

AUTOMATING THE UNDERGROUND CADASTRAL SURVEY: A PROCESSING CHAIN PROPOSAL

A. Carreaud^{a,*}, F. Mariani^b, A. Gressin^a

^a University of Applied Sciences Western Switzerland (HES-SO / HEIG-VD), 1400 Yverdon-les-Bains, Switzerland,
Insit Institute - (antoine.carreaud,adrien.gressin)@heig-vd.ch

^b Geneva Industrial Services, Switzerland - fabio.mariani@sig-ge.ch

Commission II, WG II/6

KEY WORDS: Deep learning, Photogrammetry, Automatic 3D segmentation.

ABSTRACT:

In order to ensure the proper functioning and evolution of underground networks (water, gas, etc.) over time, municipal services need to maintain accurate and up-to-date maps. Such maps are generally updated using traditional data acquisition methods (total station or GNSS), which are time-consuming, expensive, and require several teams of surveyors in the field. In this context, an important topic of research is the automation of the updating of the underground cadastre in order to save time, money, and human effort. In this paper, we present a new method that we developed ranging from the choice of the acquisition system, the tests carried out in the field to the detection of objects and the automatic segmentation in a 3D point cloud. We have chosen to use a convolutional neural network on images for the detection of objects that are part of the underground cadastre. As the next step, objects are projected to obtain a 3D point cloud segmented based on the object type. The vectorization step is still under development so that objects can be converted to vector format and therefore be used for updating the cadastre. The results based on excavation sites with well-represented objects in our training database are excellent, approaching 96% accuracy. However, the detection of rare objects is much less good and thus remains a topic for future research. Overall, the complete processing chain allowing to automate as much as possible the update of an underground cadastre is presented in this paper.

1. INTRODUCTION

Cities around the world require large and diverse underground networks of water, gas, electricity and district heating pipelines, which allow for the proper functioning of the city. To ensure the efficient operation of these networks and their evolution over time, municipal services must maintain accurate maps. Until now, these tasks have been carried out using traditional data acquisition methods (with a total station or GNSS), which are time-consuming, costly and require several teams of surveyors in the field. These surveys usually take place during excavations that are carried out as part of underground maintenance works and can be extremely complex due to the high density of the various networks, which are often overlapping (Figure 1).



Figure 1: An example of open excavations that are to be surveyed, showing overlapping objects and pipes.

To obtain an automated method for updating the underground cadastre, several important challenges have to be overcome:

1. The choice of the acquisition system, which must ensure absolute accuracy in the accuracy classes necessary for the cadastral update while also allowing the collected data type (point cloud, photos, vector data) to be used in the chosen automatic recognition algorithm. The acquisition step must be faster and at least as user-friendly as the traditional method.

2. The choice of an automatic extraction method for 3D cadastral objects must provide the best recognition rate (precision and recall rate), even under difficult conditions that are often encountered in the field.

In fact, existing methods for automating cadastre surveys generally use ortho-image data acquired by drones (Crommelinck et al., 2016, Cerioni and Meyer, 2021, Picterra, 2022). However, considering the entanglement of the objects and pipes visible in the previous examples (Figure 1), we quickly understand the limitations of such 2D methods. Therefore, we propose to use a photogrammetric survey and to benefit from the multiple points of view obtained from such acquisitions to solve this complex problem.

Moreover, the task of automatic image segmentation has seen many improvements in recent years, using increasingly deep and complex convolutional neural networks. Optimisation and regularisation methods have also become more and more efficient (such as the recent Adam optimisation algorithm, which is widely used today (Kingma and Ba, 2014), learning rate decay, Dropout method, skip connections method (Goodfellow et al., 2016)). Computing power has also evolved considerably. The combination of these factors has made the field of automatic image segmentation (a small part of deep learning) more efficient. Naturally, deep learning is gradually arriving in all fields (medicine, translation, economics, autonomous cars, etc.), as well as in the small world of surveying. This is why we have chosen this method for the automatic extraction part of the processing chain that is presented in this paper.

We present a new method that we developed ranging from the choice of the acquisition system, the tests carried out in the field to the detection of objects and the automatic segmentation in a 3D point cloud, allowing us to automate as much as possible the updating of the underground cadastre.

*Corresponding author

2. PROPOSED METHOD

Our processing chain, shown in Figure 2, is composed of a common part of data acquisition, verification and 3D reconstruction. Two independent parts follow: the first is the learning step to recognize objects (Training stage), which should be done only once, and the second that will be used in production (Production stage), which must be as automated as possible.

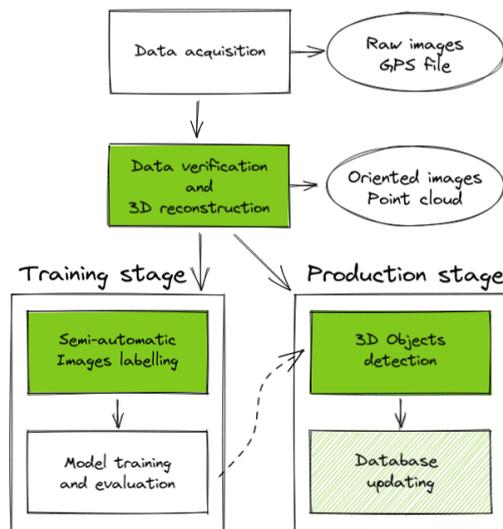


Figure 2: Processing chain that was developed. In plain green are the automated steps and in hashed green are the steps that remain to be completed. Rectangles indicate processes and ellipses the data.

2.1 Data acquisition

To choose the best acquisition systems, we assessed several commercial solutions (Figure 3) based on various criteria, such as acquisition time, processing time, purchase and maintenance costs, as well as absolute and relative accuracies.



Figure 3: The acquisition systems tested for the project.

At the time of our tests, the final choice was an SLR camera equipped with a GNSS RTK module, composed of a helical antenna and a u-blox receiver¹. Nowadays, such RTK geolocation solutions are also available for smartphones, which makes it possible to have an even more portable acquisition solution, while still ensuring sufficient image quality. Nevertheless, the processing chain presented here remains directly usable by switching to such a system. This system allows several types of data to be used: raw images, as well as oriented images, depth maps and

¹<https://www.redcatch.at/3dimagevector/>

point clouds after processing in a photogrammetric software. Furthermore, the RTK mode also allows for an absolute georeferencing of adequate accuracy for the updating of the cadastre (Forlani et al., 2018). Full details of the comparisons between the different systems were published in the 26th IGSO Gazette (Carraud and Gressin, 2021). This paper will not go into detail about the choice of acquisition system.

Finally, we were able to send surveyors in the field with this system, who were able to collect data on 50 excavation sites, for a total of about 1,700 images at this date (an example of a site is shown in Figure 4).



Figure 4: 3D reconstruction of pipeline objects based on images from multiple viewpoints.

2.2 Data verification and 3D reconstruction

Given the amount of data already acquired, or to be acquired in the future, it would have been too time-consuming to process all this data manually. We therefore developed a system for collecting, verifying and processing data. Our system is composed of two elements: (1) a web interface, allowing surveyors to deposit acquired images in the field, and to check directly if RTK GNSS information are of sufficient quality, and (2) a back-end script computing bundle-adjustment, images orientation, depth maps and point clouds. All the infrastructure is coded in python, with Flask² for the web interface, Nextcloud³ instance and the Owncloud API⁴ for the storage part and Agisoft Metashape API⁵ for the photogrammetric computation. This step, which is used for both the training and the production stages, has been fully automated, allowing us to already process all the data as they were acquired.

2.3 Training stage

The rapid developments and increasing accessibility of libraries allowing the use of so-called deep learning methods encouraged us to apply these methods for this project. Although deep learning can solve many problems in computer vision, such as image classification, object detection, segmentation and image enhancement, it requires large training datasets characterized by high-quality labels. Labeling is usually done manually and is a time-consuming and expensive task: for each captured photo, one has to manually mark all pixels belonging to the known objects. This leads to a further challenge: labeling a multitude of images accurately and very quickly.

²<https://flask.palletsprojects.com/en/2.0.x/>

³<https://nextcloud.com/>

⁴<https://github.com/owncloud/pyocclient>

⁵<https://www.agisoft.com/>

2.3.1 Data labelling To have a large number of labeled images, we developed a fast labeling method (Carreaud et al., 2021) inspired by the authors (Braun et al., 2015). As can be seen in Figure 4 (which is a site used to train the neural network), many images see the same objects, so the goal is to avoid doing the job several times. In this method, only the point cloud has to be labeled by hand, while the rest is fully automated. This allows us to save 60 % of the time compared to a fully manual method. Our method is composed of three main steps. Firstly, a 3D reconstruction by photogrammetric processing provides localized and oriented images with a dense point cloud (3D data). This process has been fully automated using the Agisoft Metashape Python API. Then, the point cloud is manually labelled by an operator in the CloudCompare software. In order to save time, no instance separation is made manually on the 3D data. Instances will be computed automatically in the last step. Finally, 3D labelled data is cleverly projected on each image to obtain labelled images (in COCO format).

This last stage is the core of our method (see Figure 5): (1) the point cloud is sorted by classes; (2) the DBSCAN algorithm is employed on each class to create object instances; (3) the depthmaps (from the 3D reconstruction) are used to project only visible points on the image; (4) the alpha-shape algorithm allows to create the hull around the objects; (5) finally labelled images and annotation files in COCO format are computed and exported. All processing steps have been automated in a Python script.

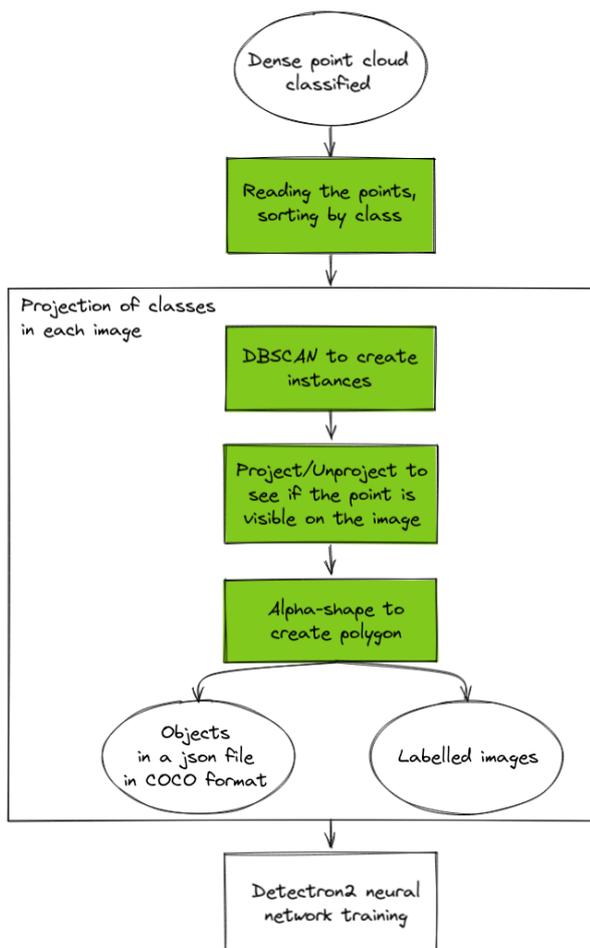


Figure 5: Processing chain of the projection step for semi-automatic labelling. In plain green are the automated steps. Rectangles indicate processes and ellipses the data.

It should be pointed out that some objects can be very tedious to label by hand on 2D images: humans can easily take shortcuts whereas the machine will remain strict. As seen in Figure 6, labelling of the 3D object is done by clicking four times in two different views (images 1 and 2), resulting in a precise labelling (image 3). A detailed description and accuracy analysis of our method is already presented in the paper (Carreaud et al., 2021) and will not be repeated here.

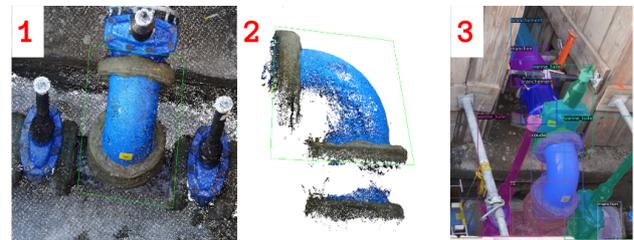


Figure 6: Manual 3D labelling process in only two steps (1+2) and the result after projecting 3D labels on one image with our method (3).

2.3.2 Model training For the update of the underground cadastre we need to detect the classes of the objects, as well as to differentiate them, i.e. to know that there are two pipes and not only that there is a pipe class on the image. Hence, in our research we focused on instance segmentation, which refers to finding the position and pixel masks of known objects in an image and differentiating them. Instance segmentation may be defined as the technique of simultaneously solving the problem of object detection (providing the classes and bounding boxes), as well as that of semantic segmentation (giving fine inference by predicting labels for every pixel) (Hafiz and Bhat, 2020). Training an instance segmentation network requires tens of hundreds of images depicting the object, ideally with diverse viewpoints and backgrounds. Once our database was constituted (to date it contains 1,700 labeled images), we could test the first training runs with the Detectron2 library (Wu et al., 2019).

To train a convolutional neural network with tens or hundreds of millions of weights, it is necessary to have very large computing power and an extremely large training data set. In many applications it is not possible to have so much data. Thus, we have used the principle of transfer learning to use a network pre-trained on classes different to ours. The principle is to freeze the weights of the convolution operations and adjust the weights of the classification (the output layer) during a quick training on data labelled with the classes we want to detect. There are still many adjustable parameters, called hyperparameters, which have been the subject of many tests. The hyperparameters that had the strongest impact on our results were first the learning rate, the pyramid of anchor box sizes with their ratio in order to better represent the size of the objects to be found in our images and the pyramid of image sizes to be used in order to partially solve the problem of different scales. Finally, the number of images per mini-batch has been increased to 4; the recommended sizes are 32, 64 or 128 to avoid a slow and chaotic convergence. However, the RAM of the graphics card (a Tesla V100 24Gb RAM) is very quickly saturated and does not allow us to increase the size of the mini-batches, which results in a case of stochastic gradient descent (Ng et al., 2022). The use of a real computing server is one of the perspectives of our research.

The network we selected is a network with a depth of 50, the R-50 FPN, pre-trained on the 2017 COCO dataset (118000 labelled images) over 37 epochs (Wu et al., 2019). In order to train the neural network as well as possible, i.e. to be able to generalize as

well as possible and/or to not overfit on the training data, we used the usual train/dev/test sets structure taught by (Ng et al., 2022) course. This structure also makes it possible not to over-adjust on the test set but only on the dev set as the various experiments are conducted. In other words, we use this structure to optimize both the bias (having a low bias, i.e. having the highest possible accuracy on the training data, approaching the Bayes error) and the variance (having a low variance, i.e. having an accuracy on the development and test datasets that is as close as possible to the training one).

2.4 Production stage

In this stage, we assume that the object detection method has already been trained in the training stage. We use the model trained on our custom data to make predictions on other images. To achieve very good results we can use a high threshold for the inference phase (above 90% confidence), which has the effect of decreasing the recall rate and increasing the precision. But as in our case each object is seen in at least three images, this allows us to optimise precision at the sacrifice of recall, as an object not detected in one image will certainly be detected in the next.

3D Objects segmentation We detect objects in the images and then create masks by class in order to build point clouds by class, which allows us to obtain various useful information for the update (altitude of an object, diameter of a pipe, etc.). The method is extremely simple, we have already used it in the context of a Master thesis to extract grapes from a point cloud for precision agriculture (Beniaouf and Gressin, 2021) and others also use it to classify point clouds of historical buildings (Pellis et al., 2022). The results are shown in Section 3..

Database update In this step, we need to introduce a vectorization step of the detected 3D objects and a comparison with the existing database to detect changes. In addition, several object-related attributes have to be known for the database update. These specific processes are still under development (see Section 5.).

3. RESULTS AND DISCUSSION

In this section we will primarily discuss the results of the deep learning and 3D point cloud segmentation processes. The results of the fast labelling method used in this research have already been published in a previous publication (Carreaud et al., 2021), as well as the results of the choice of the acquisition system.

For the results we used an excavation site from the test set. This site does not include all types of objects that we aim to detect, but it allows us to understand in a rather simple case study the results that can be achieved with the processing chain presented. Other sites have other types of objects, some classes of which are extremely under-represented in our training database and thus strongly degrade the results. The detection of these objects is still under development and research (see Section 5.). For accuracy analyses, the library we used (Detectron2(Wu et al., 2019) allows easy access to accuracy and Intersection over Union (IoU) information on each class. However, it is more interesting to analyze the results in the form of confusion matrices which give recall (producer accuracy) and precision (user accuracy) information, as well as information on the confusions between classes (Barsi et al., 2018). In this section we present the results at the image segmentation level as well as at the 3D point cloud segmentation level. Accuracy analyses are therefore done using confusion matrices.

3.1 Image level

Figure 7 shows a raw result after inference on an image of a test excavation site. From this result, we extract the masks per class in order to apply the 3D segmentation method (presented in Section 2.4) and obtain the labelled point cloud visible in figure 9.



Figure 7: Raw image after inference.

Figure 8 shows the normalized confusion matrix calculated on the 22 images of the test excavation site. There is a slight confusion between the detection of tile valves and butterfly valves, as these two types of valves are very similar in appearance. It is interesting to understand where this confusion originates from. For this, we have created a confusion matrix per image on which we notice that there are only two photos on which the convolutional neural network has made a mistake between the two types of valves. We can hope that with the method presented in Section 2.4 this confusion will be less on the point cloud (a point in 3D is created only if it is seen on 3 photos). Other, more minor, confusions are made, and by analyzing the masked images we notice that they are more often overflows of the limits between the objects rather than errors.

Table 1 presents the different precision or User Accuracy (UA) and Recall or Producer Accuracy (PA) obtained on each class, as well as the average accuracy or Overall Accuracy (OA). Overall the UA and PA are very good, the Bayes error was estimated at 99%, so it would theoretically be possible to improve these results a little bit. The PA of the butterfly valve detection is lower than the others due to the problem stated in the previous paragraph.

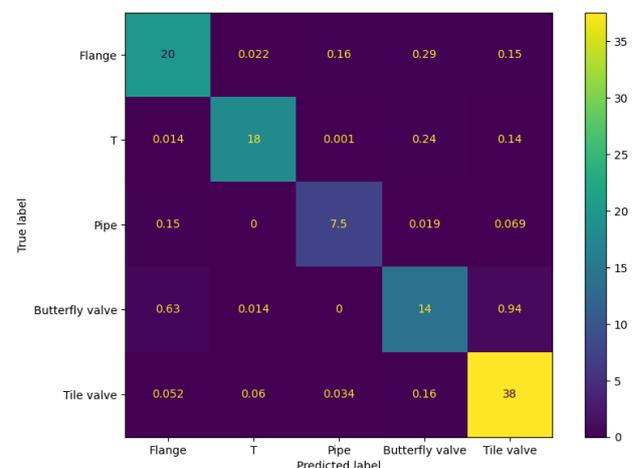


Figure 8: Normalized confusion matrix on images, each box corresponds to a percentage of the total data (pixels).

	User Acc. [%]	Producer Acc. [%]
Flange	95.9	96.9
T	99.5	97.8
Pipe	97.4	96.9
Butterfly valve	95.2	89.8
Tile valve	96.6	99.2
Overall Accuracy = 96.8 %		

Table 1: User Accuracy and Producer Accuracy obtained on each class and Overall accuracy on images.

3.2 Point cloud level

Figure 9 shows (a) the raw point cloud after photogrammetric processing, as well as (b) the point cloud segmented by class with the presented method (see Section 2.4) overlaid with the raw point cloud.

Figure 10 presents the normalized confusion matrix calculated on the 21 million points of the test excavation site. It can be seen that the confusion between the two types of valve discussed at the image level in Section 3.1 has disappeared as we might have expected. On the other hand, this time there is a confusion between the flange and butterfly valve classes, looking at the point cloud we can observe that two of these objects are stuck together, so the boundary is not well defined adding a slight confusion. In order to improve these limits it would be possible to apply different filters, we are still working on improving this. Although the most important thing for the update is to know that an object is present in the right place, if a boundary of an object overlaps another one as long as it does not hide a whole object it is not a serious problem.

Table 2 shows the different precision or UA and Recall or PA obtained on each class, as well as the average accuracy or OA. We notice that we lose a little bit of accuracy on all the classes when we switch to the 3D point cloud, going from 96.8 to 96.0% of average accuracy. The result is still very good. It is the butterfly valve and flange classes that lose the most accuracy due to the problem described in the previous paragraph.

	User Acc. [%]	Producer Acc. [%]
Flange	92.1	95.7
T	97.9	97.9
Pipe	97.5	95.3
Butterfly valve	93.2	90.0
Tile valve	97.9	98.9
Overall Accuracy = 96.0 %		

Table 2: User Accuracy and Producer Accuracy obtained on each class and Overall accuracy on 3D point cloud.

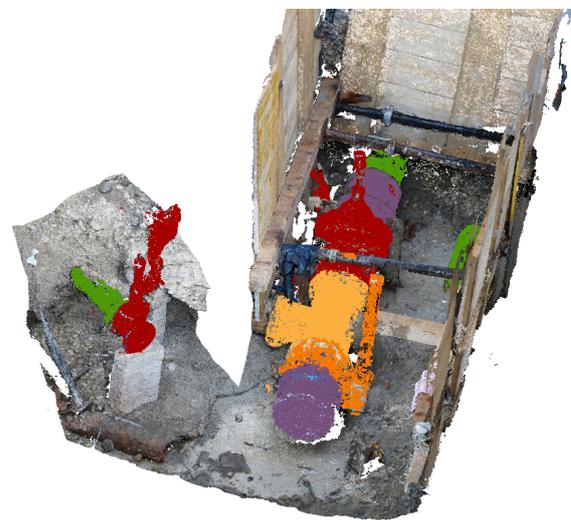
4. CONCLUSION

In this paper we presented a processing chain we developed to automate as much as possible the method of updating an underground cadastre.

The development of this processing chain required several different tasks, the first of which was the choice of an acquisition system. We opted for a photogrammetric solution (an SLR camera equipped with a GNSS antenna in RTK mode). Then we had to think about a method of automatic detection of the objects constituting the cadastre, we could go towards deep learning with 3D data (point cloud) directly or with image data. The development of deep convolutional neural networks is much more advanced



(a)



(b)

Figure 9: (a) Raw point cloud and (b) point cloud segmented by class overlaid on the raw point cloud.

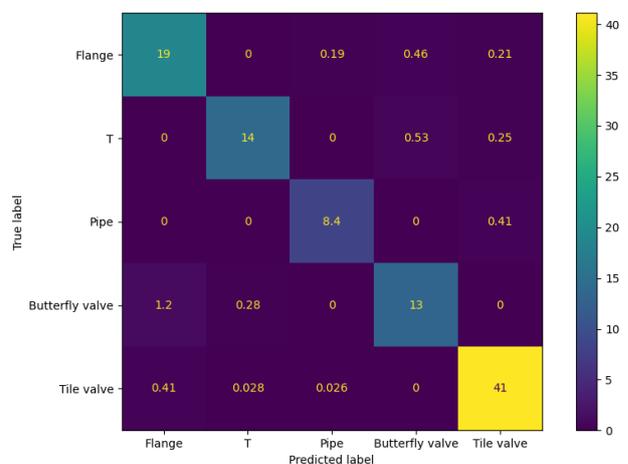


Figure 10: Normalized confusion matrix on 3D point cloud, each box corresponds to a percentage of the total data (3D points).

and older than networks for segmenting point clouds. We naturally turned to convolutional neural networks that allowed us to work with images. Thus, we had to build up a training database that we are still enriching today. In order to simplify image processing, we developed an infrastructure consisting of a web server linked to a cloud storage system that allows field teams to easily deposit and verify data. Photogrammetric processing to obtain images positioned and oriented in the Swiss coordinate system has been fully automated. The image labelling stage is perhaps the most important in the process, as its quality allows the convolutional neural network to be trained efficiently and accurately, and labelling such a large quantity of images manually is an extremely time consuming task. Therefore, we have developed a new semi-automatic labelling method that takes advantage of the 3D reconstruction and saves 60% of the time compared to a fully manual method. Next, the training of the convolutional neural network was done by transfer learning in order to save precious time; however, the hyperparameters were subject to a lot of tests in order to reach optimal results. Finally, we used a simple method already used previously to extract the 3D objects, consisting in using detection masks directly in a photogrammetric software (also an automated step).

Precision analyses were presented on the images and on the automatically labelled 3D point cloud (with 96.8 and 96.0 % average precision respectively). The step of updating the objects in vector form is still under development and will be the subject of a future publication.

5. PERSPECTIVES

The detection of rare objects remains a problem with this processing chain. A perspective would be to use generative adversarial networks (GAN) (Goodfellow et al., 2014) to create artificially real data. Augmentation methods allow for some improvement without additional time costs but are not sufficient to match the highly represented objects in the training database. In order to optimise the training of the convolutional neural network, it is planned to have access to a real calculation server with several graphic cards, which will make it possible not to be limited in the choice of certain parameters and to test several networks more quickly. A perspective that would change the processing chain presented here would be the use of a convolutional neural network directly in 3D (Qi et al., 2016, Qi et al., 2017) or on RGB-D data (Wang et al., 2021) that would theoretically take advantage of the 3D geometry as well as the colour. At present, these networks are much less developed than 2D convolutional neural networks.

The vectorization phase is still under development, in fact it requires comparing the old objects with the existing and requires specific processing because the geo-located data are sometimes distant from our new data (a few centimetres), we must be careful not to update objects that are still present.

ACKNOWLEDGEMENTS

This paper refers to a project carried out in collaboration with the Geneva Industrial Services (SIG) and funded by the National Geographic Data Infrastructure (INDG). We would like to thank Gaëtan Givord (team leader at SIG) for the management and organization of the project that led to all these interesting developments. Thanks to Maxime Fourcault who gave his time to label the 3D point clouds, an essential step for our processing chain. Finally, thanks to the SIG field teams who made it possible to collect all the training data.

REFERENCES

- Barsi, A., Kugler, Z., László, I., Szabó, G. and Abdulmutalib, H. M., 2018. Accuracy dimensions in remote sensing. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XLII-3*, pp. 61–67.
- Beniaouf, S. and Gressin, A., 2021. Suivi des Vignes par télédétection de proximité : le deep learning au service de l'agriculture de précision. Rapport de Travail de fin d'études ENSG.
- Braun, A., Tuttas, S., Borrmann, A. and Stilla, U., 2015. Automated progress monitoring based on photogrammetric point clouds and precedence relationship graphs. In: M. Malaska and R. Heikkilä (eds), *Proceedings of the 32nd International Symposium on Automation and Robotics in Construction and Mining (ISARC 2015)*, International Association for Automation and Robotics in Construction (IAARC), Oulu, Finland, pp. 1–7.
- Carreaud, A. and Gressin, A., 2021. Comparaison d'outils de levés en vue de la constitution d'une base de données d'apprentissage pour le relevé automatique de réseaux souterrains. 26th Gazette de l'IGSO.
- Carreaud, A., Lis, K. and Gressin, A., 2021. Fast image labelling using 3D reconstruction. *Swiss Geoscience Meeting 2021*.
- Cerioni, A. and Meyer, A., 2021. Swimming Pool Detection from Aerial Images over the Canton of Geneva (STDL).
- Crommelinck, S., Bennett, R., Gerke, M., Nex, F., Yang, M. Y. and Vosselman, G., 2016. Review of automatic feature extraction from high-resolution optical sensor data for uav-based cadastral mapping. *Remote Sensing*.
- Forlani, G., Dall'Asta, E., Diotri, F., Cella, U. M. d., Roncella, R. and Santise, M., 2018. Quality assessment of dsms produced from uav flights georeferenced with on-board rtk positioning. *Remote Sensing*.
- Goodfellow, I., Bengio, Y. and Courville, A., 2016. *Deep Learning*. MIT Press. <http://www.deeplearningbook.org>.
- Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A. and Bengio, Y., 2014. Generative adversarial nets. In: Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence and K. Q. Weinberger (eds), *Advances in Neural Information Processing Systems*, Vol. 27, Curran Associates, Inc.
- Hafiz, A. M. and Bhat, G. M., 2020. A survey on instance segmentation: state of the art. *International Journal of Multimedia Information Retrieval* 9(3), pp. 171–189.
- Kingma, D. P. and Ba, J., 2014. Adam: A method for stochastic optimization. *arXiv:1412.6980* Comment: Published as a conference paper at the 3rd International Conference for Learning Representations, San Diego, 2015.
- Ng, A., Bensouda Mourri, Y. and Katanforoosh, K., 2022. *Deep Learning specialization* — Coursera.
- Pellis, E., Murtiyoso, A., Masiero, A., Tucci, G., Betti, M. and Grussenmeyer, P., 2022. An image-based deep learning workflow for 3d heritage point cloud semantic segmentation. *The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences XLVI-2/W1-2022*, pp. 429–434.
- Picterra, 2022. *Geospatial insights for sustainable urban planning and management*.
- Qi, C. R., Su, H., Mo, K. and Guibas, L. J., 2016. Pointnet: Deep learning on point sets for 3d classification and segmentation. *arXiv preprint arXiv:1612.00593*.
- Qi, C. R., Yi, L., Su, H. and Guibas, L. J., 2017. Pointnet++: Deep hierarchical feature learning on point sets in a metric space. *arXiv preprint arXiv:1706.02413*.
- Wang, C., Wang, C., Li, W. and Wang, H., 2021. A brief survey on rgb-d semantic segmentation using deep learning. *Displays* 70, pp. 102080.
- Wu, Y., Kirillov, A., Massa, F., Lo, W.-Y. and Girshick, R., 2019. Detectron2. <https://github.com/facebookresearch/detectron2>.