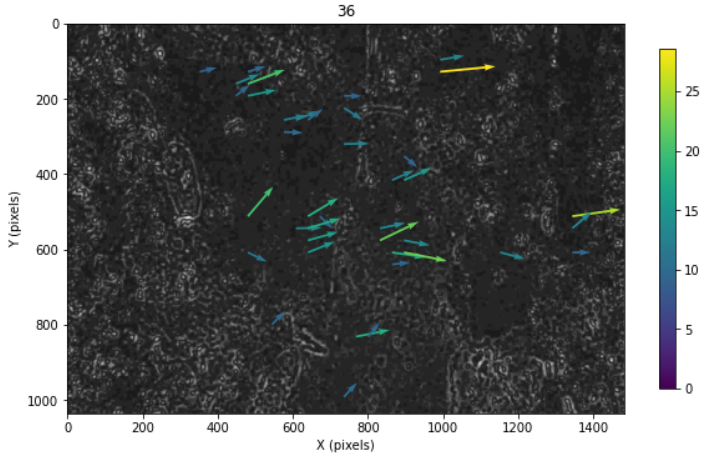
Figure 4.1: Average frame velocities produced by OV algorithms for ultraviolet/visible light when skipping two frames. The strange behaviour in the OF series is caused by its incapability of representing the velocity of a big passing particle. The following plots on the next pages are extracted from smaller parts of this full velocity series produces in the experiments.

Part of the velocity series from the experiments, when there was flow present, is presented in Fig. 4.7 and 4.9. It is extracted from the part of the full average frame velocity series presented in Fig. 4.1 after the slightly visible stabilization of the velocity, as the wave of water that was poured upstream reached the camera. Similar plots are presented in Fig. 4.3 and 4.5 for the "no flow" experiment. The three subplots show how the standard deviation of the produced velocities and the average frame velocity compare for the different algorithms when skipping zero, one, and two frames. The standard deviation of the frame velocity (for "flow") and average frame velocity ("for no flow") is computed from the remaining velocity vectors after filtering out velocities lower than 25% of the largest magnitude and vectors that deviate more than 0.5 standard deviations in any direction from the average direction.

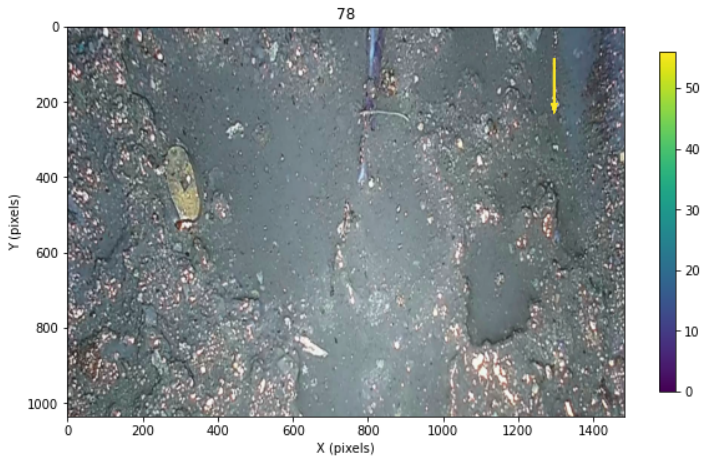
Visible light - NO FLOW



OF (th)



PIV (gb, lpbf)



PTV

Figure 4.2: Typical noise during "no flow" experiment in visible light. Pre-processing technique in parenthesis.

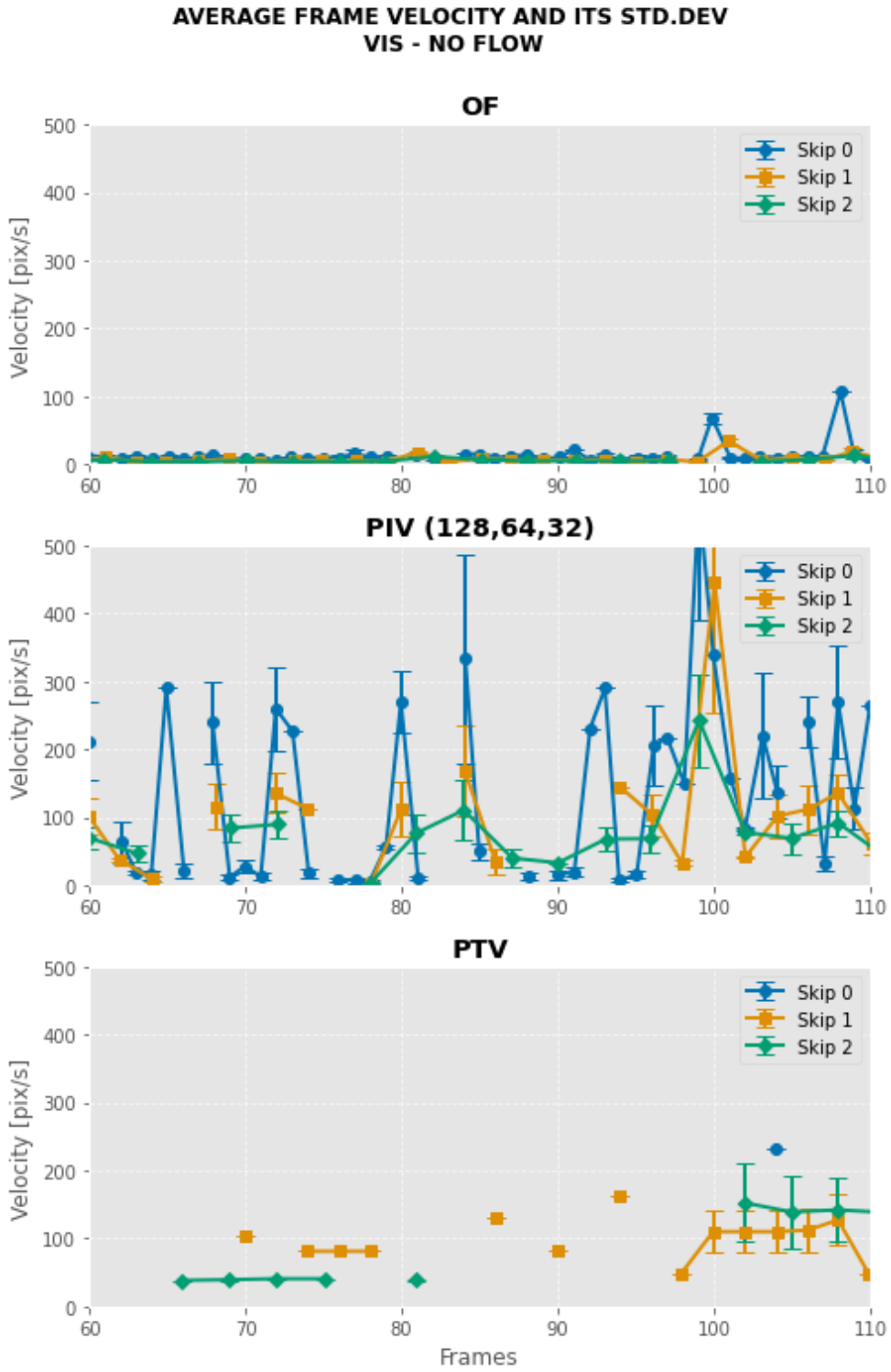
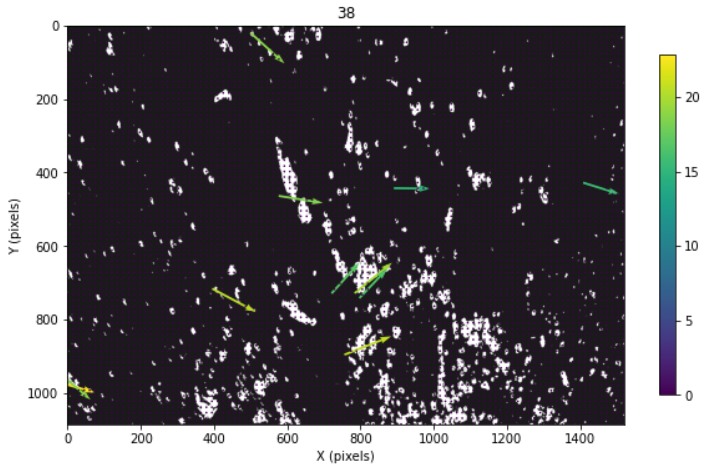
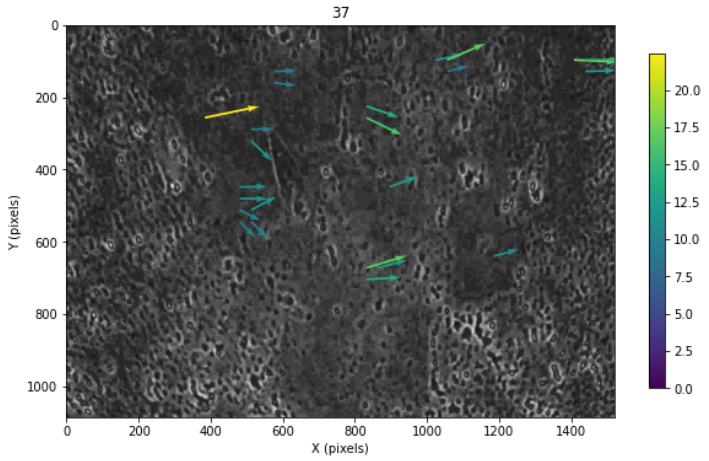


Figure 4.3: Results from "no flow" experiment in visible light.

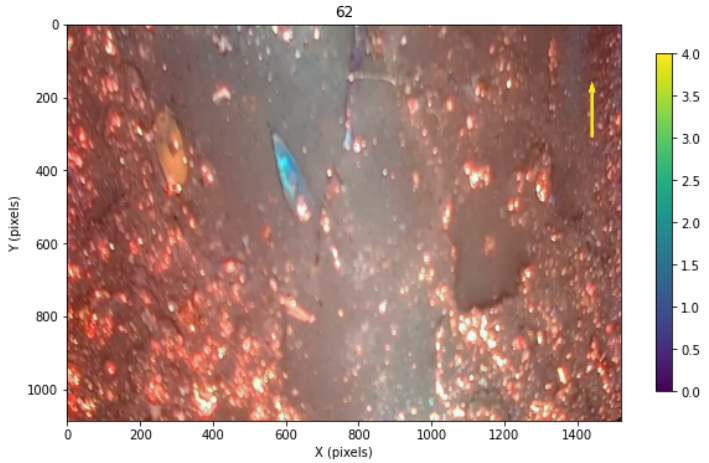
Ultraviolet/visible light - NO FLOW



OF (th)



PIV (gb, lpbf)



PTV

Figure 4.4: Typical noise during "no flow" experiment in visible/ultraviolet light.

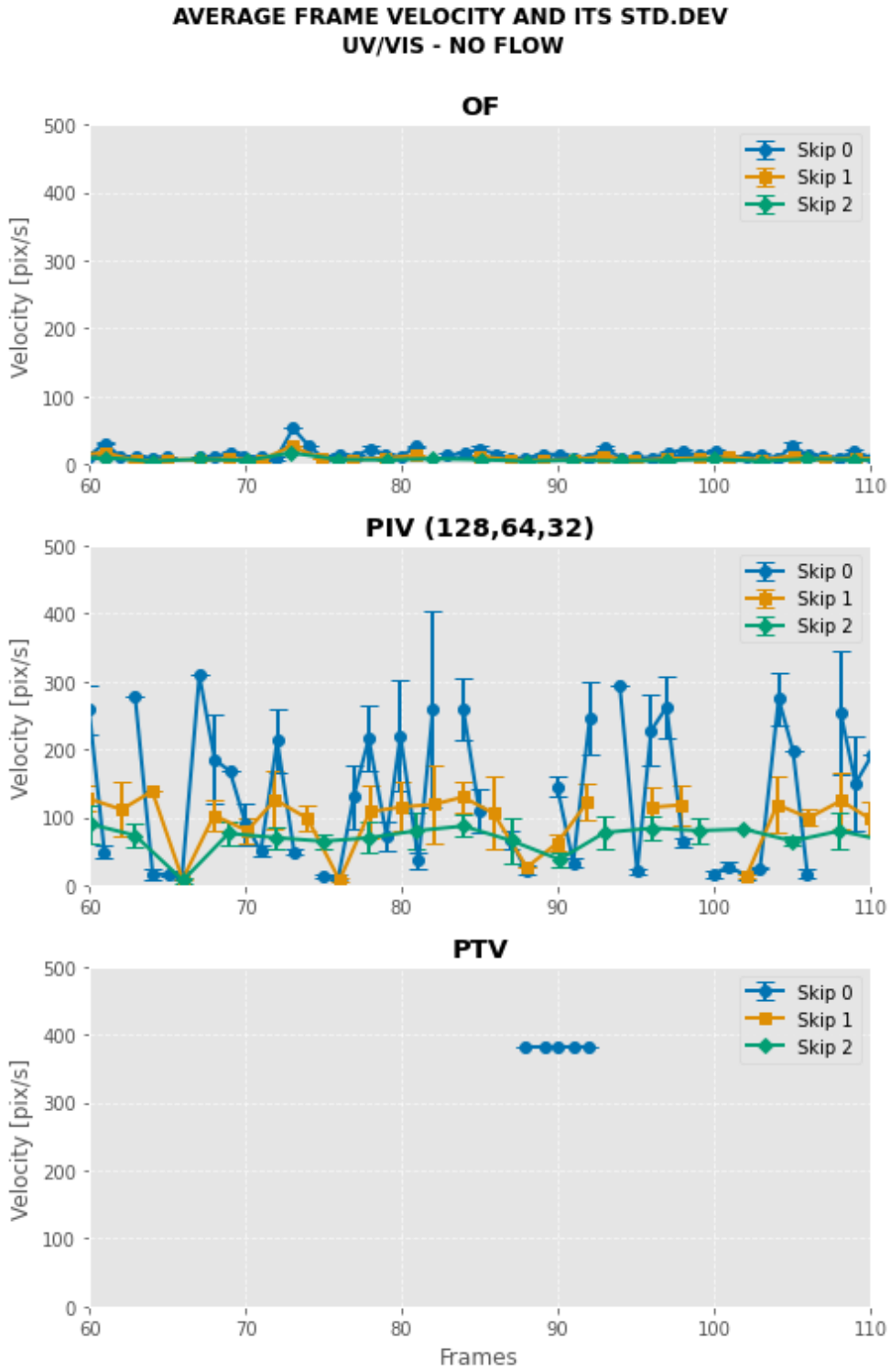
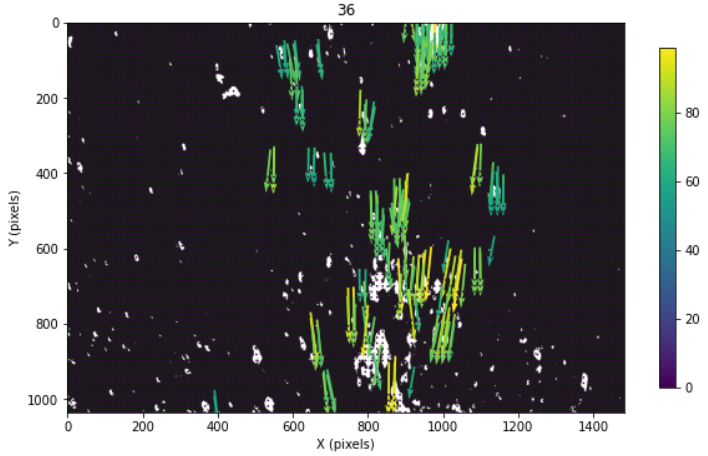
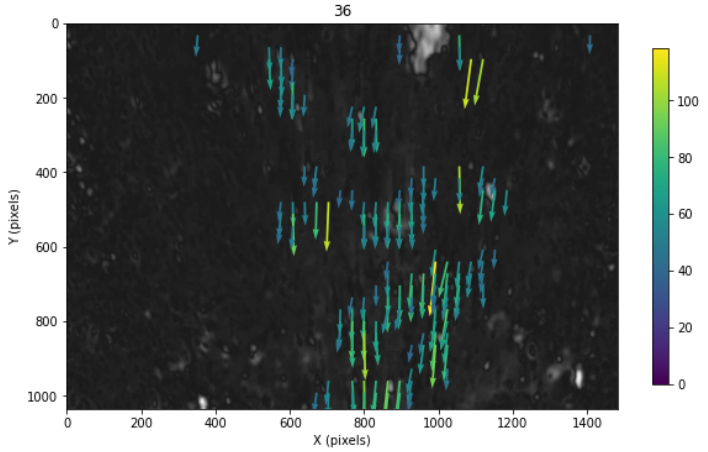


Figure 4.5: Results from "no flow" experiment in ultraviolet/visible light.

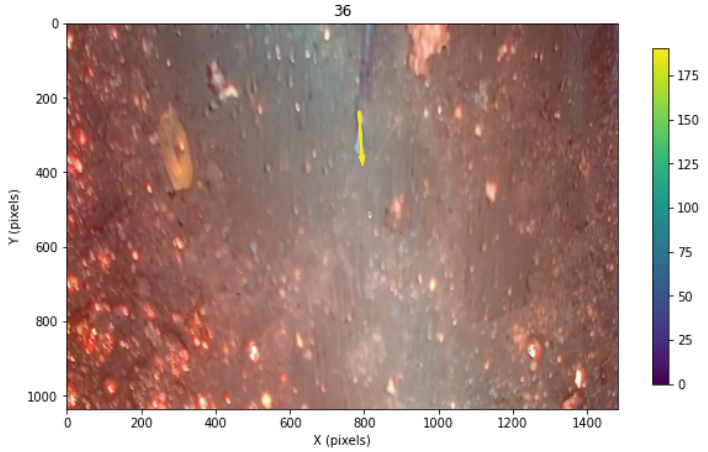
Ultraviolet/visible light - FLOW



OF (th)



PIV - (gb, lpbf)



PTV

Figure 4.6: Velocity vectors produced for the same with flow present in ultraviolet/visible light.

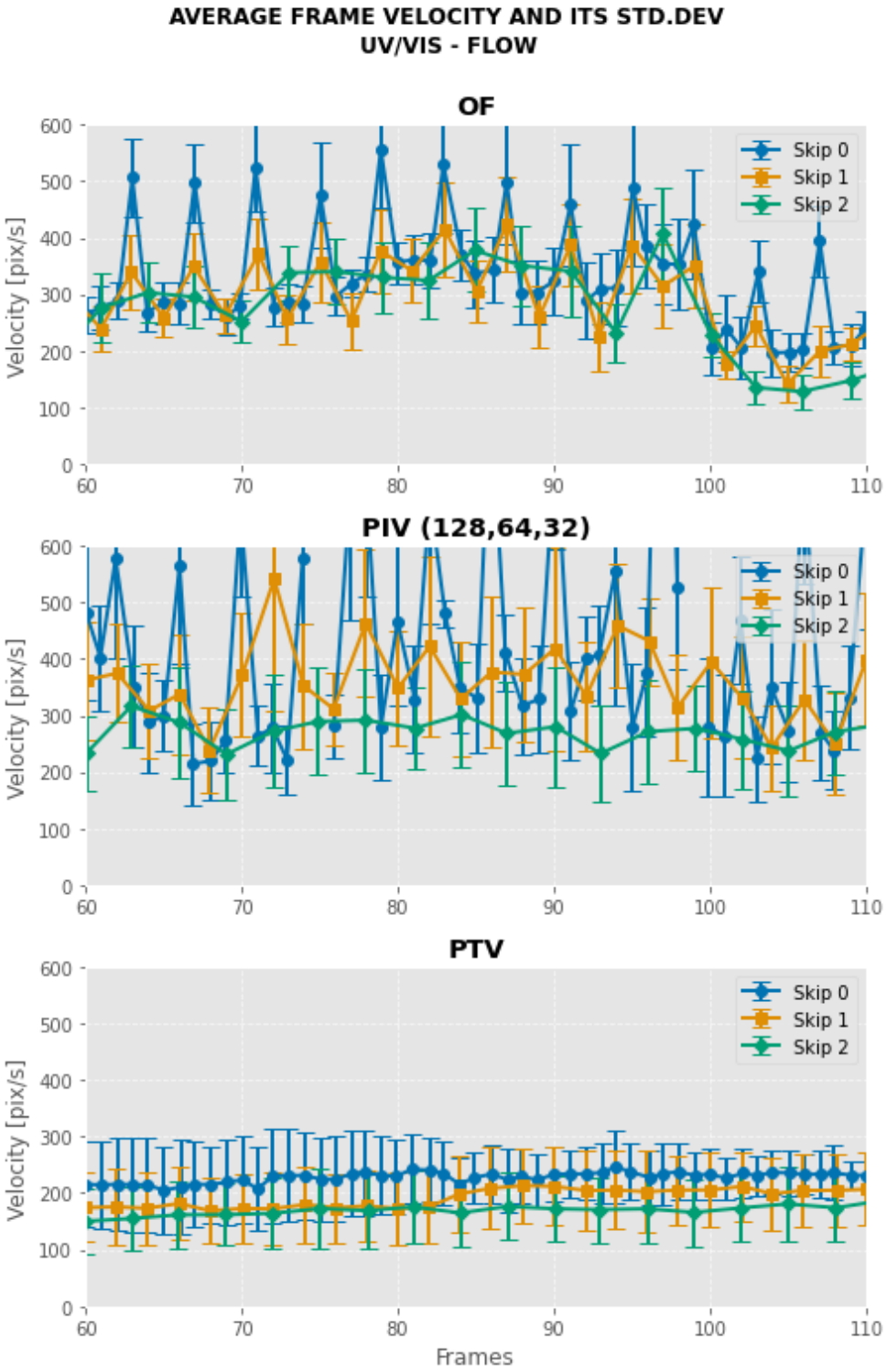
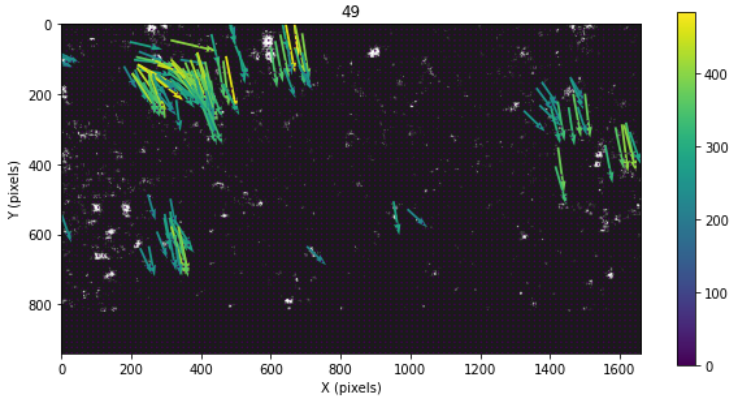
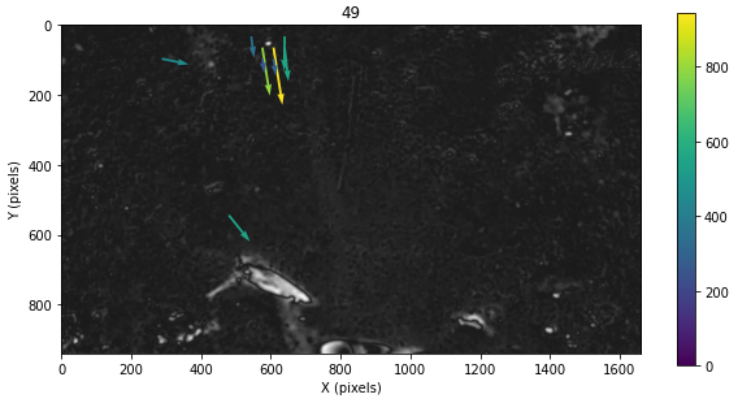


Figure 4.7: Results from flow experiment in ultraviolet/visible light.

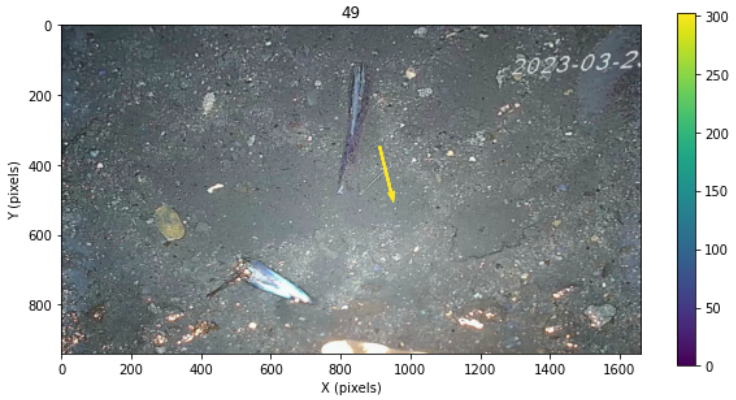
Visible light - FLOW



OF (mb)



PIV (gb, lpbf)



PTV

Figure 4.8: Velocity vectors produced for the same frame with flow present in visible light.

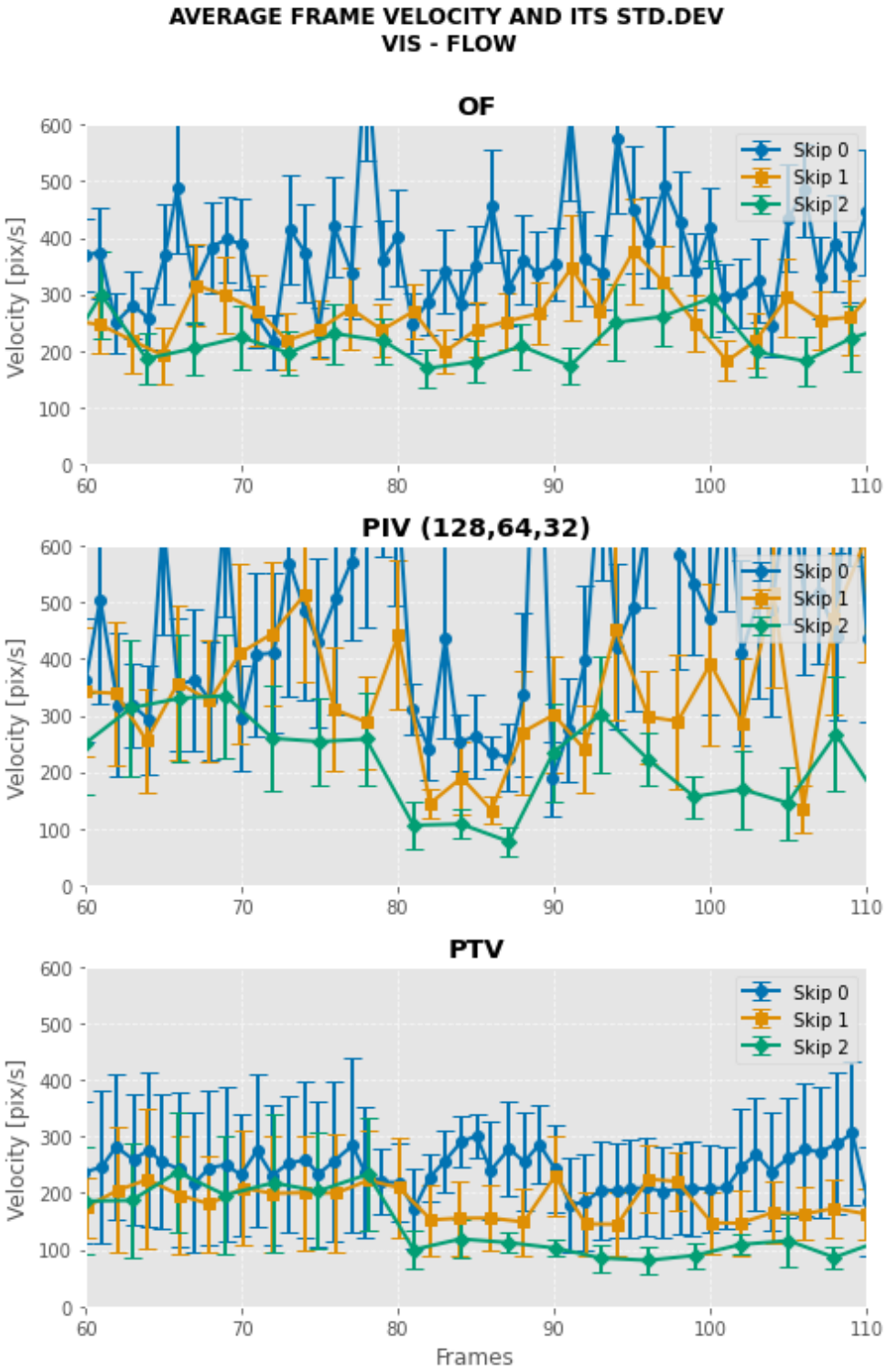


Figure 4.9: Result from flow experiment in visible light.

The average standard deviation of the frame velocity for the experiments conducted with flow present is presented in the bar chart in Fig. 4.10. Fig. 4.11 shows the average frame velocity of the velocities produced for the experiments conducted with no flow present. It should be noted that velocity estimates were not produced for all frame evaluations in the "no flow" experiment. The average number of frames for which velocity estimates were obtained from each algorithm is presented in Table 4.2. Additionally, the average frame processing time for each algorithm is provided in Table 4.1.

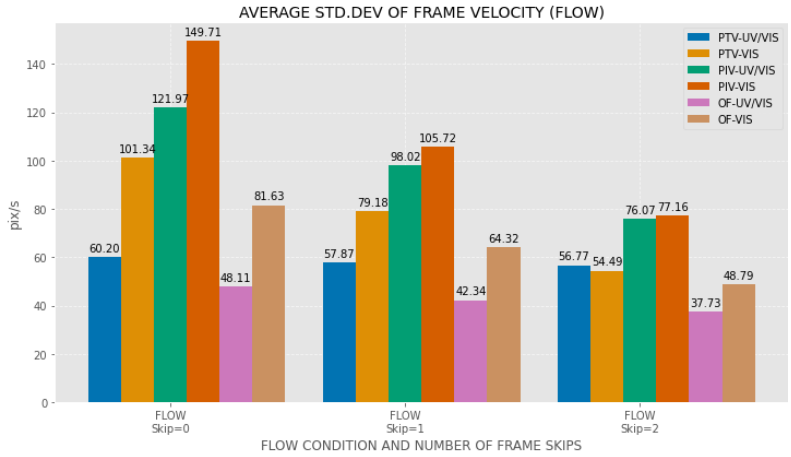


Figure 4.10: Std.dev of frame velocities during "flow" condition

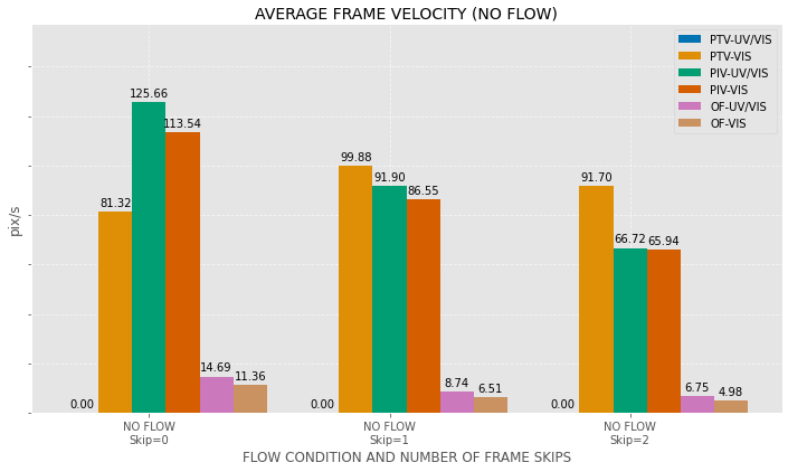


Figure 4.11: Average velocities produced in "no flow" condition

ALGORITHM	AVR. COMPUTATION TIME [sec/frame]
PIV	4.0
PTV	2.9
OF	0.89

Table 4.1: Average computation time per frame.

ALGORITHM	AVR. FRAMES PRODUCING VELOCITY IN NO FLOW
PIV	192/200
PTV	6/200
OF	169/200

Table 4.2: Fraction of frames producing velocities in the "no flow" experiment (skipping zero frames).

4.2 Calculated Effect of Skipping Frames on Uncertainty

An attempt was made to calculate the uncertainty and expected reduction in uncertainty by skipping frames. The uncertainty in pixel location was estimated by visually investigating a frame and subjectively determining the range that could be considered as the "edge" or "limits" of the particle in the digital image. This uncertainty will vary with different velocities, making the predicted uncertainty specific to the velocity being evaluated. For the time uncertainty, it was assumed that the shutter speed is the main contributor, and estimated to be 3% of the shutter time based on the findings of Simon et al. (2022) in their assessment of different methods to measure exposure time. By using eq. (4.1) with a ± 3 % deviation in the time stamp and a ± 1 pixel uncertainty in location at an approximate 5 pixels per frame displacement, the standard uncertainty of the velocity was calculated to be 59.06 pix/s for Δt = 0.05s, 29.52 pix/s for Δt = 0.1s, and 19.68 pix/s for Δt = 0.15s.

$$u_v = \sqrt{\left(\frac{\partial v}{\partial x_1}\right)^2 \cdot (\Delta x_1)^2 + \left(\frac{\partial v}{\partial x_2}\right)^2 \cdot (\Delta x_2)^2 + \left(\frac{\partial v}{\partial t_1}\right)^2 \cdot (\Delta t_1)^2 + \left(\frac{\partial v}{\partial t_2}\right)^2 \cdot (\Delta t_2)^2} \tag{4.1}$$

u_v : Standard uncertainty of velocity

$\frac{\partial v}{\partial x_1} : -\frac{1}{t_{\text{avg}}}$

Δx_1 : Uncertainty in x_1

$\frac{\partial v}{\partial x_2} : \frac{1}{t_{\text{avg}}}$

Δx_2 : Uncertainty in x_2

$\frac{\partial v}{\partial t_1} : \frac{d}{t_{\text{avg}}^2}$

Δt_1 : Uncertainty in t_1

$\frac{\partial v}{\partial t_2} : -\frac{d}{t_{\text{avg}}^2}$

Δt_2 : Uncertainty in t_2

4.3 Camera Placement and Sensor Location Optimization

The decision on where to place the camera in the field implementation made use of hydraulic simulations in SWMM to assess various inflow scenarios. The objective was to identify sensor locations highly sensitive to inflow by identifying conduits that detect velocity changes (10% increase/decrease), even for lower inflow intensities, ensuring robust results by having larger velocity increases for the same storm events. The selection of camera placement involved a trade-off between achieving a sufficiently low inflow intensity detection limit and maximizing the length of the upstream network to enhance the chances of observing illicit inflow. Among the sensor locations that detect all possible inflows (detection likelihood of one) for the same inflow intensity, the location was subjectively chosen on the basis of what seemed the best. How the different sensor locations compare is illustrated in Fig. 4.12.

Considering that the camera would remain in place long enough to capture a rainfall intensity capable of causing 1.3 lps inflow in the case of illicit connections, the chosen placement was conduit "LEI387". Another viable option, "LEI390," offered even greater robustness in detecting smaller inflow scenarios. However, it was not selected due to its location in a t-junction with potentially ambiguous hydraulic conditions that could impact the quality of footage retrieved for optical velocimetry algorithms. The upstream network from the placed camera is shown in Fig. 4.13 as a directed graph showing flow direction.

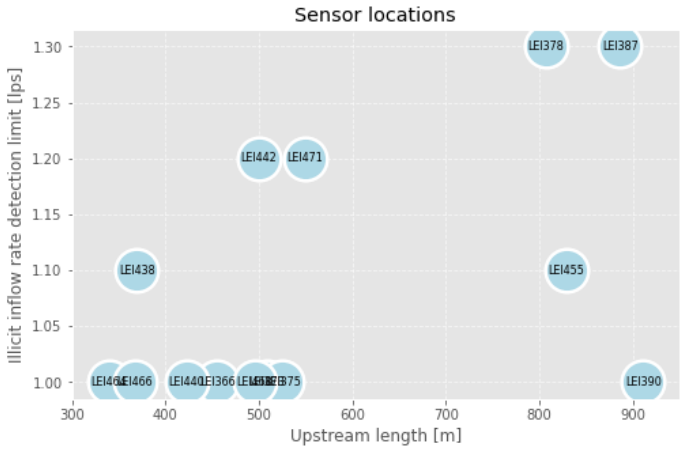


Figure 4.12: Conduits ("LEI" = conduit) are placed in the diagram based on their upstream network length and the lowest inflow rate where they still are able to detect all inflow scenarios.

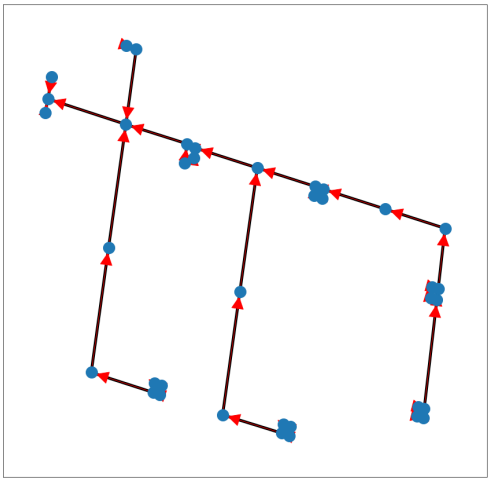


Figure 4.13: The upstream system of the camera as a directed graph with arrows indicating the direction of flow.

If the method of using camera sensors and optical velocimetry is to be implemented at a later stage, it is likely that the camera will be used as a preliminary investigation where multiple cameras are placed to quickly assess which parts of the system require further investigation using other methods. To demonstrate a possible approach for tackling such a sensor location problem with multiple cameras, an artificial example was created to make use of the genetic algorithm. The goal of the genetic algorithm was to find the most effective implementation of the cameras in combination with another more thorough investigation method, assumed here to be DTS.

The non-dominated sorting genetic algorithm (NSGA) was employed in the

optimization problem to identify the best combination of camera placements in the system, considering three available cameras. The optimization process aimed to maximize the upstream length/area of each location, the detection likelihood of the location, and select locations with upstream areas suitable for investigation using DTS. After 10 generations of mutation, selection, and recombination, the resulting solutions in the latest generation were plotted in Fig. 4.14, showing how the subsets in different fronts (by colour) compare to each other with respect to the objectives.

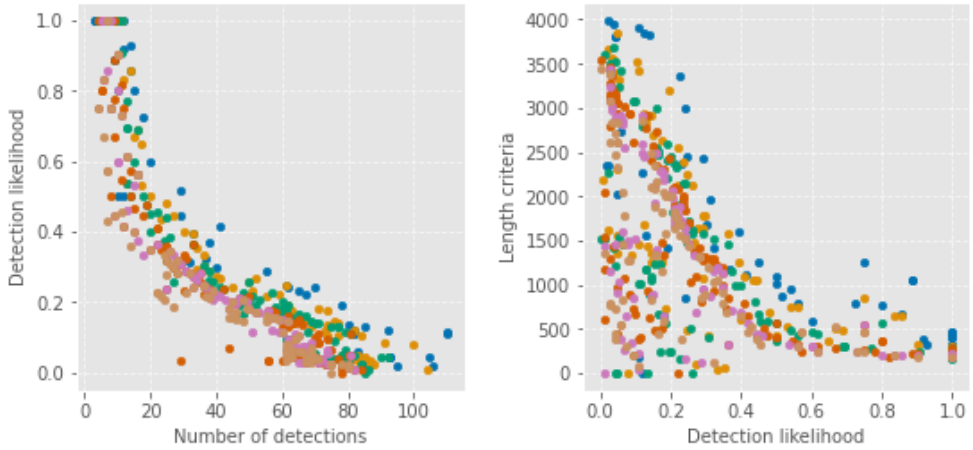


Figure 4.14: All solution in the first front (blue) dominate solution in all other fronts (other colours) as it is better in at least one objective and not worse in any other objective

In this particular simulation, a very low inflow scenario was used, which means that almost no locations will be able to detect all inflow scenarios and have a detection likelihood of one. The detection likelihood then becomes a sensitivity measure, where locations that detect more scenarios at low inflow are assigned higher values. The alternative, running the simulation at a higher inflow scenario would have resulted in all subsets having a detection likelihood of one, making it impossible to distinguish based on sensitivity.

Solutions within the first front are better in at least one objective and not worse in any other objective, but selecting the best solution among those in the first front requires a trade-off and subjective decision to prioritize one objective over the other. The selected solution is shown in Fig. 4.15. The genetic algorithm took 8.5 minutes to iterate over 10 generations with a starting population of 1000 solution subsets.

The subsequent DTS implementation, following the detection of illicit inflow at the selected camera locations, involves solving a shortest path problem to traverse all the nodes in the upstream network. The best solution to this problem is presented in Fig. 4.16, focusing on one of the nodes in the best subset identified by the genetic algorithm. The red node represents the placement of the DTS com-

puter and infrastructure, while the double arrows indicate that the cable is looped to cover dead ends in the system.

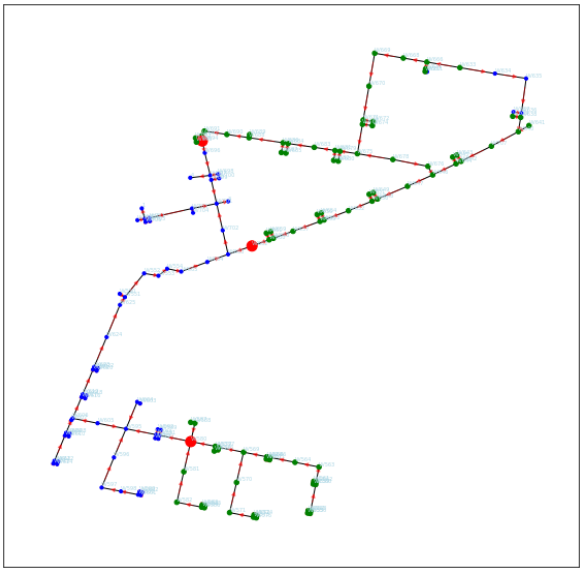


Figure 4.15: The subset of sensor location selected from the first front found by the genetic algorithm (GA) when optimizing for implementing three cameras. The sensor locations are marked in red and their upstream nodes in green.

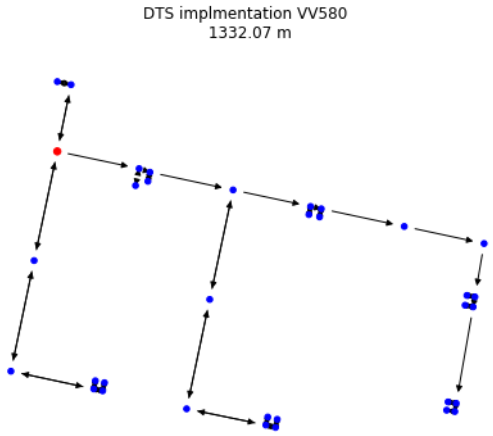


Figure 4.16: The DTS implementation, following the detection of illicit inflow, involves solving a shortest path optimization problem to traverse all the nodes in the upstream network with the cable. The best solution to this problem for node "VV580" is presented with the red node as the placement of the DTS computer and infrastructure, while the double arrows indicate that the cable is looped to cover dead ends in the system.

4.4 In-Field Placement of the Camera Sensor

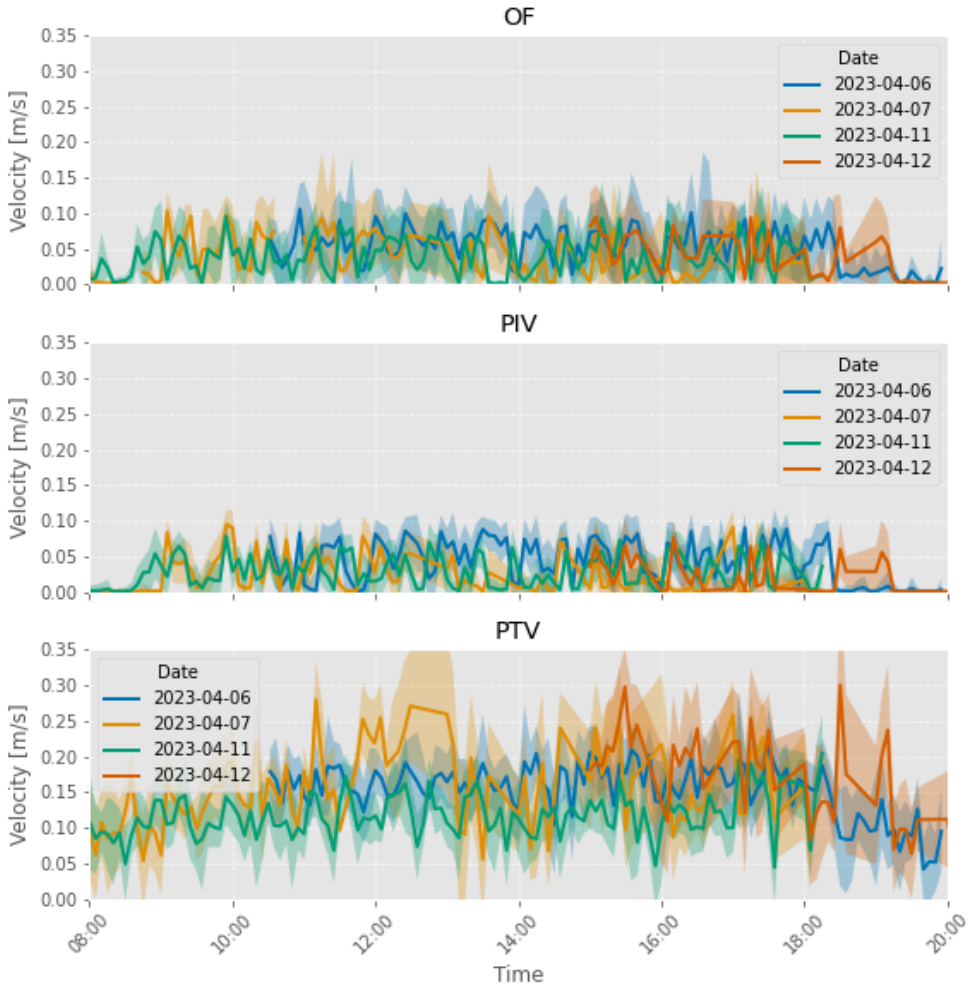


Figure 4.17: Velocity time series produced by processing videos from field implementation in Tilburg. The bandwidth of each series represents the standard deviation of the extracted frame velocities for each sequence (one for each pair of frames), depicting the range of velocities that the algorithms "observed" to produce the measurement points.

The time series obtained from analysing the footage captured during the field placement of the cameras are presented in Fig. 4.17. The mean frame velocity is first found from all pairs of frames for OF, while the maximum frame velocities are extracted for PTV and PIV. The extracted frame velocities are then averaged to produce each measurement point in the series.

Two out of the four series were captured during rain events. Although the rain intensity was below the threshold considered reasonable for the method to detect illicit inflow, the strategy of modelling the rain event scenario in the hydraulic

model can still be applied. An estimation of the roof/catchment areas that may be wrongly connected upstream was mapped to create relevant scenarios (Fig. 4.18).

Figure 4.18: Data on roof catchment/area were mapped and linked to the nearest manhole to create relevant illicit inflow scenarios in the model.

The measurement series that were known to not be influenced by rain were used to compute an average time series, which is considered the baseline flow of the system during dry weather. However, it should be noted that this average is based on only two series and is not sufficient to establish a reliable pattern. Nonetheless, the average series was used to model an inflow series for the hydraulic model, serving as the baseline when simulating different illicit inflow scenarios.

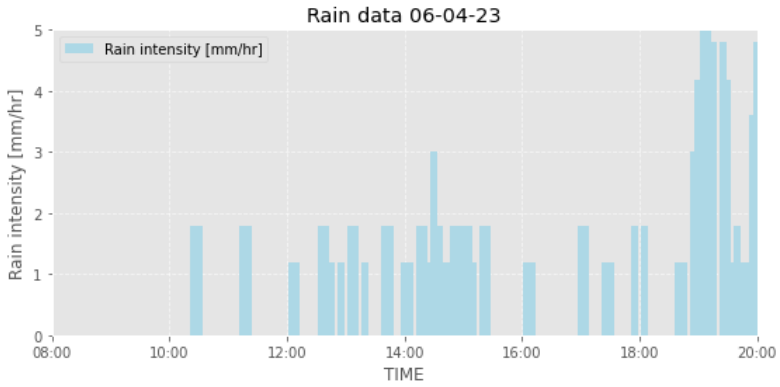


Figure 4.19: Rain data from a measuring station in close proximity to the placed camera was used when generating inflow scenarios in the hydraulic mode.

Modelling the observed velocity was implemented in the SWMM model by first using the measured velocity series to iteratively solve Manning's equation to determine the flow rate series required to generate the observed velocities in the model, with slope, Manning roughness coefficient, and diameter obtained from system data. The estimated flow series was incorporated as an inflow series at the upstream node closest to the camera sensor location in the model. Additionally, a

low baseline inflow was added to all other upstream nodes to ensure the presence of flowing water throughout the system. The calculated estimated flow series then had to be scaled down by a factor until the best R2-value was achieved, which is computed as the squared differences between the estimated flow series and the real-world measurements divided by the total sum of squared differences, serving as a measure of how well the model captures the variability in real-world measurements.

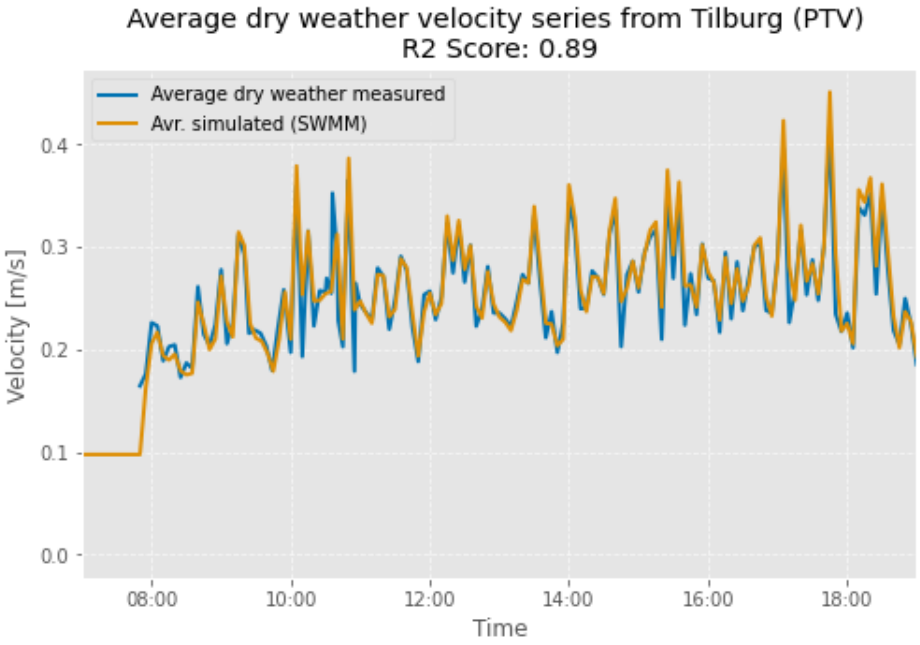


Figure 4.20: The values measured during dry weather were used to calculate an average series that was used to generate inflow in the model to reproduce the velocity series in the model. The plot shows how the avr. simulated compared with the average dry weather series that was measured.

The comparison between the modelled series and the average "dry" weather series is shown in Fig. 4.20. Two weather stations located near the camera placement location were used to verify the precipitation event used in the inflow scenario simulation. Modeled inflow scenarios are compared with the actual measured velocity series in Fig. 4.21.

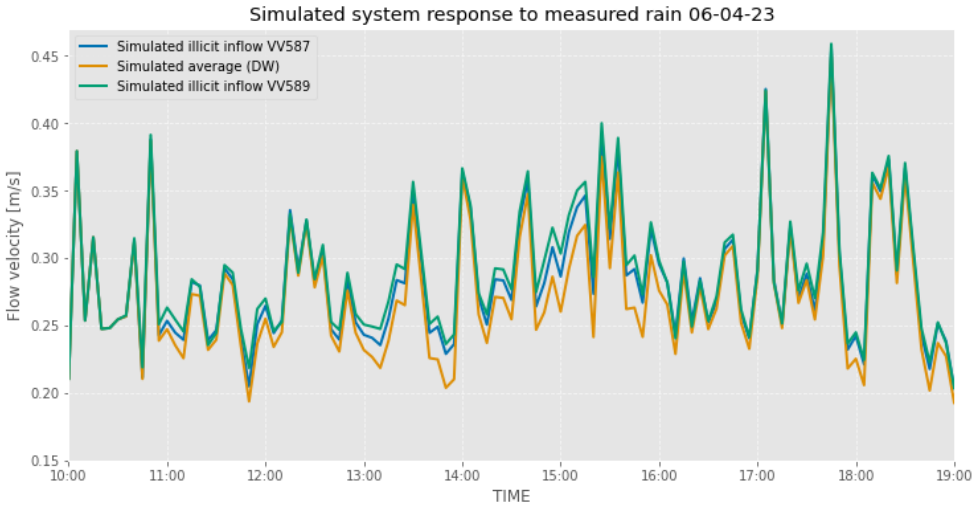


Figure 4.21: Rain/inflow scenarios were simulated in the model on top of the simulated average inflow series to see the system response in terms of increase in velocity for different scenarios. This is to see if the rain event would have caused a velocity increase big enough in case of an illicit connection upstream from the location where the camera was placed. Two velocity series for two caused by two inflow scenarios are shown in the plot.

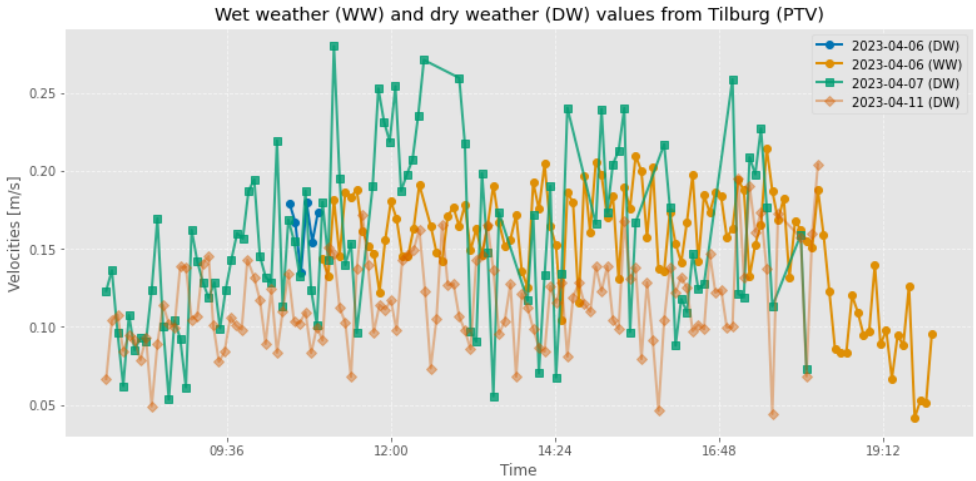


Figure 4.22: Velocity series from the same time window as rain was present in one of the series is extracted to label them as "wet" and "dry" according to if they were captured in some time after the onset of rain and potentially could have been influenced by it. The values are then used to compute the t-test by pairing all "wet" values with all the "dry" values from the same time step.

Analyzing the "wet" and "dry" values (Fig. 4.22) gathered during the short measurement campaign to identify deviations is of arbitrary nature due to the limited amount of data available to establish a true baseline. Nonetheless, attempts

were made to determine if the measured velocities after the rain were significantly different from the dry weather series. The "wet" weather (WW) values were extracted and paired with the "dry" weather (DW) values to compute the statistical significance of the observed difference using the t-test. The test statistics from the t-test were lower than the critical value (set for a 95 % confidence level), indicating that the probability of obtaining the observed data by chance alone was greater than 5 %. As a result, we cannot confidently conclude the presence of an illicit inflow without a (low enough) risk of it being a false positive.

Chapter 5

Discussion

5.1 Optical Velocimetry Experiment

Ideally, the optical velocimetry experiment should have involved testing several different mounts, frame rates, light sources, angles, elevations, and image resolutions to determine the conditions under which the algorithms perform best and the range of conditions in which they can be applied. However, conducting such experiments within the environment of a manhole and with the intended camera proved challenging. Prior to the conducted experiment, there was a preliminary phase of attempting various combinations of angles, light sources, light reflection reduction measures, and different camera heights above the surface in an iterative process to improve the quality of the footage captured by the cameras.

(a) Wooden cover/mount giving a birds-eye-view of the manhole.

(b) Steel cover/mount with penetrating rods allowing different length and angles

Figure 5.1: Mounts and equipment used during preliminary experiments

Different mounts and equipment, as depicted in Fig. 5.1 and 5.2, were assembled for this purpose only, using very basic tools and techniques. However, inconsistencies in their functionality, made it difficult to reproduce experiments and control the variables effectively. The preliminary tests highlighted that achieving good light conditions (especially the absence of reflections on the wet surfaces) would be the most significant challenge. Regardless of the setup used, there were always issues such as reflections, insufficient or uneven lighting, and light phenomena such as dark spots or glare appearing in the frames, resulting in inconsistent results even with similar setups. It was therefore focused on the setup that

consistently provided reliable results with the available camera, equipment and light sources.

(a) Wooden mount with the camera at an angle to the surface.

(b) 3D printed cone used for reducing light reflections.

Figure 5.2: Mounts and equipment used during preliminary experiments

Although the experiment was conducted using similar setups under two light conditions (UV/VIS and VIS), with the same amount of water poured into the upstream manhole, it is important to note that the amount and characteristics of particles present in the sewer may vary significantly, and this can have a significant impact on the performance of the algorithms. In the experiment, the UV/VIS condition was tested before the VIS condition, which resulted in more particles being visible in the UV/VIS footage due to the initial flush.

Quantitatively comparing the effort put into tweaking the parameters for the different algorithms or the overall quality of the algorithms is challenging, especially since they were developed by different individuals with varying programming abilities. Additionally, the experiment's findings cannot be generalized to all possible variations in wastewater characteristics and video quality. Therefore, it would be inappropriate to conclude which method or setup is the best overall based solely on these results. However, the experiments are valuable as exploratory research to better understand the possibilities and limitations of each algorithm. Interpreting and analysing the results can provide insights and potentially uncover new knowledge about the algorithms' capabilities and limitations in different conditions.

5.1.1 Experimental Results

When there is no flow present in the monitored conduit, the algorithm also needs to produce a "zero flow" velocity. This is particularly important when detecting the presence of flow during times of the day when the system usually runs dry. If

the algorithms are unable to produce "zero flow" and instead generate velocities even when there is no flow, there is a detection limit for the algorithm. This limit represents the lowest velocity that can be distinguished from the measurement noise.

In the experiment, the detection limit is investigated by examining the amount of displacement produced when there is no flow in the processed video. In this regard, the Optical Flow (OF) algorithm performs better than both Particle Image Velocimetry (PIV) and Particle Tracking Velocimetry (PTV) as OF "finds" the least amount of displacement in the "no flow" video sequence, resulting in the lowest average frame velocities. Particularly noteworthy is the fact that PIV produces vectors in the "no flow" condition that could easily be mistaken for flow vectors because they are of similar magnitude and are present throughout, making them potentially harder to filter out as noise. This is likely because a very low signal-to-noise (S2N) ratio was achieved. S2N is in the context of PIV a measure that indicates the likelihood of the discovered displacement being caused by a moving object and not just image noise. By setting an S2N threshold, vectors can be filtered out based on this criterion. However, when all displacements have a low S2N, it becomes unfeasible to apply this filtering process. While PIV can estimate displacement more or less accurately from stronger signals (S2N) during flow, the algorithm still produces displacement when the signal-to-noise is slightly worse during no flow since no filtering is applied. The inherent measurement noise for PIV in the absence of flow essentially means that any values below the measured displacement found in the "no flow" condition in the experiment must be considered as zero velocity measurements if implemented in a similar camera setup with similar image quality.

The PTV algorithm also produces velocities in the "no flow" condition but less frequently than PIV, as only 6 out of 200 frames produce velocity. PTV only generates velocities from particles recognized in consecutive frames, making it highly unlikely to produce displacement from image noise. However, PTV is very capable of detecting motion from other passing objects in the image unrelated to flow and could potentially produce displacement from bugs or other objects flying by the image, resulting in velocities. Since these would appear as significant increases for only a few consecutive frames, they are easier to distinguish and filter out compared to the case of PIV. Both PTV and OF benefit from having a temporal aspect in their methods of finding displacement, as OF uses previously calculated flow fields to compute the next, and PTV averages over several frames.

Producing no velocities at all can also pose a problem as it makes it impossible to actually state that the sensor was operational and functioning properly throughout the measurement campaign and that there are no malfunctions or problems with the camera. This lack of velocity measurements can introduce uncertainty regarding the sensor's reliability and performance. That PTV in the "no flow" experiment produces no measurements has to do with the stringent criteria set for PTV as a particle is required to be tracked over several frames to be measured. This could be a reason to relax the criteria to possibly always have some presence

of "noise" as is the case of OF as shown in Fig. 5.3.

The strict criteria is also the reason that PTV seem less capable of representing the flow when visually inspecting the vectors for each frame as it produces way fewer vectors. Reducing the number of consecutive frames required for the PTV to successfully track a particle to only two consecutive frames will possibly generate a greater number of vectors for each frame, but will not improve the velocity estimation.

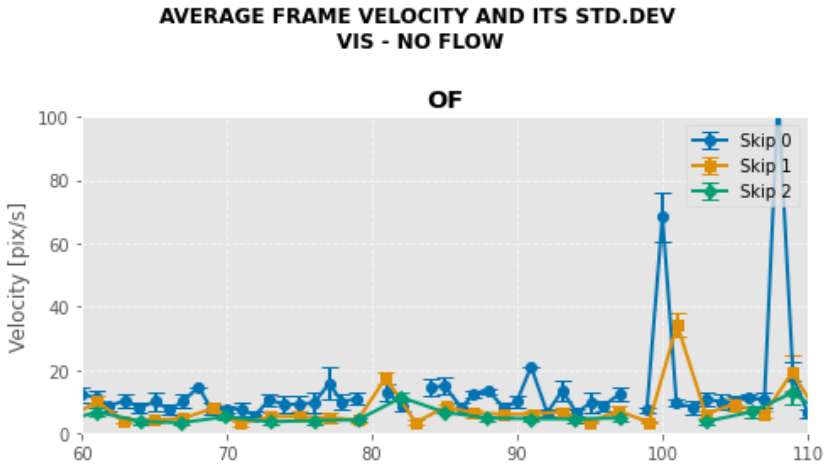
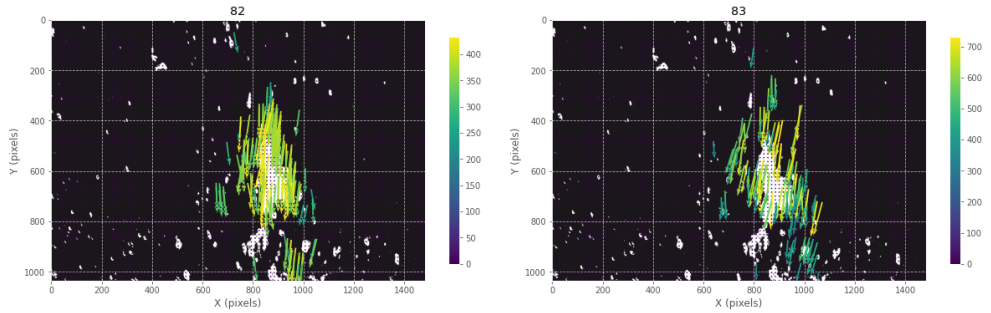


Figure 5.3: The optical flow consistently produces small velocities during "no flow" making it possible to know that the sensor actually was operational and measuring.

From the time series produced with flow present by each algorithm, it seems that all are capable of representing the flow in terms of providing somewhat consistent average velocity measurements within the short time frame of the experiment, falling within the range of what is found when manually measuring displacement with video tracking software. Two implementations of PIV were tested to see if increasing the window size improved the results. When increasing the maximum window size to 128 by 128 pixels, a higher average velocity is achieved. This could be due to the smaller windows' inability to accurately describe the real displacement if the displacement is larger than the windows, or it could be that the larger windows, with a more "global flow focus," capture fewer variations and therefore produce larger velocities, resulting in a higher average. However, increasing the window size also comes with a cost as it becomes more susceptible to producing larger spurious vectors. Three iterations with window sizes of 128, 64, and 32 were used in processing the experimental video, but trying out different window sizes showed that the results from the algorithms might be highly dependent on the algorithm settings. This is not necessarily a problem if measurements are consistently made with the same settings.

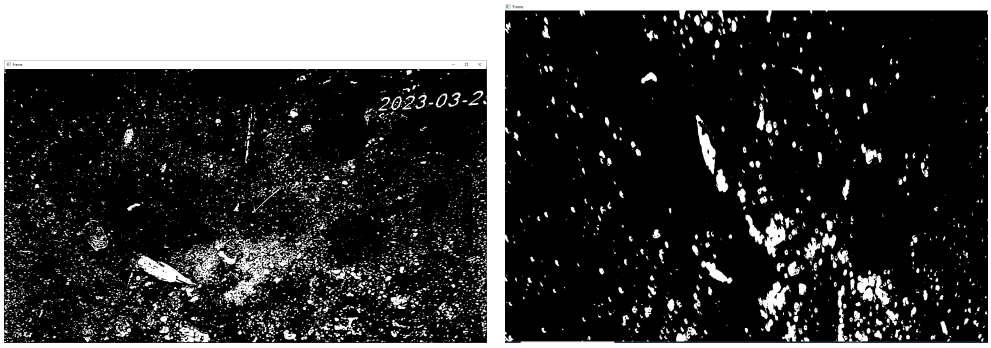
PTV and PIV with small interrogation windows produce the lowest average velocities, lower than OF and PIV with larger windows. From visually inspecting the

quiver plots produced for frames by all algorithms, it seems that OF has a tendency to produce exaggerated vectors when big particles are present in the frame. An example of this is visible in Fig. 5.4. As OF looks for movement of intensity gradient within the image, this effect may be caused by the applied pre-processing procedure (thresholding), where varying amounts of the particles are visible in two consecutive frames, causing the apparent edge of the particle to appear to move more than the actual particle displacement. The fact that PIV and PTV produce lower velocities, and smaller window PIV produces lower averages than larger window PIV, could be due to their ability to represent more of the flow velocity variation in the frame, thereby capturing the full range of velocities present to compute the average. OF and PTV produce very similar average standard uncertainties, which likely relate to both having temporal aspects to their displacement estimation.



(a) Frame 82 **(b)** Frame 83

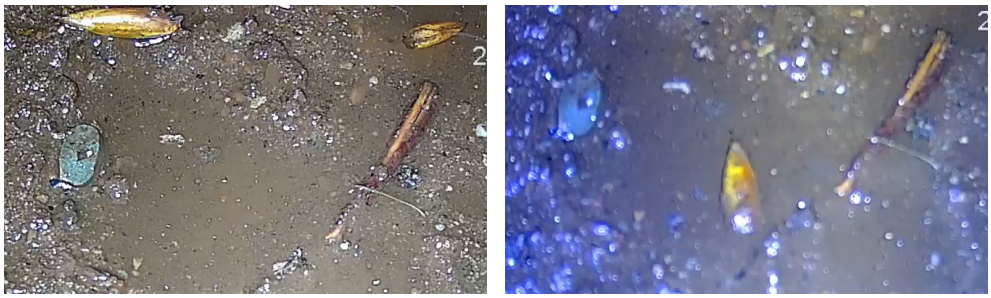
Figure 5.4: OF has a tendency to produce exaggerated vectors when big particles are present in the frame.



(a) VIS-footage thresholded **(b)** UV/VIS-footage thresholded

Figure 5.5: It was more difficult to separate particles from the background in the VIS-footage. The same threshold limit is applied to both VIS and UV/VIS footage, resulting in more alternating intensity throughout the frame for the VIS footage, causing particles to still be difficult to distinguish.

Higher standard deviation is experienced in both flow conditions in the footage captured in visible (VIS) light only. The higher frame velocity could simply be due to the fact that there were fewer particles present during the VIS experiment, making them more difficult to track. However, it could also be attributed to differences in light, focus, and image quality. When using only visible light, it seemed more challenging to distinguish between particles and the background conduit surface in the image, also when attempting to enhance visibility. An example of this can be seen in Fig. 5.5, where the same threshold limit is applied to both VIS and UV/VIS footage, resulting in more alternating intensity throughout the frame for the VIS footage. One possible explanation for this could be that lower wavelengths (for UV) are being absorbed more by the water, causing greater intensity differences between floating particles and the conduit surface as the background. Another reason could be that the effects are simply caused by the amount of light present in the VIS, as the number of LEDs emitting visible light is reduced from 8 to 2 when used in conjunction with UV. The cameras were mounted at a similar angle, but the reflections of the diodes are only visible in the VIS footage. Although these reflections were masked out in the image before processing the footage, the light reflected back onto the lens during image capturing could certainly degrade the image quality. Even when positioning the camera to avoid direct reflection from the water surface, there will always be other surfaces within the camera's view that have the potential to cause reflections, either due to being wet or having a reflective biofilm. A third reason, and possibly verifying the hypothesised reason for attempting UV as an alternative to visible light in the first place, could be the presence of fluorescent organic compounds causing distinguishable particles that are easier to track.



(a) Visible light **(b)** Visible and ultraviolet.

Figure 5.6: Comparison of footage from the experiments.

It may appear that the footage captured in UV/VIS is blurrier compared to VIS-only. The two are compared in Fig. 5.6. This could be attributed to lower wavelengths and higher frequencies causing the rays to scatter more and hit the lens from various angles, resulting in a loss of image sharpness for UV/VIS. This blurriness could potentially make it more challenging to achieve good results with PTV, as it becomes harder to distinguish particle shapes. However, it might benefit

PIV, as the grainy nature of the image can be interpreted by the algorithm as moving patterns.

Different pre-processing steps were applied to the same videos before processing them with different OV algorithms. The choices of appropriate pre-processing were based on subjective observations and trial and error, considering the understanding of the algorithms. Similar pre-processing was performed for the four different videos being processed by the same algorithm. For PIV, applying lowpass background subtraction after Gaussian blurring worked better for the UV/VIS footage compared to VIS. Lowpass background subtraction not only removes background noise but also results in a grainier image that resembles the clusters of moving particles that PIV requires. The effectiveness of this procedure for the UV/VIS footage may be due to a more optimal blurring achieved with the combination of pre-processing and original image quality. In the case of PTV, no pre-processing was applied. Thresholding was attempted for PTV but did not show significant benefits. Algorithms that smooth edges or make particles more uniform in size would make it more challenging for PTV to distinguish particles from each other. For OF, on the other hand, binarization techniques were necessary to enhance particle visibility and obtain meaningful results. Motion binarization was applied to the footage captured in visible light instead of thresholding, as it provided similar benefits but was easier to apply without having to meticulously adjust the threshold to distinguish particles from the background.

There is a significant difference in the time consumption to process frames in the different algorithms, which is influenced by algorithm parameters related to numerical precision and result resolution. In the experiment, accuracy was prioritized over efficiency. Both PIV and OF process the images multiple times to refine velocity estimates, including calculations at different image pyramid levels and resolutions for displacement calculation in all interrogated windows. This process is more complex for PIV as it involves computing the shifting and deforming of the interrogation windows. PTV is in the experiment implementation slowed down by the high number of particles it detects and calculates displacement for. PTV is usually the fastest with not as many particles, as it only calculates displacement for particles that are present in several previous frames, and it discards/skips frames with no or too many particles (related to noise), contributing to efficiency.

The PTV algorithm requires more finesse as the selection and tracking of particles can be made highly complex. PIV and PTV are relatively easier to implement using "standard" algorithms from computer vision libraries, while still achieving good results. It is also possible to use simpler versions and implementations of algorithms from PTV libraries, but these may yield lower-quality results, as they are usually prepared for other types of footage with potentially more distinct differences between particles and the background, as well as more uniform particles.

5.1.2 Calculated Uncertainty

The calculated uncertainty was similar to that found in the measurements for OF. There was also a similar benefit observed for increasing the time between evaluations in the theoretical calculations as in the experiment, where skipping one frame reduced the uncertainty by 20 pix/s, and skipping two frames further reduced it by another 10 for OF in visible light. The calculated uncertainty was also close to the standard deviation observed in PTV in UV/VIS conditions. This suggests that the assumed uncertainties in the variables used in the computation are valid and that it is reasonable to assume negligible covariance terms. However, the fact that only some combinations of algorithm and light condition approach the calculated uncertainty could be because these combinations achieve results that are closer to the optimal performance of the algorithms. Since the calculated uncertainty, in general, is lower than what is observed in the data, it is possible that the uncertainties involved are underestimated or conservative, or there may be covariance terms impacting the uncertainty that has not been accounted for.

5.2 Sensor Strategy

Drawing conclusions and making decisions based on the hydraulic model regarding the effectiveness of sensor locations in detecting illicit inflows and simulating inflow events is a process influenced by a significant amount of uncertainty. The data on the system contains some errors that could be investigated and corrected, but the model will never be able to account for all hydraulic conditions affecting how water enters the system and travels within it. Extensive flow measurements at various locations throughout the system and calibration would be necessary to produce highly accurate results. Whether or not the uncertainty impedes the effectiveness of the sensor strategy could not be fully tested as the collected data has not yet allowed for the verification of its success or failure in detecting illicit inflows. This could potentially be done by testing the strategy in systems with known illicit inflows or by assessing the general success of the strategy in detecting inflows when implemented over a certain period of time.

5.2.1 Finding Sensor Locations

In the proposed method for sensor placement strategy, the accuracy of the hydraulic modelling results is of less importance. Instead, the focus is on assuming that the accuracy could be equally incorrect for all locations and then applying a high enough rain/inflow to distinguish between sensors that detect inflows and those that do not, regardless of the inherent uncertainty. Having such a model and conducting scenario analyses would be particularly valuable in systems with less obvious hydraulic conditions, such as in the Netherlands where the lowest point in the system is not always the system outlet, as is often the case in Norway. If all steps of model configuration and inflow scenarios are automated, there is no

reason to refrain from investigating the sensitivity of different locations despite the uncertainty. Having some information to base the decision of sensor placement on is better than having no information at all. But whether or not the methods find the best locations can not be fully evaluated as testing all locations to check if any is better is unfeasible in practice.

5.2.2 Genetic Algorithm

The results from the Genetic Algorithm (GA) are, of course, no better than the input, but they are likely to provide a solution that is very close to being optimal for the problem at hand, and better than what any human could achieve in the same time. Using a non-dominated sorting genetic algorithm does not guarantee the optimal solution, but it provides (possibly) all solutions (in the first front) that are better in at least one objective and not worse in any other objective, allowing for further investigation of these solutions. Setting up the network for SWMM implementation and solving system optimization problems using NetworkX in Python offers the opportunity to consider both hydraulic and system-related aspects simultaneously, exploring the entire solution space in a more efficient manner compared to considering the two aspects separately. But the same goes for evaluating the performance of the GA, as for finding the best sensor locations, as it is impossible to test all other solutions to state that it is in fact the best overall.

5.2.3 Camera Sensor

The ideal camera placement would involve positioning it perpendicular to the water surface, with a wide enough field of view to capture the entire flow, even in cases where the channel capacity is exceeded. Although attempts were made in the preliminary experiments to achieve such a view, it was not feasible with the available cameras and mounts. Fig. 5.7 showcases footage captured with a GoPro in 4K, which closely resembles the "ideal" view in terms of quality and perspective. Placing the camera at a higher position is also advantageous to reduce the risk of submerging the camera or water splashing onto the lens. Additionally, mounting the camera vertically would minimize accuracy loss caused by perspective transformation.

To ensure accuracy, the camera should be securely held in place to minimize any image noise resulting from movement, such as passing cars, which could impact the measurement precision. Moreover, the installation process should be as straightforward as possible and enable operation from above ground. However, these criteria are limited by the challenging geometry, restricted access, and harsh lighting conditions typically present in manholes.

Having the camera and light source combined in one unit, such as the C70, simplifies the mounting process since there is no need to install an external light source. However, this integration also leads to reflections on the lens, as the light is directed at the same angle as the camera. These reflections create blind spots in the image and distort colours and details, adding difficulties to the analysis

process. Even when the camera is angled, reflections can still occur depending on the lens' angle of view and the positioning of the light source relative to the lens. It is in consequence not always possible to mount the camera in a way that eliminates reflections due to the restrictions posed by the manhole geometry.

Using indirect light sources, such as shielding them from hitting the lens or dispersing the light through filters or walls, may help alleviate the reflection issue. However, this approach requires ensuring that the entire scene is still well and evenly lit, which can be challenging. The equipment used to mount the camera should also be easily implementable and offer flexibility to accommodate different sizes and shapes of manholes. During experimentation with different placements, it was discovered that achieving a complete overview from just underneath the manhole cover was unrealistic without fine-tuning the light conditions and potentially using another camera.

In the field implementation, two expandable bars spanning between the manhole walls were used to hang the camera at an angle. This approach worked well, but it was not as straightforward to place as desired since it required descending into the sewer. A camera setup that closely resembles the ideal configuration, surpassing what was achieved in the field implementation, exists. However, this would likely necessitate a more flexible mount, a better camera, and a way to refine and optimize the lighting conditions that can be applied in any manhole. Two potential options for mounting the camera are presented in Fig. 5.8. One option utilizes adjustable bars with external light sources, while the other incorporates the camera and external light sources into a metal sheet that fits within the manhole opening.

Demonstrating the effectiveness of optical velocimetry with a camera and mount specifically designed for the purpose would likely be easier than proving the capabilities of a specific camera like the C70. While the C70 has many great qualities, such as its sturdiness, it lacked the necessary flexibility at this stage of development. Ideally, the camera or mount should have the flexibility to handle various conditions for the camera to work effectively under any circumstances. This flexibility includes the ability to adapt to different manhole placements and capture videos of sufficient quality regardless of the mounting method.

Incorporating features like self-adjusting optics, such as auto-focus to ensure the surface is always in focus, image stabilization to compensate for motion caused by passing vehicles, image noise reduction for low-light situations, and automatic exposure control to prevent overexposure (a common issue with reflective surfaces like biofilm) would benefit image quality. The C70's ability to establish a local connection, which can be accessed using a smartphone, tablet, or other WiFi-enabled devices, is advantageous for monitoring the camera's placement and functionality after installation. Having a high-quality connection with the camera from just outside the manhole to check its status and retrieve footage, especially for long-duration placements, is beneficial for continuous monitoring and processing of the footage. The camera sensor should be designed with minimal room for error when handling it, allowing anyone to mount and retrieve it with min-

imal instruction, possibly limiting the operation to a couple of buttons, allowing for mishaps to happen and not necessarily follow a certain sequence of operation for it to still be working as intended.

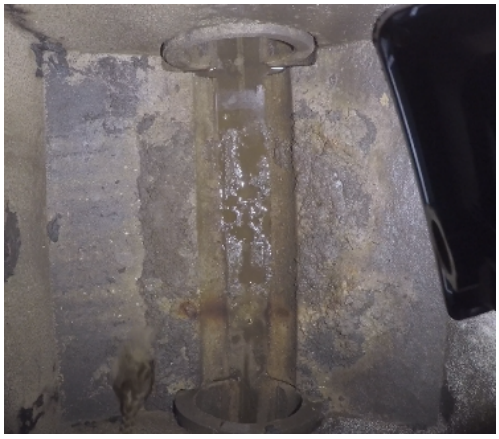
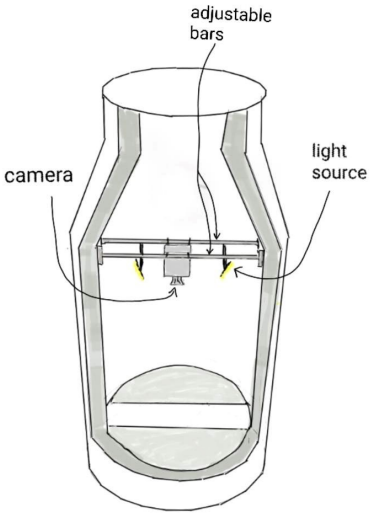
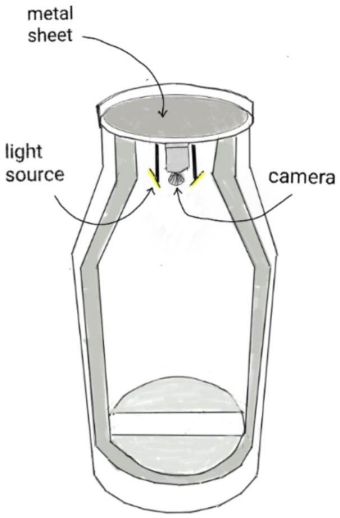


Figure 5.7: This camera view is optimal for the purpose of optical velocimetry implementation in sewers as it can capture the entire flow field. Captured with a GoPro (4K resolution).



(a) Adjustable bars span between the manhole walls with light and camera attached.



(b) Thin metal sheet with light and camera attached underneath. The manhole cover goes on top.

Figure 5.8: Two possible ways of mounting the camera with separate light sources.

5.2.4 Weather Station

Due to practical considerations, the plan to mount a weather station near the field study area was not implemented. Instead, rain data was collected from three measuring stations, with two of them providing consistent enough data to be used in the simulations. However, as all three stations were located within a relatively small area (two within a square kilometre) and still exhibited some deviations, the data cannot be completely trusted. Hydraulic simulations are inherently uncertain, and the accuracy of recreating inflow in the model is further hindered by unreliable rain data. Having a local rain gauge near the camera, collecting precipitation data representative of the camera's proximity would be beneficial. This would enable the assessment of data validity and ensure that data collection is ongoing by allowing regular checks of the gauge's status.

5.2.5 In Field Placement Of The Camera - Significance, verification and backpropagation

The intention when placing the camera was to gather sufficient measurements to determine the presence of upstream illicit connections. However, this goal could not be achieved due to technical issues with the cameras. The original plan was to perform paired tests for all possible combinations and determine if a significant difference was observed, thus confirming the presence of inflow. Unfortunately, with only two comparisons possible from the collected data, and with contradictory results, it is not possible to draw any conclusions based on it. It is more likely that the observed results stem from slight natural variations in the daily inflow series, with some measurements being slightly lower than the previous ones and slightly higher than the subsequent ones, rather than indicating illicit inflow.

Figure 5.9: PTV showed capable of tracking particles in trickling water in an otherwise dry conduit.

The algorithms process the footage without any means of controlling the quality of the output values or verifying the data, except for visually observing how the series plots out with the standard deviation of the frame velocity. Lower deviation, indicating greater agreement among evaluated frames regarding the velo-

city, would suggest higher confidence in the measurement. Some consistency observed between the algorithms and a hint of an emerging pattern with the lower flow in the morning and evening, which aligns with the expectation of reduced activity as people leave offices and stores, suggests that the algorithms produce flow-related velocities and not just velocities at random. However, some peculiar patterns also emerge in the data series. Isolated instances of sudden very high flows or very low flows are unlikely to occur, and if they are measured by only one of the algorithms, it may indicate a weakness of that particular algorithm in dealing with certain flow characteristics or phenomena.

The quality of the data series was evaluated by reviewing the footage and identifying periods of least and most flow to assess whether the algorithms were able to accurately represent both conditions. Three distinct periods in the data stood out (as low flow), occurring during the early morning and late evening, which corresponded to the expected observable pattern. During these periods, the lowest flows were observed, but the conduits never completely dried up. Optical Flow (OF) and Particle Image Velocimetry (PIV) both interpreted the footage as very close to zero flow velocity, while Particle Tracking Velocimetry (PTV) measured velocities as low as 0.04 m/s and captured particles even when the flow was only trickling and barely visible in the footage as in Fig. 5.9. Thus, PTV may be closer to capturing the true velocity. The ability of OF and PTV to produce "no flow" velocities during low flow conditions may not necessarily be problematic, as it may be unrealistic to expect the methods to distinguish velocity changes accurately at such low flow rates anyway. The period of highest flows was only evident as prominent peaks in the PTV series, while PIV and OF seemed to produce similar values for all "larger" flow events, possibly lacking the ability to differentiate between big and slightly larger flows.

All algorithms exhibited little consistency from one value to another in the time series. As there are few sources of inflow upstream from the camera placement, it is likely that inflow from each source is visible as a pulse, causing temporary increases over shorter durations that may coincide with the captured footage. Consequently, the fluctuations in the series may be representative of the actual flow pattern. Throughout the series, there is a correlation between data series from different algorithms during certain parts in terms of showing the same velocity patterns, but there are also sections where they do not co-vary. This can be attributed to the fact that an optimal setup for the camera was not obtained and that the video quality, therefore, did not give the processing algorithms the chance to perform towards their potential. The captured footage from the cameras exhibited significantly poorer quality compared to the experiment and what could be expected, leading to greater variation in the results.

The poor quality footage is believed to make the influence of variation in particle and water flow characteristics greater. Due to the low-quality image, there may be particles that become unrecognizable as particles as in Fig. 5.10, resulting in a lower chance of accurately estimating velocities for the sequence. The blurriness of the image also makes it more challenging to perform pre-processing

and establish algorithm parameters that work consistently every time. Certain algorithms may excel in this low-quality image when there is much continuous matter, while other times, as when the water is clear with fewer particles, it may fail completely. This inconsistency can lead to covariance between series in some parts and no covariance in other parts, as observed among the algorithms. One possible solution to address this issue is to implement the three algorithms in an interconnected manner, always utilizing the one that produces the most consistent results. Alternatively, attempting to find the algorithm that consistently performs best in all conditions could be explored. However, improving the image quality should be a priority before investigating further, as the quality of the results is highly dependent on the analysed footage. Among the algorithms, PTV demonstrated the most promising results by showing its ability to represent the full range of velocities. However, it is limited by the low-quality image, as it often struggles to recognize smeared particles or fails to detect particles due to insufficient resolution of the flow.

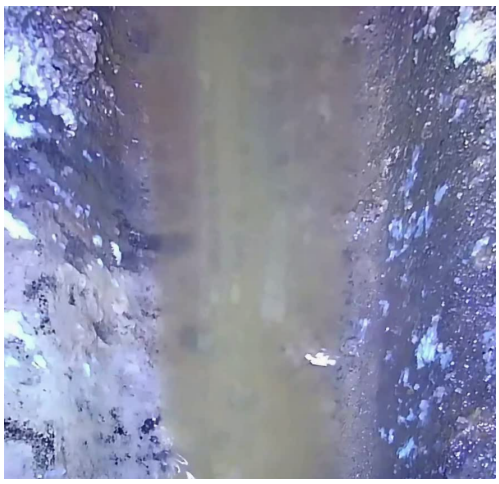


Figure 5.10: Poor video quality in the footage cause smearing of particles making them indistinguishable from flow.

Chapter 6

Conclusions

The implementation of camera vision and Optical Velocimetry (OV) in sewers was tested under favourable conditions during the experiment, where all three algorithms - Particle Tracking Velocimetry (PTV), Particle Image Velocimetry (PIV), and Optical Flow (OF) - successfully represented the flow by producing vectors in the direction of flow and within the expected range of velocities obtained from manual tracking. The experiment demonstrated variations in the algorithms' ability to not produce displacement in footage with no motion, indicating their detection sensitivity specific to the image quality in the experiment. Furthermore, the experiment showed that the average displacement and standard deviation of velocities were generally higher in visible light (VIS) conditions compared to using a combination of ultraviolet light and visible light (UV/VIS), suggesting the potential benefits of using ultraviolet light for OV purposes. It is important to note that these results are specific to the exact footage captured in the experiment, and conclusions about the superiority of a specific algorithm cannot be made considering the factors influencing the result which vary based on camera placement and lighting conditions.

In the field implementation, the captured footage exhibited lower image quality with fewer distinct particles but a higher presence of floating matter. Despite these challenges, all three algorithms were still able to produce meaningful measurements. PTV demonstrated the best capability to represent the range of flows accurately when manually reviewing the footage to verify if the periods of highest and lowest flow were correctly reflected in the time series produced by OV. OF and PIV appeared to be more reliant on the continuous presence of floating matter to generate reasonable velocity estimates, and they struggled to distinguish between very low flow and no flow. On the other hand, PTV showed capable of producing velocity estimates even from individual particles, also performing satisfactorily for bigger continuous matter with fewer distinct features as long as consistent portions of the floating matter were visible over time. All three algorithms are believed to have been restricted by the image quality produced in the field experiment but are still able to produce measurements that could detect illicit inflow in periods where the system is running dry as they are all able to distinguish out higher flow from no flow, by producing low or no velocity.

Both the experimental findings and field implementations of the algorithms have shown that OF and PTV are better candidates for further research. Based on a prior understanding of how PIV works, more specifically in its need to have clusters of particles track, it was anticipated that PIV would not perform well in several aspects, and this expectation was indeed confirmed. Specifically, PIV has been found to produce more spurious vectors overall, as the footage's quality is

not sufficient to obtain the required signal-to-noise (S2N) ratio for validating the displacements. Increasing the S2N ratio while preserving the necessary particle clusters is not easily achievable through pre-processing alone. However, in terms of beneficial pre-processing, more can be done for PTV and OF. Additionally, OF and PTV require much lower computational time compared to PIV. Overall, it seems that PIV is unable to offer anything that the other two methods do not already provide. It should be further investigated to optimize the camera settings and the camera setup in order to obtain better image quality. This may potentially reduce the need for pre-processing in OF, which could be causing a significant overestimation for larger particles. Furthermore, it is important to address the issue of having no possibility of verifying the state of the camera sensor when the PTV produce complete zero measurements.

The ability to detect illicit inflow using significance tests on data collected by the camera sensor, as well as the sensor strategy involving placement, verification, and source location, have not been fully tested yet. It requires the presence of illicit inflow in the data, which can be generated in a controlled experiment by introducing water upstream or occurring naturally. The effectiveness of the sensor placement strategy also needs to be assessed by evaluating its ability to detect illicit inflows and minimize false negatives. Long-term verification of the strategy's success or failure will rely on assessing its results over time in terms of detecting a sufficient number of illicit inflows where it is implemented.

The measurement campaign will be resumed and the camera will continue to be deployed in the field implementation to complete the ongoing measurement campaign and acquire additional data. Additionally, an experiment involving pouring water into the system upstream from the camera will be conducted once the cameras are operational again to verify the method's ability to detect the additional inflow.

Bibliography

Adrian, R. J., & Westerweel, J. (2011). *Particle image velocimetry*. Cambridge university press.

Baker, A., coble paula, p., lead jamie, j., Spencer, R., & Reynolds, D. (2014). *Aquatic organic matter fluorescence*. https : / / doi . org / 10 . 1017 / CBO9781139045452

Beheshti, M., & Sægrov, S. (2019). Detection of extraneous water ingress into the sewer system using tandem methods – a case study in Trondheim city. *Water Science and Technology*, *79*(2), 24–34. https://doi.org/10.2166/wst.2019.057

Beheshti, M., Sægrov, S., & Ugarelli, R. (2015). Infiltration/inflow assessment and detection in urban sewer system. *VANN*, (01), 24–34.

Boutelier, D., Schrank, C., & Regenauer-Lieb, K. (2019). 2-D finite displacements and strain from particle imaging velocimetry (PIV) analysis of tectonic analogue models with tecPIV. *Solid Earth*, *10*, 1123–1139. https://doi.org/10.5194/se-10-1123-2019

Bradski, A., Gary ; Kaehler. (2008). *Learning OpenCV: Computer vision with the opencv library*. O'Reilly Media.

Brunner, G. (1997). *Hec-ras river analysis system: Hydraulic reference manual*. US Army Corps of Engineers, Hydrologic Engineering Center.

Butler, D., & Davies, J. (2000). *Urban drainage* (2nd Edition). Spon Press. https://doi.org/10.4324/9780203244883

Dabiri, D., & Pecora, C. (2019). *Particle tracking velocimetry*. IOP Publishing. https://doi.org/10.1088/978-0-7503-2203-4

Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002). A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, *6*, 182–197. https://doi.org/10.1109/4235.996017

Deshpande, S., & Tallapragada, P. (2018). Particle slip velocity influences inertial focusing of particles in curved microchannels. *Scientific Reports*, *8*. https://doi.org/10.1038/s41598-018-30171-9

Erispaha, A. (2020). *Swmmio documentation*. Retrieved May 15, 2023, from https://swmmio.readthedocs.io/en/latest/

Feng, Y., Goree, J., & Liu, B. (2011). Errors in particle tracking velocimetry with high-speed cameras. *Review of Scientific Instruments*, *82*(5), 82–87. https://doi.org/10.1063/1.3589267

Hartley, R., & Zisserman, A. (2004). *Multiple view geometry in computer vision* (2nd ed). Cambridge University Press.

Horn, B., & Schunck, B. (1981). Determining optical flow. *Artificial Intelligence*, *17*, 185–203. https://doi.org/10.1016/0004-3702(81)90024-2

Jeanbourquin, D., Sage, D., Nguyen, L., Schaeli, B., Kayal, S., Barry, D. A., & Rossi, L. (2011). Flow measurements in sewers based on image analysis: automatic flow velocity algorithm. *Water Science and Technology*, *64*(5), 1108–1114. https://doi.org/10.2166/wst.2011.176

Kim, Y.-S., Yang, S.-K., Yu, K., & Kim, D. (2014). Flood runoff calculation using disaster monitoring CCTV system. *Journal of Environmental Science International*, *23*. https://doi.org/10.5322/JESI.2014.4.571

Ku, H. H. (1966). Notes on the use of propagation of error formulas. *Journal of Research of the National Bureau of Standards. Section C: Engineering and Instrumentation*, *70C*(4).

Legleiter, C. J., & Kinzel, P. J. (2021). Surface flow velocities from space: Particle image velocimetry of satellite video of a large, sediment-laden river. *Frontiers in Water*, *3*. https://doi.org/10.3389/frwa.2021.652213

Liberzon, A., Lasagna, D., Aubert, M., Bachant, P., Käufer, T., jakirkham, 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* (Version 0.22.2). Zenodo. https://doi.org/10.5281/zenodo.3930343

Løfald, A. N. (2022). *Specialization project; wastewater sensors and sensor placement strategies* [Unpublished manuscript].

Løvås, G. G. (1999). *Statistikk - for universiteter og høgskoler*. Universitetsforlaget.

McDonnell, B. E., Ratliff, K., Tryby, M. E., Wu, J. J. X., & Mullapudi, A. (2020). Pyswmm: The python interface to stormwater management model (swmm). *Journal of Open Source Software*, *5*(52), 2292. https://doi.org/10.21105/joss.02292

Meier, R., Tscheikner-Gratl, F., Steffelbauer, D., & Makropoulos, C. (2022). Flow measurements derived from camera footage using an open-source ecosystem. *Water*, *14*, 424. https://doi.org/10.3390/w14030424

Mishra, S., Tripathy, H. K., Mallick, P. K., Sangaiah, A. K., & Chae, G.-S. (2022). *Cognitive big data intelligence with a metaheuristic approach*. Academic Press.

Moreno-Rodenas, A., Duinmeijer, A., & Clemens, F. (2021). Deep-learning based monitoring of fog layer dynamics in wastewater pumping stations. *Water Research*, *202*, 117482. https://doi.org/10.1016/j.watres.2021.117482

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*, 885–900. https://doi.org/10.5194/hess-25-885-2021

NetworkX documentation. (2023). *Networkx — networkx documentation*. Retrieved May 15, 2023, from https://networkx.org/documentation/stable/release/release_3.1.html

OpenCV manual. (2022a). *Basic concepts of the homography explained with code*. https://docs.opencv.org/4.x/d9/dab/tutorial_homography.html

OpenCV manual. (2022b). *Camera calibration with square chessboard*. https://docs.opencv.org/4.x/dc/d43/tutorial_camera_calibration_square_chess.html

OpenCV manual. (2022c). *Camera calibration with square chessboard*. https://docs.opencv.org/4.x/dc/d43/tutorial_camera_calibration_square_chess.htmll

OpenCV manual. (2022d). *Contours : Getting started*. https://docs.opencv.org/4.x/d4/d73/tutorial_py_contours_begin.html

OpenCV manual. (2022e). *Cv.farnebackopticalflow*. https://docs.opencv.org/4.x/de/d9e/classcv_1_1FarnebackOpticalFlow.html

OpenCV manual. (2022f). *Image thresholding*. Retrieved March 15, 2023, from https://docs.opencv.org/4.x/d7/dd0/tutorial_js_thresholding.html

OpenCV manual. (2022g). *Motion analysis*. https://docs.opencv.org/4.x/de/de1/group__video__motion.html

OpenCV manual. (2022h). *Smoothing images*. https://docs.opencv.org/4.x/dd/d6a/tutorial_js_filtering.html

OpenCV manual. (2022i). *Videocapture class reference*. https://docs.opencv.org/3.4/d8/dfe/classcv_1_1VideoCapture.html

OpenCV manual. (2022j). *Videowriter class reference*. https://docs.opencv.org/4.x/dd/d9e/classcv_1_1VideoWriter.html#details

Ostfeld, A., Uber, J. G., Salomons, E., Berry, J. W., Hart, W. E., Phillips, C. A., Watson, J.-P., Dorini, G., Jonkergouw, P., Kapelan, Z., di Pierro, F., Khu, S.-T., Savic, D., Eliades, D., Polycarpou, M., Ghimire, S. R., Barkdoll, B. D., Gueli, R., Huang, J. J., … Walski, T. (2008). The battle of the water sensor networks (BWSN): A design challenge for engineers and algorithms. *Journal of Water Resources Planning and Management*, *134*(6), 556–568. https://doi.org/10.1061/(ASCE)0733-9496(2008)134:6(556)

Raffel, M., E. Willert, C., T. Wereley, S., & Kompenhans, J. (2007). *Particle image velocimetry: A practical guide* (2nd). Springer.

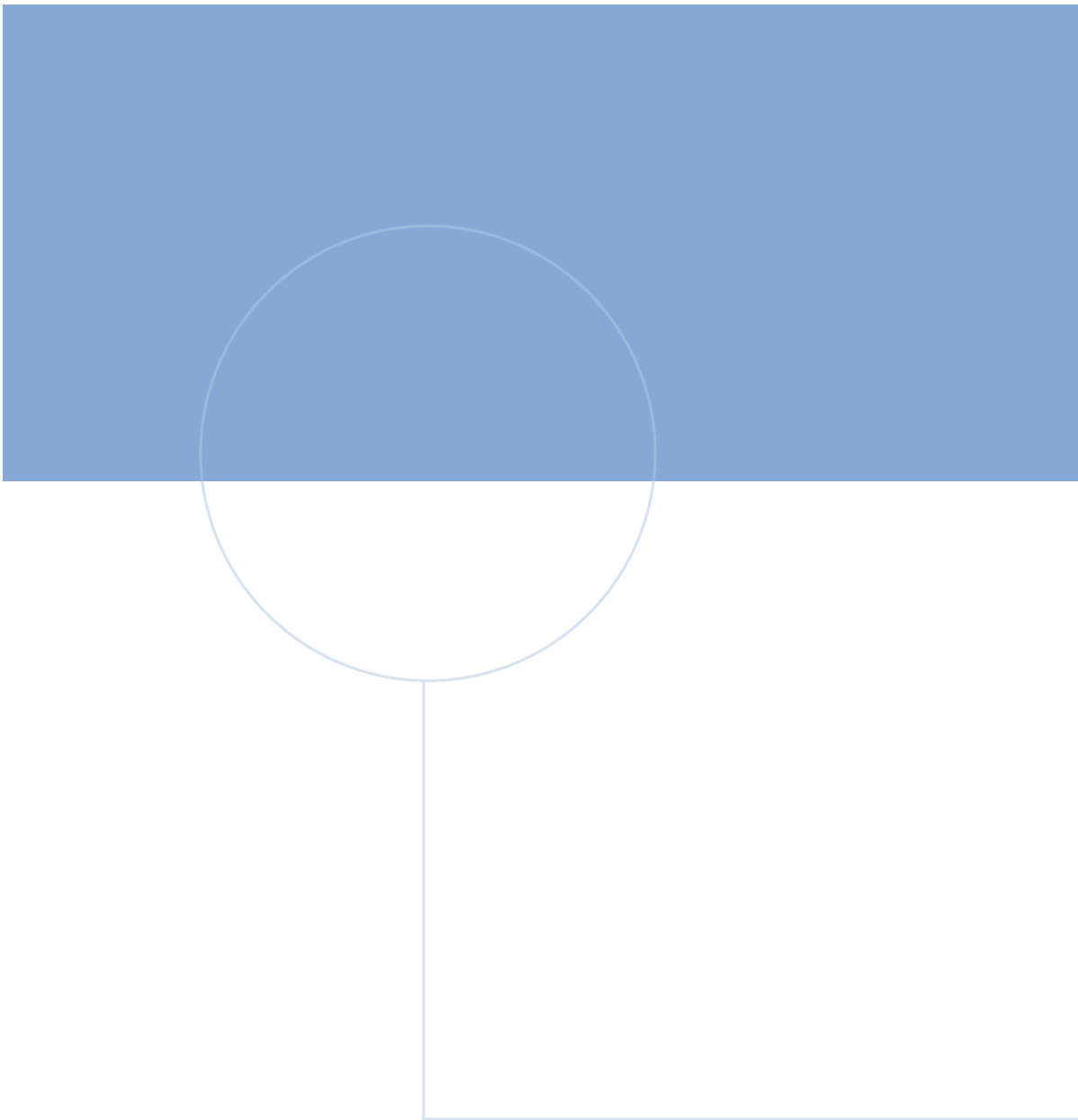
Sadeghikhah, A., Ahmed, E., & Krebs, P. (2022). Towards a decentralized solution for sewer leakage detection – a review. *Water Science and Technology*, *86*(5), 1034–1054. https://doi.org/10.2166/wst.2022.263

Scarano, F. (2001). Iterative image deformation methods in PIV. *Measurement Science and Technology*, *13*, R1–R19. https://doi.org/10.1088/0957-0233/13/1/201

Simon, G., Vakulya, G., & Rátosi, M. (2022). The way to modern shutter speed measurement methods: A historical overview. *Sensors*, *22*, 1871. https://doi.org/10.3390/s22051871

Ufuoma, G., Sasanya, B. F., Abaje, P., & Awodutire, P. (2021). Efficiency of camera sensors for flood monitoring and warnings. *Scientific African*, *13*, e00887. https://doi.org/https://doi.org/10.1016/j.sciaf.2021.e00887

Zeng, R., Mannaerts, C. M., & Shang, Z. (2021). A low-cost digital colorimetry setup to investigate the relationship between water color and its chemical composition. *Sensors*, *21*(20). https://doi.org/10.3390/s21206699



NTNU

Norwegian University of
Science and Technology