On object detection based on similarity measures from digital maps

Arthur Marzinkowski, Salem Benferhat, Anastasia Paparrizou, Cédric Piette

On object detection based on similarity measures from digital maps

Arthur Marzinkowski[1], Salem Benferat[1], Anastasia Paparrizou[2], and Cédric Piette[1]

[1]CRIL Université d'Artois, CNRS UMR 81881, France.
[2] LIRMM, Université Montpellier, CNRS UMR 5506, France.
{marzinkowski, benferat, paparrizou, piette}@cril.fr

Abstract. This paper deals with the problem of object detection from digital maps. We are interested in detecting objects in a map which are defined in the legend. We will explore different similarity measures to compare the legend objects to those detected in different areas of the map. Our object detection method is evaluated on maps representing wastewater networks. In particular, we are interested in the detection of objects that represent lifting stations and manholes. The ultimate goal, after detecting correctly the target objects, is to repair misfunctions or inconsistencies in the water supply or evacuation network. The experimental results show that our similarity measures give good accuracy results on the detection of the objects of the legends.

Keywords: object detection, maps, similarity measures, wastewater networks' data

1 Introduction

This paper focuses on map-like data that represents wastewater networks. Objects of interest are described in the map legend, with a focus on the detection of objects that represent lifting stations and manholes. The principle is therefore to take the sub-images associated with the objects in the legend as the reference image. The goal is then to detect all parts of the map that are sufficiently similar to the reference image. This problem is known as template matching (e.g. [1]) and is used in many applications processing images (e.g., [5]). Even if the images to be processed are of high quality, it is often difficult to detect all the legend objects present in the map. This is particularly true for images or maps that represent wastewater networks as the one given in Figure 1. Indeed, let us consider a legend object that represents a manhole. At the level of the legend (see Figure 2), a manhole is described by a complex geometric figure: a full blue circle inside another circle. At the map level, in some areas of the map (as the one illustated in Figure 4.(a), manholes appear isolated and independent (even if the background is rarely white as in the legends). However, in general the manholes do not appear exactly as they are as described by the legends. Namely, they do not appear independently and well separated on a white background (as in the legend). In reality, we find some objects that are superimposed on each other (as illustrated by Figure 4.(b)), others are on non-white backgrounds, which evidently makes the task of detection difficult. It is clear in this case that it will not be useful to search for geometric figures which have exactly the same shape as those given in the legend, but rather search for shapes which are (sufficiently) similar.

One of the objectives of this paper is to study different measures of similarities which are adapted to our framework of study, namely the maps of wastewater networks. The field of

water sciences (including wastewater networks's data management) is rich in heterogeneous data. There have recently been many works that call for the use of machine learning techniques (e.g., [9], [8], [6]) and classifiers ([14]). On the management of wastewater network data, there are already works to detect objects of interest, such as manholes, from images (e.g., [2], [13]). Existing work mainly used machine learning techniques on images to detect an object of interest which is the manhole (the most visible of urban networks). The work considered in this paper does not deal with images but with maps, and therefore can detect different objects identified in the legend of the map. An extra difficulty we face when dealing with maps is the heterogeneity of the data, meaning that the maps in our database come from different sources and may have different formats.

The rest of the paper is organized as follows. The next section presents the problem considered in the paper. Section 3 describes the approach we propose and the similarity measures for comparing the legend objects to segments of the map and extract the most plausible objects of interest. Section 4 presents the experimental results of our paper. The last section concludes the paper.

2 Problem description

The problem tackled in this paper concerns the management of heterogeneous information. The approach that is followed consists in transforming data of a different nature into a format that is easily exploitable by the Artificial Intelligence mechanisms of query answering. This is especially true for image type data or Portable Document Format (PDF) files, where it is important to extract their content and represent it as a factual database (which may or may not be uncertain).

Figure 1 shows an example of the maps we have to manage. In order to be able to query it and exploit its content, it is necessary to identify the relevant objects it contains. The legend (often present in maps) is a good starting point for identifying objects of interest associated with the map.

Figure 2 contains the legend of the map, given in Figure 1, from a small village of south France (written in French). It gives four elements (or objects) important for the description of wastewater networks.

The first object of interest represent manholes. A manhole is a structure allowing the cleaning and the inspection of a pipe of a network of waste water. Manholes play an important role because they are the access points to the networks. They are placed at regular intervals on the network and at each bend and intersection of the network.

Figure 3 gives an example of an image that contains such manholes. In Figure 2, we can find the representation of manholes on the maps and on the legends. A manhole is represented by a geometric figure: a blue circle included in another circle of black color.

Another important object is the so-called lifting station, which is a pump system used to raise the level of water. In the legend and maps, the lifting station is also represented by a geometric figure, more complex than the one associated with manholes. A lifting station

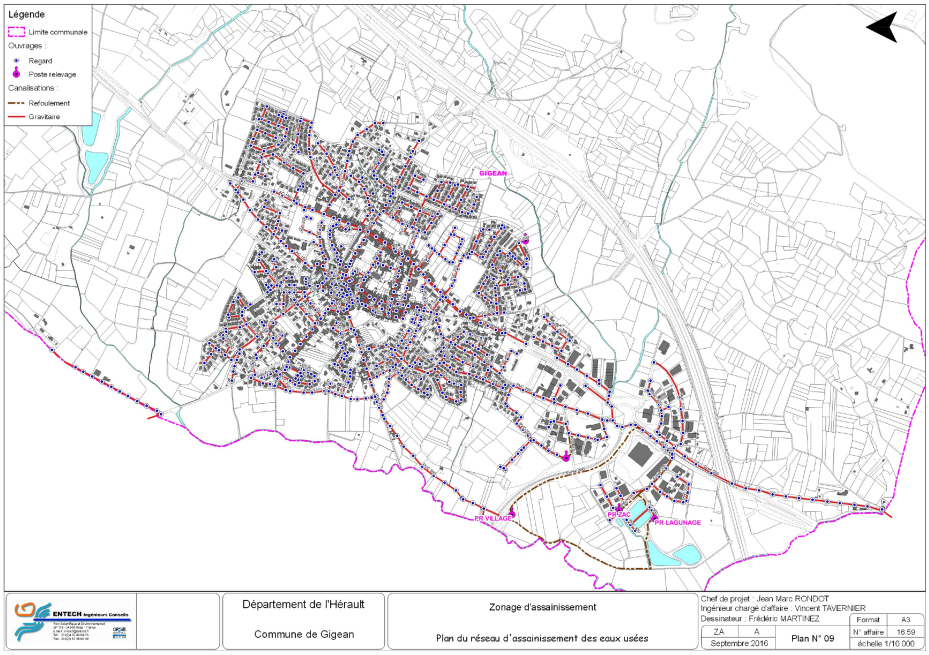


Fig. 1: Wastewater network map for `Gigean` village in France

is represented by a black circle inside a pink circle, the whole is attached to a pink arrow (appears below the manhole symbol in Figure 2).

In this paper, we focus on these two objects of sewerage networks. Note that these two objects are not represented in the same way on all maps. There are very few lift stations (six in all) on the map. On the other hand, the manholes are numerous in the maps (e.g., more than 800 manholes for `Gigean`) and are distributed in a non-uniform way. In some areas (which represent the most populated places), manholes are shown in a superimposed manner; which makes the task of detecting them very difficult.

2.1 Elements of wastewater networks

In the following sections, several terms in the field of sanitation networks will be used. We will therefore define them in the preamble:

- Manhole
  The manhole is an opening allowing the cleaning and the inspection of a pipe. They are placed at regular intervals on the network and at each bend and intersection of the network (see Figure Figure 3).

- Lifting station
  The lifting station is an automatic pump system that is used to raise the level of water

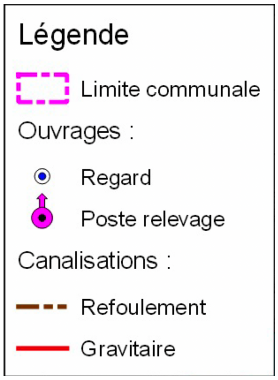


Fig. 2: Legend associated with `Gigean` map given in Figure 1, where "regard" means manhole and "poste de relevage" means lifting station

that accumulates below the so-called discharge level. The waters that flow below this level cannot flow via gravity.

– Gravity
The gravity is a pipe allowing the flow of the water without the use of a pump, using just the force of gravity.

2.2 Data on wastewater networks

As mentioned earlier, Figure 1 represents an example of data that we will work on. This is a map of the `Gigean` wastewater network in image form. Several symbols are visible on the legend such as manholes, lifting stations, discharges and gravity. We will first focus on the manholes and lifting stations because they are specific elements on the map.

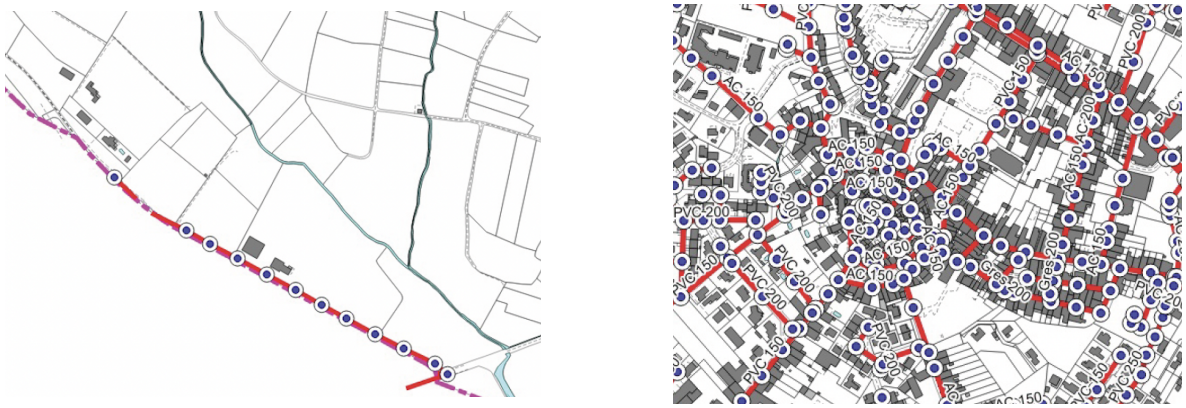
3 The proposed approach

3.1 Extraction methods

The method we use in this work is referred in the literature as template matching method [3]. This method is widely used in order to find one or more elements in an image that are similar to another one used as reference. Recall that, the reference elements are those appearing in the legend of the map. The element matching is done by calculating a matching score between the reference element and each sub-region of the image. A sub-region is defined by the dimensions of the reference image. This operation is a convolution over the image.

Figures 5a, 5b and 5c (figures which are inspired by the document given in https://towardsdatascience.com/intuitively-understanding-convolutions-for-deep-learning-1f6f42faee1), illustrate the main steps of our method. These steps are clearly similar to convolution operations (e.g., [4]); with one of the main difference is that in a convolution operation the output

Fig. 3: Example of images representing a manhole
Source: https://www.tubao.fr/tuyau-canalisation-pehd-reseaux-gravitaires



(a) A sub-image of the `Gigean` map (Figure 1) representing distant and independent manholes

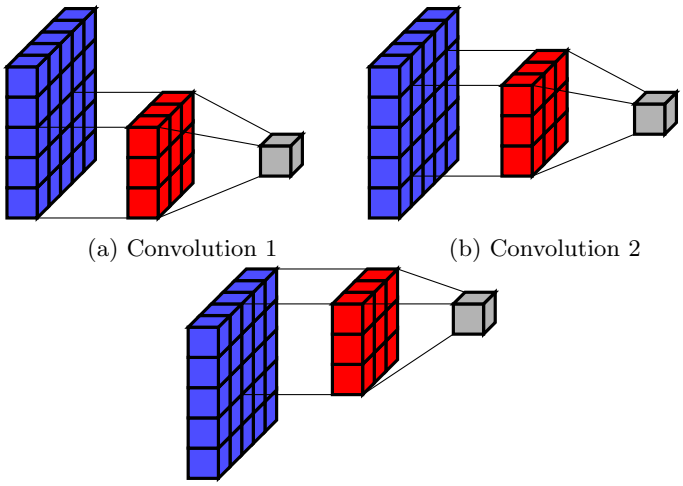
(b) A sub-image of the `Gigean` map (Figure 1) representing intertwined manholes

Fig. 4: Two examples of sub-images of the `Gigean` map (Figure 1) representing two different arrangements of manholes

is an image whereas our method the output is a 2 dimensional array of similarity degrees.

The blue part, in Figures 5a, 5b and 5c, represents the base image (in our example, the map of Gigean given in Figure 1) where we seek to detect the elements of interest (e.g., manholes and lifting stations).

The red part is the reference image which represents the description of the object that we seek to detect. In our example, the reference image can be the sub-image associated with a



(a) Convolution 1 (b) Convolution 2

(c) Convolution 3

Fig. 5: Conceptual Illustrations of Convolution

manhole, which is given by the legend (e.g., Figure 2).

The idea is to apply template matching techniques in order to detect references images in the map. More precisely, we propose to use the reference image (the red image) as a sliding window in order to iterate through all the sub-images (with the same size as the red image) of the base image (blue part).

At each iteration, we calculate a degree of matching or similarity between the sub-image of the reference image (image given by the legend) with the base image (in our case the Gigean map). We will use different similarity measures that are appropriate for image comparison. The results of all these degrees of similarity will be stored in a third matrix. In our example (Figures 5a, 5b and 5c), the similarity matrix is represented by the gray color. To be more precise, each of the comparisons of the reference image, with different parts of the base image, results in a degree of similarity, stored in a cell of the gray matrix.

Algorithm 1 summarizes our matching method based on similarity measures. The inputs to our algorithm are the two images: the base image (or the source image), denoted I, and the reference image, denoted J. Let (w1, h1) be the matrix's dimensions of the source image (base) and (w2, h2) be the reference's dimensions. The output of our algorithm is a similarity matrix, denoted S. This matrix is of size (w1 - w2, h1 - h2).

Indeed, only the sub-images of the base image which have the same dimension as the reference image are taken into account in the detection of objects. For each x and y coordinates of the similarity matrix S, the degree of similarity $S[x, y]$ is the result of the application of a similarity function between the reference matrix J and the base sub-image I composed of

the rows that are between x and $x + w2$ and the columns that are between y and $y + h2$.

The simarity matrix calculated by the algorithm will then be used to detect the objects of interest. Assuming that we seek to detect a single object, in this case we take the sub-image of the base image whose degree of associated similarity is maximum. In the general case of multiple object detection, we need to define a threshold above or below which we consider the similarity degree is good enough to represent the reference image.

Algorithm 1: Similarity-based matching algorithm

Input: Source image I
Input: Reference image T
Output: Similarity matrix S
$I \leftarrow Imgsrc$;
$T \leftarrow Imgref$;
foreach *coordonates (x,y) in S* **do**
  $S[x, y] \leftarrow matching\_score(T, I[x, y : x + w2, y + h2])$;
end

Once the similarity matrix is calculated by the Algorithm 1, a decision step must be added to it to determine which sub-images (or regions) of the map represent the reference image. In this paper, we first used a threshold to determine the list of regions of the map that may correspond to the objects we seek to detect. Then, we defined an algorithm for grouping very close regions which all potentially represent the detected objects. The difficult question posed here is whether two regions which are very close (namely, which are only a few pixels apart) correspond to a single object or to two different objects.

3.2 Matching measures

In this subsection, we present some similarity measures. We first introduce some notations. Let f and g be two matrices (not necessarily of the same size). These two matrices will intuitively encode the two images that are inputs of our algorithm (the base image and the reference image). These matrices will contain positive numbers where in our example will encode the luminosity of a pixel (typically the gray level or gray intensity of a pixel).

We denote by h the height of the image or matrix g and by w its width. We will also use the value n as $n = w \times h$ (the number of elements in g).

The matching measures are split in two main categories: intensity measures and correlation measure.

Intensity measures consist in computing the difference pixel by pixel between the reference image and the source image. The sum of squared differences method belongs to this category.

Sum of squared differences-based methods The sum of squared differences is a pixel by pixel intensity differences measure. The lower the matching score is, the more similar the

compared images are. The measures we will compare are the Root Mean Square Distance (RMS) and the Sum of Squared Differences or squared error (SSD). Formally:

– Sum of squared differences or squared error (SSD)

$$M_{SSD}(f,g) = \sum_{i=0}^{w-1} \sum_{j=0}^{h-1} (f(i,j) - g(i,j))^2 \tag{1}$$

– Root mean square distance (RMS)

$$M_{RMS}(f,g) = \sqrt{\frac{1}{n} \sum_{i=0}^{w-1} \sum_{j=0}^{h-1} (f(i,j) - g(i,j))^2} \tag{2}$$

The formulas 1 and 2 are equivalent in the sense that SSD and RMS produce the same relative order. In practice, only SSD is used due to its lower computing cost [3]. Other measures similar to (SSD) and (RMS) have been used in the context of image or signal processing (e.g., [12]).

For the SSD measure, when $M_{SSD}(f,g) = 0$, then we have a perfect matching. The maximum score depends on the type of image processed. SSD is symmetric since $M_{SSD}(f,g) = M_{SSD}(g,f)$ and satisfies the triangular inequality as a classic distance measure. The SSD measure is used, for instance, to track objects in videos [10]. The reference image is extracted from the video and used as template for the next frames.

We will illustrate with an example the sensitivity of SSD to luminosity/intensity variations, which is one of the main drawback of this measure.

Fig. 6: Fruit and vegetable image [1]

Figure 6 is an image of fruits and vegetables, considered as the source image. Let us now consider two examples of reference images, given by Figures 7a and 7b. The two images contain the same object, except that the image, given in Figure 7b is a strongly darkened version of the image given in 7a. In fact, the image, given in Figure 7a, is directly taken from the source image 6. Our aim is to find the exact locations of the reference object in the original image (Figure 6) from both reference images given Figures 7a and 7b.

Our algorithm based on the SSD measure is able to find the exact location of the template image 7a. However, it fails finding the original position of the darkened template. This is one

[1] https://monjardindidees.fr/optimisons-l-espace-au-potager

(a) An example of reference image)

(b) Darkened version of the image 7a

Fig. 7: Illustration of reference images

limitation of SSD (and RMS similarly), a measure which is sensitive to luminosity variations. Although in our case the reference image is heavily darkened, we have observed that even smaller brightness changes can significantly deteriorate the matching score.

Correlation measure Correlation measures are known to be a robust similarity measure that has been used in different application contexts (e.g. [5]). In this section, we are interested in the normalized version of cross-correlation measure (NCC), given by Equation 3 (see for instance [11] for its use in image correlation).

– Normalised cross-correlation(NCC)

$$M_{NCC}(f,g) = \frac{\sum_{i=0}^{w-1} \sum_{j=0}^{h-1} f(i,j).g(i,j)}{\sqrt{(\sum_{i=0}^{w-1} \sum_{j=0}^{h-1} f(i,j)^2) \cdot (\sum_{i=0}^{w-1} \sum_{j=0}^{h-1} g(i,j)^2)}} \tag{3}$$

When using a normalized cross-correlation (NCC) measure, the higher the obtained similarity value, the more similar are the images to be compared.

In particular, when $M_{NCC}(f,f) = 1$ then we have a perfect matching, which is the maximum score for NCC. Similarly, the degree 0 is the minimal score for NCC.

The main reason for NCC being more robust is that it is less sensitive to luminosity variations [7] comparing to SSD measures.

Indeed, consider again our example with the figures 6, 7a and 7b. The use of the NCC similarity measure has made it possible to detect the object of interest from the two reference images (7a and 7b) which was not the case with SSD. This illustrates the robustness of NCC even with really high variations of luminosity.

We now illustrate different similarity measures on academic examples. Let us consider two grey scale source images represented by the matrices I_1 and I_2 respectively. These two matrices are of dimensions 3 x 3. These two images are such that $\forall i \in \{0,1,2\}, \forall j \in \{0,1,2\}, I_2(i,j) = I_1(i,j)/2$.

We also assume that we have a unique reference image T of dimensions 2x2:

$$I_1 = \begin{bmatrix} 10 & 20 & 30 \\ 40 & 50 & 60 \\ 70 & 80 & 90 \end{bmatrix}$$

$$I_2 = \begin{bmatrix} 5 & 10 & 15 \\ 20 & 25 & 30 \\ 35 & 40 & 45 \end{bmatrix}$$

$$T = \begin{bmatrix} 10 & 20 \\ 40 & 50 \end{bmatrix}$$

The four matrices (S_1, ..., S_4) below give the similarity results, based on the two measures SSD and N, between the reference image T and each of the images sources I_1 and I_2. Each matrix is of dimension 3x3. More precisely, the matrix S_1 (resp. S_3) is the result of applying the SSD similarity measure to the images I_1 and T (resp. I_3 and T). Similary, the matrix S_2 (resp. S_4) is the result of applying the NCC similarity measure to the images I_1 and T (resp. I_3 and T).

$$S_1 = \begin{bmatrix} 0. & 0.06855912 \\ 0.42772362 & 0.6574571 \end{bmatrix}$$

$$S_2 = \begin{bmatrix} 1. & 0.9941072 \\ 0.9742593 & 0.9656402 \end{bmatrix}$$

$$S_3 = \begin{bmatrix} 0.5 & 0.22281712 \\ 0.05940606 & 0.07190938 \end{bmatrix}$$

$$S_4 = \begin{bmatrix} 1. & 0.9941072 \\ 0.9742593 & 0.9656402 \end{bmatrix}$$

We can observe that the matrices S_2 and S_4, both obtained by using the NCC measure, are identical. This illustrates what was aforementioned, NCC is less sensitive to intensity variations on pixels. However, as indicated at the level of the array S_1, the use of the SSD measure only allows to find (perfectly) the location of T in I_1. However, when we apply SSD on the source image I_2, we obtain a lower similarity score (a degree 0.5 in the matrix S_3,). This shows that SSD allows us to find reference images only if they are nearly identical to the original. Note that in our map example application, this should not be a problem. This is because there is not a large variation in pixel values between map items and the reference inside the legend.

The similarity measures seen so far have been defined on gray level images. These measurements can easily be extended to colored images. A natural solution is to simply apply the similarity measure on each channel (R, G, B) and thus sum up the result. This will give us a single similarity value on each pixel.
To illustrate these similarity measures, we consider the following example with two colored images, simply denoted again as I_1 and I_2. Again, I_2 is obtained by dividing each number in the I_1 matrix by 2:

$$I_1 = \begin{bmatrix} (0,0,0) & (255,0,0) & (0,255,0) \\ (0,0,255) & (255,255,0) & (255,0,255) \\ (0,255,255) & (255,255,255) & (127,127,127) \end{bmatrix}$$

$$I_2 = \begin{bmatrix} (0,0,0) & (127,0,0) & (0,127,0) \\ (0,0,127) & (127,127,0) & (127,0,127) \\ (0,127,127) & (127,127,127) & (63,63,63) \end{bmatrix}$$

$$T = \begin{bmatrix} (0,0,0) & (255,0,0) \\ (0,0,255) & (255,255,0) \end{bmatrix}$$

As we did with grayscale images, we apply SSD and NCC to the two color source images I_1 and I_2. We first add a matrix, S, which will be the result of non-normalized SSD on I_1 to illustrate the operation on different channels:

$$S = \begin{bmatrix} 0. & 520200. \\ 260100. & 374022. \end{bmatrix}$$

Let us consider for example the $S[0,1] = 520200$. This example is simple to check, because we only have 255 and 0 as values. We so need to count the range of different channels (between 0 and $255^2 = 65025$).

Below is the sub-matrix of I_1 in $I[0,1]$:

$$S_1 = \begin{bmatrix} (255,0,0) & (0,255,0) \\ (255,255,0) & (255,0,255) \end{bmatrix}$$

This gives us $S[0,1] = 255^2 * 1 + 255^2 * 2 + 255^2 * 3 + 255^2 * 2 = 520200$.

$$S_1 = \begin{bmatrix} 0. & 1. \\ 0.70710677 & 1. \end{bmatrix}$$

$$S_2 = \begin{bmatrix} 1. & 0.20412417 \\ 0.70710677 & 0.5383155 \end{bmatrix}$$

$$S_3 = \begin{bmatrix} 0.50591326 & 1. \\ 0.7099016 & 1. \end{bmatrix}$$

$$S_4 = \begin{bmatrix} 1. & 0.20412415 \\ 0.70710677 & 0.53781 \end{bmatrix}$$

Again, applying NCC measures leads to identical results for I_1 and I_2.

4 Experimental Evaluation

This section presents the experimental results where the two main similarity measures, namely SDD and NCC, have been evaluated on the `Gigean` wastewater network map given in Figure 1. The Gigean map has been used as an PNG image with a resolution of (19850,14033) and using

format RGB (3 bytes value per pixel) for representing the colored image. The experiments were conducted on a 12 core CPU 3.7GHz, 16Go RAM computer.

We used two reference images, extracted from the legend (given in Figure 2), associated with the objects of manholes and lifting stations. Table 1 gives the number of objects considered in our study.

Tables 2 and 3 summarize the results obtained during our experiments.

Objects	Total number of objects
Manholes	885
Lifting stations	5

Table 1: Number of manholes and lifting stations in the map

Objects	Thresholds	Candidate sub-images	Detected objects	False positive	PCC	False positive False rate
Manholes	0.10	30503	595	0	67.11%	0 %
	0.148	79675	726	1	81.80 %	0.13 %
	0.15	82215	730	1	82.25 %	0.13 %
	0.20	182934	942	157	88.58 %	16.66 %
Lifting stations	0.10	247	2	0	20 %	0 %
	0.15	609	3	0	40 %	0 %
	0.20	1375	5	0	80 %	0 %
	0.22	1932	6	0	100 %	0 %

Table 2: Results of normalized SDD on `Gigean` map

Objects	Thresholds	Candidate sub-images	Detected objects	False positive	PCC	False positive False rate
Manholes	0.95	30984	598	0	67.45 %	0 %
	0.935	31945	685	0	74.40 %	0 %
	0.9	139101968	-	-	-	-
	0.85	242781641	-	-	-	-
Lifting stations	0.95	249	2	0	20 %	0 %
	0.9	1613	5	0	80 %	0 %
	0.85	14890	8	2	100 %	25 %

Table 3: Results of normalized NCC on Gigean map

The "Objects" column simply contains the two types of objects that our algorithm have to detect. The "Thresholds" column provides the different used thresholds. If the degree of

similarity of a region, of the source image, is above (for NCC), or below (for normalized SSD) the threshold, then the region is considered to be sufficiently similar to the reference object. In this case, the region will therefore be considered as potentially containing the reference object.

The column "Candidate sub-images" indicates the number of elements of the similarity matrix which satisfy the threshold criteria. Specifically, this represents all sub-regions that have a sufficient similarity score to be considered as representing the reference image. However, multiple regions can represent a single object. The next column, "Detected Objects", specifies how many different detected objects exist on the map.

Therefore, the column "Detected objects" represents the number of detected objects once the sub-regions have been grouped. The algorithm that selects the set of detected objects from the candidate objects has a complexity of $O(n^2)$. Consequently, when the number of candidate regions is too high (in our experiments above 200,000), it is not possible to accurately determine the number of detected objects (which explains the presence of the symbol "-" in the Table 2). Tables 4 and 5 summarize running times for the computation of similarity scores (which is independent of the threshold) and for grouping algorithm.

Finally, the column "False positive" simply indicates the number of false positive. This represents the number of sub-regions that have a sufficient similarity score (therefore considered by our algorithm representing the reference objects) while the sub-region does not contain any reference object. Note that the detection of false positives is done manually.

In addition, the column "PCC" (Percent Correctly Classified) shows the ratio between the total number of elements to detect and the number of elements actually detected (after removing false positive). The column "False positive rate" provides the ratio between the number of false positive and the total number of object detected (false positive regions included).

Objects	Thresholds	Computing similarity scores (seconds)	Region grouping algorithm (in seconds)
Manholes	0.10	50	13
	0.148		30
	0.15		32
	0.20		105
Lifting stations	0.10	50	10
	0.15		10
	0.20		10
	0.22		12

Table 4: Running time when using normalized the SDD measure on `Gigean` map

Figure 8 is a black and white image representation of the similarity matrix (with a threshold equal to 0.9). Each white pixel in this image is a value of the similarity matrix whose value exceeds 0.9. Thus, each white pixel is the starting point of a valid candidate sub-image (of the source image). Future work is to develop efficient clustering algorithms for clustering areas that are sufficiently similar to be detected but they refer to the same object.

Objects	Thresholds	Computing similarity scores (seconds)	Region grouping algorithm (in seconds)
Manholes	0.95	50	10
	0.935		20
	0.9		-
	0.85		-
Lifting stations	0.95	50	10
	0.9		10
	0.85		11

Table 5: Running time when using normalized the normalized NCC on `Gigean` map

Fig. 8: Image representation of similarity matrix with a threshold equal to 0.9

Figure 9 shows some rare cases of false positives recovered by NCC for lifting stations. Unlike using NCC for manhole detection, these false positives are more localized. Most of them are located in the pink region of the map like; in particular on certain texts written in pink color. False positives are almost non-existent (with a threshold of 0.9 for example) in the detection of manholes, this is due to the specific and proper manholes representation.

5 Conclusion and Further Work

In this paper, we studied the detection of multiple objects in the context of maps of wastewater networks. We particularly applied a template matching algorithms augmented with region grouping algorithm. We examined two similarity measures (i.e., the Sum of Squared Difference (SSD) and the Normalized Cross-Correlation (NCC)). We showed some advantages and drawbacks of those measures. The SSD measure is efficient at finding exact replicates of the

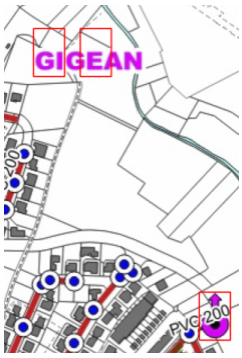


Fig. 9: False positive example with NCC on lifting station

reference image inside the source image, but intensity variation can deteriorate the results, even though they seem visually identical. The NCC measure on the other hand is more tolerant with those variations, though it can lead to more incorrect results. We tested both measures on the wastewater network map for `Gigean` with both manhole and lifting station. SSD brings more accurate results in object detection than NCC. We concluded that this was due to an overall high brightness across the whole image. Experimental results showed that it is the appropriate method for punctual object detection in this digital map.

Still some questions remain open; the first one is how to choose the right threshold automatically, namely, without any prior knowledge on the map (e.g., giving the number of elements inside it). The second is how to improve the region grouping algorithm. In this study we used a partitioning algorithm, but on high quality maps, the number of region matches can be quite high. Obviously, this limits the maximum number of objects we can detect and also the resolution of the images we operate on. One solution could be to assume that all matching regions that are adjacent in the similarity matrix represent a single object. Then, we can pick the location of the best score for the bounding box of the object. But, what if two different objects are so close that this assumption is wrong? This is still an open question.

Another direction is to study other types of elements in maps, like lines and areas of interest. Such types of elements cannot be grouped as easily as punctual ones, since defining a bounding box might not be a satisfying solution in such cases. Another problem to be solved is the automatic detection of those elements. Currently, the template images have been extracted manually in order to be able to have a general algorithm. A method of automatically detecting elements inside the legend, as well as labeling these elements, is under development.

Acknowledgemnts 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). This research has also received support from the french national projet ANR CROQUIS (Collecte, représentation, complétion, fusion et interrogation de données de réseaux

d'eau urbains hétérogènes et incertaines) project, grant ANR-21-CE23-0004 of the French research funding agency (Agence Nationale de la Recherche ANR).

References

1. Roberto Brunelli. *Template Matching Techniques in Computer Vision: Theory and Practice*. Wiley Publishing, 2009.
2. Nanee Chahinian, Carole Delenne, Benjamin Commandré, Mustapha Derras, Laurent Deruelle, and Jean-Stéphane Bailly. Automatic mapping of urban wastewater networks based on manhole cover locations. *Comput. Environ. Urban Syst.*, 78, 2019.
3. G. F. Cox. Template matching and measure of match in image processing. 1995.
4. Jiuxiang Gu, Zhenhua Wang, Jason Kuen, Lianyang Ma, Amir Shahroudy, Bing Shuai, Ting Liu, Xingxing Wang, Gang Wang, Jianfei Cai, and Tsuhan Chen. Recent advances in convolutional neural networks. *Pattern Recognition*, 77:354–377, 2018.
5. Nazanin Sadat Hashemi, Roya Babaei Aghdam, Atieh Sadat Bayat Ghiasi, and Parastoo Fatemi. Template matching advances and applications in image analysis. 2016.
6. Hanna Meyer and Edzer Pebesma. Machine learning-based global maps of ecological variables and the challenge of assessing them. *Nature Communications*, 13(1):2208, Apr 2022.
7. Badrul Mohamad, Shahrul Yaakob, Rafikha Aliana A. Raof, A. Nazren, and Mohd Wafi Nasrudin. Template matching using sum of squared difference and normalized cross correlation. pages 100–104, 12 2015.
8. Steve Mounce. A comparative study of artificial neural network architectures for time series prediction of water distribution system flow data. 04 2013.
9. Huu Du Nguyen, Tai Quang Dinh Nguyen, Hien Nguyen Thi, Bui Quoc Lap, and Thi-Thu-Hong Phan. The use of machine learning algorithms for evaluating water quality index: A survey and perspective. In *2022 International Conference on Multimedia Analysis and Pattern Recognition (MAPR)*, pages 1–6, 2022.
10. Kevin Nickels and Seth Hutchinson. Estimating uncertainty in SSD-based feature tracking. 2002.
11. Theodore A. Scambos, Melanie J. Dutkiewicz, Jeremy C. Wilson, and Robert A. Bindschadler. Application of image cross-correlation to the measurement of glacier velocity using satellite image data. *Remote Sensing of Environment*, 42(3):177–186, 1992.
12. Yi Sun. Root mean square minimum distance as a quality metric for stochastic optical localization nanoscopy images. *Scientific Reports*, 8(1):17211, Nov 2018.
13. Yongtao Yu, Haiyan Guan, Dilong Li, Chunhua Jin, Cheng Wang, and Jonathan Li. Road manhole cover delineation using mobile laser scanning point cloud data. *IEEE Geoscience and Remote Sensing Letters*, 17(1):152–156, 2019.
14. Kaiqiang Zhu, Yushi Chen, Pedram Ghamisi, Xiuping Jia, and Jón Atli Benediktsson. Deep convolutional capsule network for hyperspectral image spectral and spectral-spatial classification. *Remote Sensing*, 11(3):223, 2019.