



HAL
open science

Building a Wastewater Network Graph from Inspection Videos

Minh-Thu Tran-Nguyen, Salem Benferhat, Nanée Chahinian, Carole Delenne,
Thanh-Nghi Do

► **To cite this version:**

Minh-Thu Tran-Nguyen, Salem Benferhat, Nanée Chahinian, Carole Delenne, Thanh-Nghi Do. Building a Wastewater Network Graph from Inspection Videos. ISDS 2024 - 2nd International Conference on Intelligent Systems and Data Science, Nov 2024, Nha Trang city, Vietnam. pp.188-202, 10.1007/978-981-97-9613-7_14. hal-04783710v2

HAL Id: hal-04783710

<https://hal.science/hal-04783710v2>

Submitted on 4 Feb 2025

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.



Distributed under a Creative Commons Attribution - NonCommercial 4.0 International License

Building a wastewater network graph from inspection videos

Minh-Thu TRAN-NGUYEN¹, Salem BENFERHAT²,
Nanee CHAHINIAN³, Carole DELENNE³, and Thanh-Nghi DO¹

¹ College of Information Technology

Can Tho University, 92000-Cantho, Vietnam

² CRIL, CNRS UMR 8188, Universite d'Artois, France.

³ HSM, Univ Montpellier, CNRS, IRD, Montpellier, France.

tnmthu@ctu.edu.vn, benferhat@cril.fr, nanee.chahinian@uird.fr,
carole.delenne@umontpellier.fr, dtngghi@ctu.edu.vn

Abstract. The management of wastewater networks is essential to ensure optimal hydraulic modeling. However, data on wastewater systems, usually represented in Geographic Information Systems (GIS), are often incomplete. Other sources of information are thus required to complete them. Among these, pipe inspection videos, so far mainly used for maintenance and detection of anomalies such as degradation, can be used for this purpose. This article goes in this direction and proposes to take advantage of these inspection videos to extract the structure of wastewater networks, represented here in the form of graphs. For this, we propose automatic detection mechanisms of the different annotations associated with the inspection videos. These annotations contain enough information to compose a graph of the wastewater networks. Regular expressions and text recognition tools, based on optical character recognition (OCR), are used to extract manholes identifiers, water flow direction in pipelines, as well as other useful information to determine the geographical positions of wastewater network objects. Experimental results on real data show the good performance of our method.

Keywords: wastewater inspection, · Deep Learning, · Optical Character Recognition.

1 Introduction

Wastewater systems are essential elements of urban infrastructure. Therrien et al. [12] carried out a critical analysis of raw data related to wastewater networks. With the increasing volume of data being collected – particularly generated by sensors – wastewater network professionals are facing difficulties in analyzing these vast quantities of data and identify network anomalies. In addition, they highlighted that machine learning is a valuable tool for overcoming these difficulties, and thus helping sewerage system professionals to make effective use of the collected data [12].

For several decades, industry and universities have been conducting research

into the automated inspection of wastewater pipes. This work has led to the development of various robotic platforms and specialized algorithms for collecting and analyzing data on wastewater network pipes. Closed-circuit television videos (CCTV) inspection for the preventive maintenance of wastewater pipes is often carried out in two phases: an on-site video data collection phase, and an off-site analysis, annotation and evaluation phase. In the first phase, different types of machines are used to examine the pipes and collect data. These videos are then inspected and often annotated to identify any information that might be useful for maintenance purposes.

As indicated below in the related section, most studies conducted on wastewater pipe inspection videos have focused on object detection from the images present in the videos (e.g., [8,5,14,11,3,7,8,10,1,13]). The problems addressed are mainly of the automatic classification type for detecting anomalies in wastewater pipes. However, many inspection videos also contain text annotations. There are very few studies (e.g., [4,11]) that extract these annotations from a perspective of diagnosis, classification or determination of the structures of sanitation networks. However, existing work on anomaly detection or textual annotation extraction does not allow us to build wastewater network graphs from inspection videos.

The work presented in this paper also focuses on the automatic exploitation of pipe inspection videos, and in particular on the automatic extraction of textual annotations from real wastewater pipe data. However, we adopt a different direction, in which we are not interested in detecting anomalies in pipes. Our objective is to build a graph of the wastewater network. The resulting graph will have manholes as nodes and pipes as edges. Each video will provide information on two manholes and one pipe in the wastewater network. The constructed graph will then be useful to enrich Geographic Information Systems for wastewater networks. Indeed, a major problem in such Geographic Information Systems (GIS) is data incompleteness, that can refer to the absence of values for certain attributes such as the flow direction in a pipe. Our study will fill this gap by constructing a graph oriented in the direction of the flow. Incompleteness can also result from the non-information of certain pipes or manholes in GIS. The graph we are constructing enables us to supplement GIS with nodes and lines extracted from videos, but not reported in current Geographic Information Systems.

Figure 3 gives an example of the pipe inspection video¹ we are working on. Text annotations are generally (but not always) distributed across the four corners of each video frame. These annotations contain summary information about the inspection process. They include manhole identifiers, direction of inspection, direction of wastewater flow, distance travelled, date of inspection, name of the street where the pipe is located, etc. With almost 900 videos from a small town

¹ We are grateful to "Montpellier Méditerranée Métropole" for providing us with the data needed for this research activity.

near Montpellier (Prades-le-Lez), this graph will be large enough to complement and compare with the data present in the existing Geographic Information System. However, the task of graph construction is not always an easy one. In some frames of the inspection videos, the text is clear and easily detectable by text recognition tools (such as OCR), while in others, the text is difficult to detect. This is mainly due to the fact that some pipes have a bright, clean background, making annotations easier to detect, while others have a dark background, drowning out the text annotations. Added to this difficulty is the fact that the location of the annotations does not always respect the usual positioning. To overcome these drawbacks, we take advantage of the fact that certain annotations remain sufficiently identical over a large number of frames in a same inspection video. This is valid for the street name, manhole identifiers, direction of inspection and direction of wastewater flow. As a result, we can, thanks to fusion strategies (even basic), determine what is the most plausible text associated with the annotations. Regarding the extraction of manhole identifiers, we also use regular expressions to compensate some detection imperfections of OCR type tools. The experimental results obtained using real wastewater inspection videos show very good detection results.

The rest of the document is organised as follows. Next section provides related work. It then describes the contributions of this work in two main sections. Section 3 briefly presents the algorithm used to extract the pipe names from all videos of the investigated region to build a graph associated with a wastewater network. Section 5 presents the experimental results, followed by the conclusions and future works in section 6.

2 Related work

In 2022, Li et al. reviewed the methodologies, datasets and problems associated with wastewater network pipes since the year 2000 [8]. There are, however, very few public datasets. An open source dataset on sewer pipe anomalies called Sewer-ML [5] exists since 2021, and contains 1.3 million images from 75,618 videos collected by three Danish water companies over a nine-year period. The images have the structure shown in figure 1, and each (anonymized) image has one or more of a total of 17 annotated defect classes. These images are then analyzed by machine learning tools using convolutional neural network (CNN) architectures to automatically detect anomalies in sewer pipes [5,14,11,3,7,8,10].

In addition to the open-source Sewer-ML dataset, authors Biswas et al. [1] also processed data from 221 German municipalities, including 10,205 video files. From these videos, they extracted 156,028 images representing various wastewater pipe anomalies. Figure 1 shows a representative example of the image structure (e.g., video frame) used in [1]. The anomalies of wastewater pipes were classified according to the standard inspection codes used in Germany for wastewater pipe maintenance. The same approach is also used in the database we have processed in this paper. In their study [1], 25 different categories of anomalies

are used, but the distribution of instances of these categories is highly unbalanced. Six different classifiers were employed, with significantly varying numbers of trainable parameters, in order to better identify each anomaly category. They evaluated their models with the Sewer-ML dataset [5], obtaining very satisfactory results.

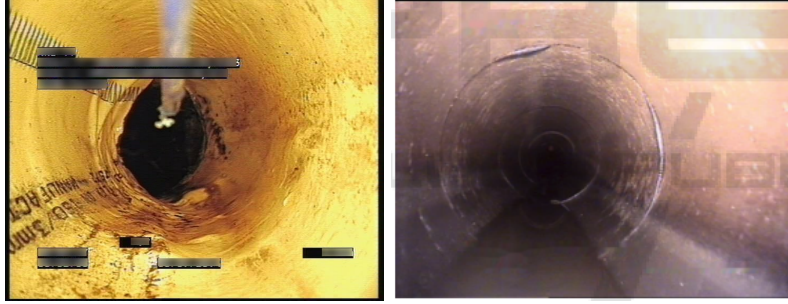


Fig. 1: Sewer images were collected in Denmark (in the Sewer-ML dataset) [5] and in Germany [1].

In [4], the authors collected textually annotated videos of wastewater pipe inspections from the Korea Institute of Construction Technology, containing eight types of information: pipe distance, pipe number, survey date, survey time, sewer pipe size, sewer pipe type, exploration start and end locations, and pipe direction (see an example of annotated video in Figure 2). In addition to using convolutional neural networks for the problem of automatically classifying pipe anomalies, the authors also exploited the annotations associated with inspection videos. These annotations were automatically extracted using text recognition tools and used to determine the distance between the starting point and the location of the fault. The results showed slight differences between the automatically extracted distance and the actual distance. Although the subtitles include the current pipe code and pipe direction, this information is not sufficient to create a sewer pipe graph for wastewater network management purposes. To create such a graph, it was essential to have the pipe direction between the two extremities of each pipe. In a similar work, the authors of [11] have proposed mechanisms for detecting sewer pipe anomalies, as well as for recognizing textual location information from CCTV videos. The improved YOLOv5 [6] model was used for text detection, after noise removal. The resulting model was able to extract, with a high detection rate, two key pieces of information from the subtitles: the pipe identifier and the distance travelled.

The existing work presented above on the automatic extraction of textual annotations is geared towards anomaly detection, and localization of these anomalies in wastewater pipes. However, existing work does not allow the construction of wastewater network graphs.

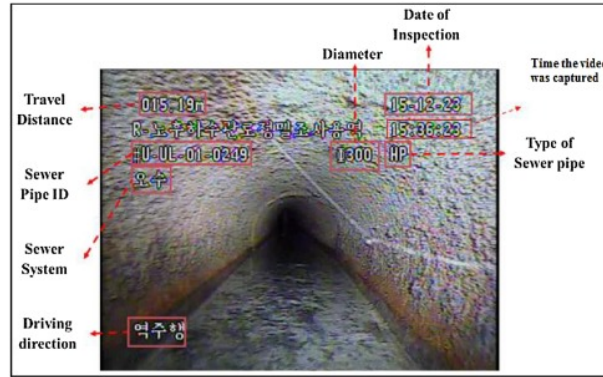


Fig. 2: Sewer images were collected in Korea [2].

3 Description of the wastewater pipe inspection video dataset

In this section, we first describe the data used in this paper. These data were obtained from “Montpellier Méditerranée Métropole”, a public organisation responsible for water management throughout the metropolis of Montpellier, located in the south of France. Different portions of the network were studied, and for each, videos of wastewater pipe inspections were taken. In addition to these videos, PDF reports were created, containing information such as details of wastewater pipe anomalies: positions of anomalies, codes and names associated with anomalies, as well as maps showing the positions of manholes, pipes and manhole identifiers.

The data from the inspection videos of sewage pipes that we use are all annotated. Figure 3 provides a typical example of the structure of textual annotations in a pipe investigation video. The textual annotations are typically (but not always) positioned in the four corners of the image, forming four blocks of texts, each providing different information about the inspection of the wastewater pipelines. The annotations do not diagnose anomalies in the inspection pipes but rather provide factual information about the inspection process. More precisely, textual annotation present in the inspection videos contain the following information:

- The first block of texts contains information about manholes and directions of wastewater pipes. This block, which is typically located at the top left of the video frames as indicated in Figure 3, is the most useful in our study as it mainly allows us to construct the graph. It provides information about:
 - The manhole identifiers.
 - The direction of inspections, represented by the identifier of the starting manhole followed by the identifier of the ending manhole. It indicates the direction in which the robot investigating the wastewater in the pipeline

moves. It is recognizable by the keyword "Inspect." (or "Inspection"), followed by the identifiers of the two (departure and arrival) manholes which are separated by a delimiter (here "_").

- The wastewater flow, represented by the identifier of the starting manhole followed by the identifier of the ending manhole. It has the same structure as the one indicating the direction of inspections, except that the keyword "Ecoule." (or "Ecoulement") is used instead of "Inspect."
- A second block of text (usually in the bottom left-hand corner of video frames, as in Figure 3) contains the Name of the city and street where the manholes and pipe are located. This block of text may be useful for determining the geographical positions of the manholes and wastewater pipes at a later stage.
- A third block contains the date and time of the pipe inspection and is usually located in the bottom right-hand corner of video frames, as shown in Figure 3. The format is not always the same from one video to another. However, this block of text is not decisive for this work and for the perspective of building a wastewater network graph.
- A last block (located in the top right-hand corner of video frames, as shown in Figure 3) provides information on the distances covered by the robot. This information may be relevant for locating anomalies, but it can also be used to pinpoint the geographical positions of manholes and wastewater pipes.

The name and flow direction of the sewer pipe, as well as the name of the town, the date, and the street being investigated, remain unchanged throughout the video. However, the investigation time and distance from the beginning of the video obviously change. Another observation is that the direction of wastewater flow is not always the same as the direction of the robot's investigation.

In the rest of the paper, we focus on the automatic detection of the first text block, as this is the one that will be useful for graph construction. We don't go into detail about the detection of elements in the other blocks; however, we will comment briefly on them at the end of the paper.

4 Building a graph of sewer pipes from video investigations

4.1 Presentation of the problem and issues to be addressed

Let us start by describing the problem we wish to address. Our input is a set of videos of wastewater pipe inspections. These videos are annotated as described in the previous section.

From this set of inspection videos, we wish to construct a graph such that :

- The nodes of the graph are composed of all the manholes present in the videos. Each node of the graph will simply be named using the identifier of the manhole found in the videos.

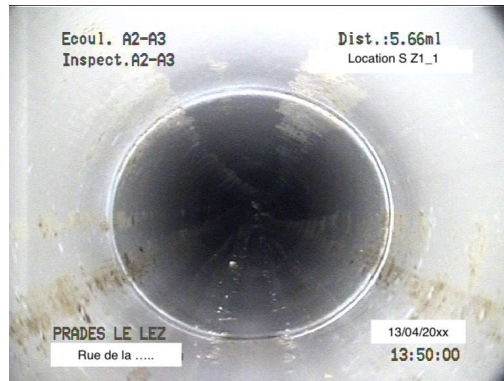


Fig. 3: Example of inspection video.



Fig. 4: The quality of annotation in investigated video.

- An edge is defined in the graph, as soon as we have a frame of a video which indicates a flow (with the keywords "Ecou1." or "Ecoulement") between two wastewater pipe manhole identifiers.

If the videos are of superior quality and the text recognition tools (of the OCR type) are precise then the problem we are dealing with is relatively easy to manage. Indeed, a naive algorithm consists in first searching all the textual information of the first block (described in the previous section) in each video frame. Then, extract all the manhole identifiers, and insert a link between two manholes as soon as the keyword "Ecou1." (or "Ecoulement") is detected. In practice, image-based text recognition systems, such as those based on OCR tools, are not always accurate. Consequently, it is not surprising that manhole identifiers recognized from one video frame may differ from those recognized from another video frame, even if both frames concern the same wastewater pipe inspection video and the same OCR tool is used for both frames. This is problematic for the construction of our graph, and we need to define decision rules for choosing the identifier of the manholes, associated with a video, to put as nodes in the graph.

In addition to the inaccuracy of OCR tools, video images of wastewater pipe inspections are not of very good quality. Indeed, video images are often blurry, dark and unclear. This can make the extraction and identification of the manholes impossible. In Figure 4, on the video image extracted at 13:53:13, the names of the manholes are readable (and therefore a priori easy to extract). However, on the video frames extracted at 13:50:09 (or again at 13:53:49), it is difficult to read the manhole identifiers (and if they are extracted, they will probably be wrong). Therefore, it is important to define, on one hand, the set of video frames that will be used to extract manhole identifiers (along with other information such as sewer pipe flow directions, street names, etc.), and on the other hand, to define fusion rules to aggregate the OCR results from each of the video frame considered.

It is in this spirit that the algorithms described in the following section are designed; where we recall that we only focus on the extraction of manhole identifiers as well as the direction of wastewater flows (the extraction of other elements follows the same process).

4.2 Algorithms

Algorithm 1 outlines our method for constructing the graph associated with a set of wastewater pipe inspection videos. The input of the algorithm is first this set of videos, which is denoted by \mathcal{V} . Our algorithm also takes as input a parameter which is a string of characters representing the list of keywords (each separated by a semicolon ";") that will be used to identify the line of text containing the manhole identifiers, as well as the direction of wastewater flow. This string is simply noted as *str* in our algorithm. In the video examples used in this paper, *str* is initialized when calling the algorithm 1 by "Ecoule.;Ecoulement".

The output of our algorithm is a directed graph, denoted by $\mathcal{G} = (L, E)$ where L represents the set of nodes in the graph and E the set of arcs. Both sets are initialised to the empty set (line 2 in the Algorithm). We recall that the names of the nodes are defined as those of the manholes identifiers in the videos. We also remind readers that the arcs represent the flow of a pipe linked by two manholes in the graph. These two nodes will be denoted by θ_S (start) and θ_E (end) in our algorithm. For each video, the detected θ_S and θ_E nodes are added to the set of graph nodes (line 8 of our algorithms) and the pair (θ_S, θ_E) is added to the set of graph arcs (line 9 of the algorithm).

Our algorithm then consists of a loop that parses each of the videos in the set \mathcal{V} . Each video provides a piece of information about the pipe linked to two manholes. In other words, for each video, denoted v , our algorithm will extract two manhole identifiers (i.e. two graph nodes) and the direction of wastewater flows (i.e. an arc between the two extracted nodes).

The first step in the "for" loop of the algorithm is to retrieve a set of textual annotations associated with each wastewater pipe inspection video. We use an OCR type text recognition tool as a basic tool to accomplish this task [9]. Now, as we mentioned above, the OCR tools are not accurate and the frames are not all

of good quality. Besides, for practical and computational considerations, it is not possible (or even useful) to take the textual annotations from each frame image of the videos. In fact, in general, there is no major difference between two frames that follow each other. Therefore, it is important to define a function, which we have named "Select-annotations", that selects subset of the Image frames that will be retained before extracts the texts they contain. All the texts returned by this function will be stored in a variable noted ω in line 4 of Algorithm 2.

Algorithm 1: Algorithm for building a graph from pipe inspection videos

Input :

- \mathcal{V} : a set of pipe inspection videos,
- str : a character string containing a list of keywords used to locate manhole identifiers in blocks of text.

Output:

- $\mathcal{G} = (L, E)$: a directed graph generated from all wastewater pipe inspection videos:
 - L : a set of nodes (representing manholes)
 - E : a set of arcs (representing wastewater flow)

```

1 begin
2    $L, E \leftarrow \emptyset$ 
3   for  $v \in \mathcal{V}$  do
4      $\omega \leftarrow \text{Select-Annotations}(v, 4, 50, 25)$ 
5     # Select-Annotations is given in Algorithm 2
6      $\theta_S, \theta_E \leftarrow \text{Extract-manholes}(\omega, str)$ 
7     # Extract-manholes is given in Algorithm 3
8      $L = L \cup \{\theta_S, \theta_E\}$ 
9      $E = E \cup \{(\theta_S, \theta_E)\}$ 
10  end
11  Return  $\mathcal{G}$ 
12 end
```

Now let us detail the parameters and content of function *Select-annotations* (see Algorithm 2). This function presents how annotations can be extracted from video frames. It accepts a wastewater pipe inspection video as the first obvious parameter. To avoid processing frame images that are too close in time (those that follow each other, for example), we use a second parameter, denoted f_s , which represents the "step" with which we progress through the video frames, starting with frame number "0", then " $0+f_s$ ", " $0+2*f_s$ ", etc., until we have covered the whole video. When $f_s=1$, this means we take all the frames in the video. The case of $f_s=0$ is suitable for very short videos (just a few seconds). In practice, the parameter f_s is determined by the length of the video (and

the computing capacity of the machines used), and the name of the variable f_s actually stands for the number of frames per second (we took as an example the value $f_s=25$). Another parameter, denoted ψ , is the maximum number of frames we consider sufficient to make a decision. Note that manhole identifiers do not change throughout the video. In fact, it is the quality of the video images from one frame to the next that can lead to different detection results. This parameter (or a threshold) is mainly used when the video size is very large (otherwise the f_s parameter may suffice). Finally, a last parameter, denoted by β , is used to indicate the minimum number of text blocks required to consider that an image frame can be retained for the extraction of manhole identifiers. In our case studies β can be equal to 4, meaning that if a text recognition tool (OCR), detects less than 4 text blocks we can consider that the frame image is not good qualities. If we set $\beta=0$, then all frame images are considered.

Algorithm 2: Select-annotations(v, β, ψ, f_s)

input : v : current video, β : threshold of extraction, ψ : threshold of observation frames, f_s : frame per second;

output:

ω : a set of text annotations

```

1 begin
2    $i \leftarrow 0$  # number of the current frame in the video
3    $k \leftarrow 0$  # number of already processed frames
4    $\omega \leftarrow \emptyset$ 
5   while  $(k < \psi) \wedge (i \bmod f_s = 0)$  do
6      $f \leftarrow \text{getFrame}(v)$ 
7      $\mathcal{L} \leftarrow \mathcal{D}_{\text{DocTR}}(f)$  #  $\mathcal{D}_{\text{DocTR}}$  : is a function to extract annotation in  $f$ 
8     if  $|\mathcal{L}^B| \geq \beta$  then
9        $k \leftarrow k + 1$ 
10       $\omega \leftarrow \omega \cup \mathcal{L}$ 
11    end
12     $i \leftarrow i + f_s$ 
13  end
14  return  $\omega$ 
15 end

```

The set of selected text annotations returned by the *Select-annotations* function is stored in a variable called ω . Based on these parameters, the *Select-annotations* function scans the frame images using f_s steps. For each frame, the OCR text extraction tool is used. If the number of text blocks extracted from this frame is greater than or equal to β , then these text blocks are added to ω . Otherwise, the frame is considered of poor quality and will not be retained for the next step. The algorithm stops when there are no more frames to process, or the number of frames already processed reaches the threshold represented by the parameter ψ . Once the annotations have been extracted from a video, it is

necessary to extract the identifiers of the two manholes or graph nodes (noted θ_S and θ_E in the algorithm) associated with the video and the orientation of the wastewater pipe. This is step 6 of Algorithm 1. These two nodes are then added to the L set of nodes being constructed (line 8), as is the (θ_S, θ_E) arc, which is also added to E (line 9).

The “Extraction-manholes” function is described and commented in Algorithm 3. This function first takes as input the string ω , which represents all the texts of all the frame image annotations selected by the “Selection-annotations” function given in Algorithm 2. The principle of the “Extraction-manholes” function is to use an initial regular expression to locate the lines containing the manholes identifiers. A second regular expression is used to extract all start and end manholes. Finally, a majority rule is applied to determine the final start and end manholes to be associated for a given video.

Algorithm 3: Extract-manholes (ω, str, reg_m)

input :

- ω : a set of textual annotations,
- str : a list of keywords to locate manhole identifiers,
- reg_m : a regular expression for manholes identifiers.

output:

- θ_S, θ_E : identifiers of start and end manholes

```

1 begin
2    $L_S, L_E \leftarrow \emptyset$  # list of pairs  $(\varphi, k_i)$  where:
3   #  $\varphi$  are manholes identifiers (start and end)
4   # and  $k_i$  represents their occurrence in the  $\omega$  string.
5   for  $\alpha \in \omega$  do
6     #  $\alpha$  is set of texts present in  $\omega$  and associated with frame images
7      $\beta \leftarrow \text{Regular\_expression}(\alpha, str)$  #returns the line that contains str in  $\alpha$ 
8      $\varphi_S, \varphi_E \leftarrow \text{GetManholeName}(\beta, reg\_m)$ 
9     If  $(\varphi_S, k_i) \in L_S$ 
10      Replace  $(\varphi_S, k_i)$  by  $(\varphi_S, k_i + 1)$  in  $L_S$ 
11     Else
12      Add  $(\varphi_S, 1)$  to  $L_S$ 
13     If  $(\varphi_E, k_i) \in L_E$ 
14      Replace  $(\varphi_E, k_i)$  by  $(\varphi_E, k_i + 1)$  in  $L_E$ 
15     Else
16      Add  $(\varphi_E, 1)$  to  $L_E$ 
17   end
18    $\theta_S \leftarrow \varphi_S$  such that  $(\varphi_S, k_i) \in L_S$  and  $k_i$  is maximum
19    $\theta_E \leftarrow \varphi_E$  such that  $(\varphi_E, k_i) \in L_E$  and  $k_i$  is maximum
20   return  $\theta_S, \theta_E$ 
21 end
```

5 Experimental results

The graph construction algorithm presented in the previous section has been implemented using the Python language. The experimental results presented in the following sections were performed on a computer with the following configuration: MasOS Sonoma 14.5, Apple M1, 8 GB RAM. To extract texts from video frames, we used the DocTR library [9]. In DocTR, the pre-trained ResNet50 architecture is used for text detection and the pre-trained VGG16 architecture is used for text recognition. The NetworkX library is used to create a directed graph, which is saved in JSON format. Graphs generated by our algorithm are also exported in Graph Json format.

5.1 Dataset description

The real data used for evaluating our work was collected for the town of Prades-le-Lez, Montpellier. In total, wastewater pipe inspection video data from 27 locations of Prades-le-Lez were collected. Each location has an average of 30 inspection videos. The videos have a resolution of 352x258 pixels. For this research, we selected 7 locations, described in table 1, for experimentation and evaluation. The smaller locations contain 14 videos, while the larger location contain 58 videos.

Table 1: Dataset description

No	Region	#video	Travel distance
1	Region P Z1 - 1	49	1358.64m
2	Region P Z1 - 2	32	1029.90m
3	Region P Z1 - 3	58	1337.79m
4	Region S Z1 - 1	40	1034.76m
5	Region S Z1 - 2	33	1002.46m
6	Region S Z1 - 3	14	570.25m
7	Region S Z2 - 1	37	942.05m

Typically, a video has 25 to 30 frames per second. Figure 5 shows the distribution of video duration in the used video dataset. Most videos are less than 5 minutes long, while around 20% are longer than 5 minutes. Therefore, in this research, annotations are extracted every 25 frames for long videos. Since most videos last less than 5 minutes, we used 50 frames from each video for our search. Using OCR and regular expressions, we extracted information on manholes (including direction and sewage flow), streets, dates and times. It takes about a minute to extract annotations from a video. The time needed to obtain manhole names and construct a graph for a 40 minute video is less than a minute.

Based on the list of outgoing and incoming manholes detected from the wastewater pipe inspection videos, the structure of the wastewater networks is represented in graph form as shown in figure 6. Manholes are represented by

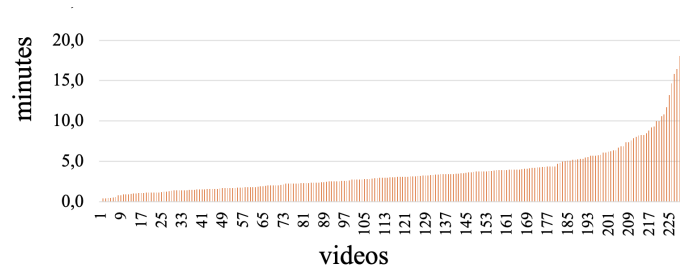


Fig. 5: Duration of investigated videos

nodes, and arrows indicate the direction of wastewater flow in a pipe connected to two manholes. This directional information is important as it is considered one of the missing values in Geographic Information Systems for wastewater networks.

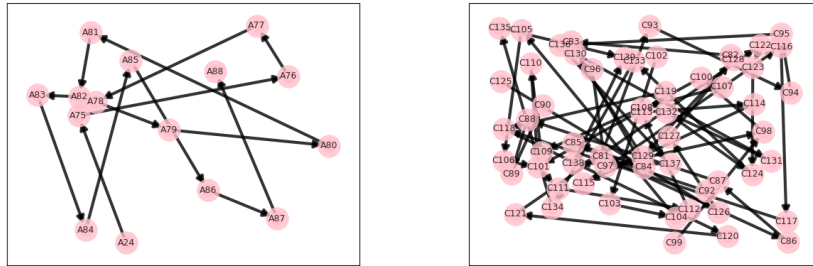


Fig. 6: Graph of manholes from investigated videos

In terms of overall accuracy, the results in table 2 clearly show that our algorithm is efficient, with a detection rate close to 100%. It's important to note that even if the videos are processed individually and independently, the overall graph (associated with all the videos) is totally connected, with no isolated nodes. This is further confirmation that the detection of the manhole identifiers (and the orientations of pipes) was very successful.

6 Conclusion and future works

In this paper, we have presented an algorithm for extracting the structure of a wastewater network from a set of wastewater inspection videos. This structure is represented as a directed graph of the wastewater pipes, constructed automatically from the annotations present in the sewer videos. Numerical test results on

Table 2: Accuracy of manhole name extraction

No	Region	Videos	Start manhole		End manhole	
			DocTR	Proposed Method	DocTR	Proposed Method
1	Region R Z1 - 1	49	100%	100%	100%	100%
2	Region R Z1 - 2	32	100%	100%	100%	100%
3	Region R Z1 - 3	58	96,6%	100%	100%	100%
4	Region S Z1 - 1	40	95,0%	97,5%	97,5%	97,5%
5	Region S Z1 - 2	33	90,9%	97,0%	97,0%	97,0%
6	Region S Z1 - 3	14	92,9%	92,9%	100%	100%
7	Region S Z2 - 1	37	100%	100%	97,3%	100%

the Prades-le-Lez dataset show that our method delivers very good detection results. In this paper, we have focused on the detection of manhole identifiers and the orientation of wastewater flows. Other annotations (duration, time, street names, etc.) are also detected and follow the same principles, whether for the selection of frame images to be considered or for the use of regular expressions.

Future work includes the determination of geographic coordinates from street names and distances between manholes. We then plan to compare these georeferenced data with the GIS data we have also collected.

Acknowledgments This research has received support from the European Union’s Horizon research and innovation programme under the MSCA (Marie Skłodowska-Curie Actions)-SE (Staff Exchanges) grant agreement 101086252; Call: HORIZON-MSCA-2021-SE-01, Project title: STARWARS (STormwAtER and WastewAtER networkS heterogeneous data AI-driven management). We would like to express our gratitude to "Montpellier Méditerranée Métropole" for having provided us with data essential to this research.

This research has also received support from the French national project ANR (Agence Nationale de la Recherche) CROQUIS (Collecte, représentation, complétion, fusion et interrogation de données hétérogènes et incertaines de réseaux d’eaux urbains).

References

1. Biswas, R., Mutz, M., Pimplikar, P., Ahmed, N., Werth, D.: Sewer-ai: Sustainable automated analysis of real-world sewer videos using dnns. In: International Conference on Pattern Recognition Applications and Methods (2023), <https://api.semanticscholar.org/CorpusID:257356921>
2. Dang, L.M., Hassan, S., Im, S., Mehmood, I., Moon, H.: Utilizing text recognition for the defects extraction in sewers cctv inspection videos. *Computers in Industry* **99** (08 2018). <https://doi.org/10.1016/j.compind.2018.03.020>
3. Ha, D.B., Schalter, B., White, L., Köhler, J.: Automatic defect detection in sewer network using deep learning based object detector (Apr 2023). <https://doi.org/10.5220/0011986300003497>, <https://publica.fraunhofer.de/handle/publica/449102>

4. Hassan, S., Dang, L.M., Mehmood, I., Im, S., Choi, C., Kang, J., Park, Y.S., Moon, H.: Underground sewer pipe condition assessment based on convolutional neural networks. *Automation in Construction* (06 2019). <https://doi.org/10.1016/j.autcon.2019.102849>
5. Haurum, J.B., Moeslund, T.B.: Sewer-ml: A multi-label sewer defect classification dataset and benchmark. In: *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. pp. 13456–13467 (June 2021)
6. Karthi, M., Muthulakshmi, V., Priscilla, R., Praveen, P., Vanisri, K.: Evolution of yolo-v5 algorithm for object detection: Automated detection of library books and performace validation of dataset. In: *2021 International Conference on Innovative Computing, Intelligent Communication and Smart Electrical Systems (ICES)*. pp. 1–6 (2021). <https://doi.org/10.1109/ICES52305.2021.9633834>
7. Kumar, S., Wang, M., Abraham, D., Jahanshahi, M., Iseley, T., Cheng, J.: Deep learning-based automated detection of sewer defects in cctv videos. *Journal of Computing in Civil Engineering* **34**, 04019047 (01 2020). [https://doi.org/10.1061/\(ASCE\)CP.1943-5487.0000866](https://doi.org/10.1061/(ASCE)CP.1943-5487.0000866)
8. Li, Y., Wang, H., Dang, L.M., Song, H.K., Moon, H.: Vision-based defect inspection and condition assessment for sewer pipes: A comprehensive survey. *Sensors* **22**(7) (2022). <https://doi.org/10.3390/s22072722>, <https://www.mdpi.com/1424-8220/22/7/2722>
9. Liao, H., RoyChowdhury, A., Li, W., Bansal, A., Zhang, Y., Tu, Z., Satzoda, R., Manmatha, R., Mahadevan, V.: Doctr: Document transformer for structured information extraction in documents. In: *Proceedings of the 2023 IEEE/CVF International Conference on Computer Vision (ICCV)*. pp. 19527–19537 (10 2023). <https://doi.org/10.1109/ICCV51070.2023.01794>
10. Moselhi, O., Shehab, T.: Classification of defects in sewer pipes using neural networks. *Journal of Infrastructure Systems - J INFRASTRUCT SYST* **6** (09 2000). [https://doi.org/10.1061/\(ASCE\)1076-0342\(2000\)6:3\(97\)](https://doi.org/10.1061/(ASCE)1076-0342(2000)6:3(97))
11. Oh, C., Dang, L.M., Han, D., Moon, H.: Robust sewer defect detection with text analysis based on deep learning. *IEEE Access* **10**, 46224–46237 (2022). <https://doi.org/10.1109/ACCESS.2022.3168660>
12. Therrien, J.D., Nicolai, N., Vanrolleghem, P.A.: A critical review of the data pipeline: how wastewater system operation flows from data to intelligence. *Water Science and Technology* **82**(12), 2613–2634 (08 2020). <https://doi.org/10.2166/wst.2020.393>, <https://doi.org/10.2166/wst.2020.393>
13. Wang, M., Cheng, J.: Development and Improvement of Deep Learning Based Automated Defect Detection for Sewer Pipe Inspection Using Faster R-CNN, pp. 171–192. Springer International Publishing (01 2018). https://doi.org/10.1007/978-3-319-91638-5_9
14. Yin, X., Ma, T., Bouferguene, A., Al-Hussein, M.: Automation for sewer pipe assessment: Cctv video interpretation algorithm and sewer pipe video assessment (spva) system development. *Automation in construction* **125** (May 2021). <https://doi.org/10.1016/j.autcon.2021.103622>