Section a: Extended Synopsis of the scientific proposal

1 State-of-the-Art

The goal of the 'explorer' project is to develop methods for automatically *capturing* and *labeling* video data in "open worlds". Its ultimate goal is the great facilitation of the creation and maintenance of Digital Twins, which are virtual 3D copies of complex scenes such as cities, factories, or construction sites. Not just a 3D reconstruction, they should capture the scene's semantics, *i.e.* the identity of each object and the scene's dynamics, *i.e.* how objects move.

Figure 1: **A mockup of the explorer dataset.**

Digital Twins have the potential to be extremely useful for monitoring large complex sites and planning the development of these sites, and their market has been forecast to $48B for 2026 by Markets and Markets [1]. However, with the exception of a few anecdotal projects, they remain mostly a concept because of important limitations of the current technology. The capture of the raw 3D geometry of a scene is a problem that is now mastered, at least for static scenes, by using techniques from photogrammetry and computer vision applied to images, or using 3D sensors such as Lidars. The identification of each item, object, etc. remains much more challenging. In the computer vision field, this is referred to as "3D scene understanding". To give a better idea of what 3D scene understanding means for the Digital Twin of a construction site, Figure 1 shows a mockup of the dataset we plan to create in this project to validate our methods. We discuss below possible directions to solve 3D scene understanding, focusing on the context of Digital Twins.

Like most of the other computer vision topics, 3D scene understanding has significantly benefited from the development of data-driven methods and Deep Learning. Many recent works have considered static indoor scenes [3, 6, 15, 9], but other works in particular for autonomous driving consider dynamic outdoor scenes []. Such inference is very fast and can provide very impressive results.

However, such methods cannot perform correctly in "open worlds", where objects of unknown nature can appear. They rely on training data, which are examples of data annotated with the desired properties (3D shapes, 3D poses, etc.) of the objects, and they are limited to objects that have been considered by the creator of the training set. If an object does not belong to the categories appearing in the training set, it will simply be ignored by the supervised method, resulting in a failure.

Adapting to new environments—such as a new scene for the creation of its Digital Twin—is thus impossible without creating new training data including the objects specific to this scene. This would happen frequently for factories, which have often specific equipment, but it is virtually impossible for training sets to cover every single item that may appear in a real scene. In fact, existing datasets for 3D problems are limited to a dozen or less object categories.

This very small number of object categories in 3D datasets, especially compared to datasets for 2D problems which can be made of thousands of categories, is due to the complexity of creating new data with the desired properties (3D shapes, 3D poses, etc.) of the objects. The current options are:

- Manually capturing pictures of the objects of interest and labeling them also manually. 3D manual annotations are particularly cumbersome to create, and mistakes in manual annotations are actually common in existing datasets as we observed in [10].
- Manually capturing pictures of the objects of interest and labeling them with a combination of manual annotations and *ad hoc* methodology. For example, [**?**] asked professional annotators to point the 2D locations of specific 3D points in images of human bodies. Pointing in 2D is much easier than 3D input, but significant work is still involved.
- Creating synthetic data. This is a tempting approach as 3D labels are then obtained in a straightforward way. However, even if we ignore the fact that synthetically generated images still suffer from a "domain gap" with real data, creating these images has a high cost in terms of money and time. For example, [16] reports a total of $57K for scene creation and image rendering and 2.4 years of wall-clock rendering time to create a photorealistic synthetic dataset for indoor scene understanding.

Researchers have long recognized the critical importance of data and have already proposed many approaches for reducing the need for data, for example by transfer learning [], combining real and synthetic images [], learning generative models [**?**], exploiting multiple views of the same object [], learning features in a self-supervised way [], etc. However, these approaches still require some labeled data for each object category.

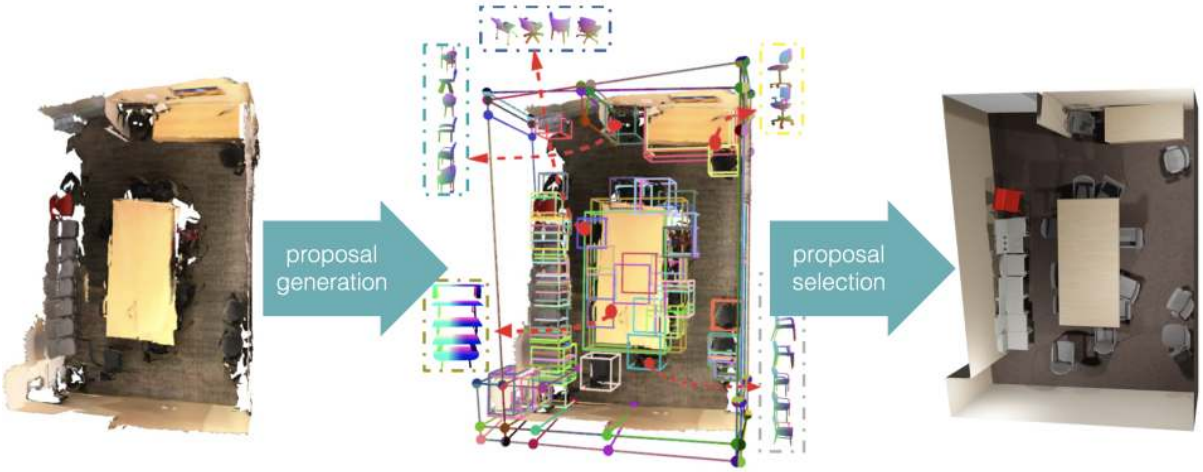

Figure 2: **Some of our recent work [10].** Here, we first generate "proposals" for the objects and walls using a simple supervised method applied to a given point cloud, and filter these proposals *jointly*. We retrieve all the objects and walls despite missing data. This two-step approach is robust: Even if the proposal generation generates many false positives, they will be filtered efficiently.

One may object that, since Digital Twins will be lucrative, it will be worth investing in the manual creation of data for new objects. However, 3D label creation currently does not just cost money but also takes time and expertise. Having to wait for label creation thus impairs reactivity as well.

We should also mentioned that "self-supervised methods" that can build models of unknown objects have also been proposed. They typically work by exploiting the redundancy between multiple images, sometimes with an analysis-by-synthesis approach [20, **?**]. They can obtain impressive results, given the fact they do not require human input. However, they are still limited to simple setups, with a single object present in the scene and center of attention in the images. They also require significant expertise when capturing the training data, which is a "hidden" human input, as we discuss below.

Data capture. So far, we have considered that data has been captured manually with the goal of creating the training set (except in the case of synthetic images). Such manual capture require expertise and time. Typically, training sets need to cover a diversity of possible object appearances from multiple viewpoints. The capture of such datasets is still often done by researchers who, consciously or unconsciously, capture data suitable for training supervised methods. However, it is not possible to expect a non-expert to be able to do the same. Capturing such data may also be time-consuming.

In the case of Digital Twins, the problem is exacerbated, as they would require a first capture to create a training set, and regular captures to update the state of the Digital Twin to guaranty it is a reliable copy of the real world.

To the best of our knowledge, there is no automated method for the physical capture of training data for computer vision applications. One may consider the data capture implemented on the cars of companies working on autonomous driving as such a method. However, such capture is very constrained as it is not controlled and can only be done from the users' cars. *For the explorer project, we envision the automated but active capture from mobile platforms*: We will develop methods to control ground-based robotic platforms and UAVs to capture data suitable for training supervised methods.

The closest concept to this automated capture in the literature is probably the "Next best view": Works looking for the "Next best view" aim to predict where to move a camera to take a shot of an object in order to reconstruct its 3D geometry reliably and accurately []. These works however consider one isolated object, most of the time in a simulator rather than in the real world. The automated capture of training data, as we plan to do it, will be done in open worlds made of multiple objects, which is much more challenging. Moreover, while it is relatively easy to come up with a criterion to optimize for the "Next best view" when aiming for the 3D geometry of an object, it is not clear which criterion should be optimize when capturing images for creating a training set.

1.1 State of our Research

We have been working on scene understanding for some time now []. As illustrated in Figure 2, we recently considered an approach where we first generate "proposals" about the objects present in the scene. At this stage, these proposals may actually correspond to an object or be wrong. The second step

filters the proposals, based on how well they explain the observations about the scene. In [10], which we published at CVPR 2021, we experimented with Monte Carlo Tree Search (MCTS) [7, 5] to perform this filtering. MCTS is a general discrete AI algorithm for learning to play games. For example, it is a key component of AlphaGo and AlphaZero [17, 18], which achieve super-human performance for several two-player games such as Go and chess. To the best of our knowledge, it was the first time MCTS was applied to a perception problem.

Using a supervised method, we first extract proposals for possible objects and walls from a point cloud. We then cast the proposal selection as a single-player game: Each move consists in selecting a proposal from the pool and the goal is to find select all the proposals that explain the scene. Introducing MCTS to the problem had many advantages over optimization methods used in earlier works for similar purposes, such as Conditional Random Fields (CRFs) [12, 21, 14] or Hill climbing [11, 23]: *With MCTS, we could optimize an objective function that does not have to have a special form, without the use of heuristics, and we could handle a large percentage of false positives among the proposals*, *i.e.* proposals that do not correspond to actual objects.

While this recent work allowed us to advance the state-of-the-art, it still suffers from the same limitations as the supervised methods mentioned above, since the proposals come from a supervised method: They cannot deal with "open worlds", where objects from new categories can appear. Moreover, we could not deal with dynamic scenes, to recover the objects' motions for example.

2 Objectives

We will develop non-supervised methods for guiding robotic platforms to capture visual data in open worlds, and for automatically label this data. Labels will be for example in the form of the 3D bounding boxes and models of the visible objects. These methods should be able to adapt to new environments—such as factories or construction sites—with objects unknown to the system. This is beyond the state-of-the-art, which is restricted to objects with prior information.

The objects may be rigid, but also articulated (say a bulldozer) or deformable (such as a cable). This is still very challenging as even supervised methods are still restricted to rigid objects. The objects may also have different scales (such as a bulldozer and a hammer). The ultimate goal is to significantly speed up the creation and maintenance of Digital Twins.

An important part of the explorer project is the development of algorithms able to move a robotic platform to capture the visual data for training purposes. To the best of our knowledge, this has not been attempted already. A camera will be mounted on a ground-based moving platform or a Unmanned Aerial Vehicle (UAV) and our algorithms will guide the platform or UAV to capture the training data automatically. This will relax the need for human expertise and time: Currently, capturing such data is done manually by researchers and requires strong understanding of what the learning algorithms require. It is also cumbersome, especially for complex objects of multiple sizes.

The visual data can be simply color images, or a 3D sensor can be joined to the camera to capture the depth data of the captured images. Depth data would be extremely useful, however Lidars are still heavy and expensive and cheaper depth sensors *e.g.* based on structured light do not work well outdoors. We will thus probably focus on color images only, but the algorithms we will develop can be adapted relatively easily to different types of data.

We will rely as little as possible on *a priori* knowledge and human input in general. By this, we do not just mean that we will not rely on manually labeled data. Human input can take the form of synthetic data. It can also be parameterized models of known objects [22]. We will refrain from requiring such input for new objects, as creating these input also demands a very significant effort. These are the conditions to truly adapt to new environments such as factories or worksites, for which little labeled data is available. The exception will be for common classes such as human bodies, for which large amounts of training data already exist and trying to learn a model from scratch would be pointless.

Our developments will need to be evaluated. To do so, we will first create a dataset. Many datasets of indoor scenes already exist and we will use then, but they do not cover most of the challenges posed by Digital Twins. We will create a dataset from working sites, to which we can have access thank to the link of ENPC Paris Tech with companies such as Vinci. To create the annotations, we will rely on 2D annotations created by a specialized company and develop a method to leverage these 2D annotations to 3D, to obtain annotations as suggested in Figure 1. To test the robotic part, we will create a simulated environment based on this dataset by integrating it into an existing simulator for robotics platforms.

3 Methodology

While our current work mentioned in Section 1.1 was still limited to "close worlds", in the sense that it cannot deal with unknown objects correctly, it has two key aspects that are very interesting:

1. By relying on MCTS to filter the object proposals, the requirements on the proposal generation becomes very light. MCTS can handle problems with high complexity, and when we applied it to scene understanding, it could handle and reject many proposals that are actually not in the correct solution. That means that the proposal generation method does not have to be particularly accurate. This is a strong starting point to handle open worlds.
2. While MCTS had never been used for perception problems, it has already been applied to control and planning [2]. This creates a bridge between perception and control, which we want to investigate to develop the approach for this project.

Looking deeper into the technical aspects, it is not MCTS *per se* that brings these key aspects:

- MCTS is based on the principle of "optimism in the face of uncertainty" to guide the search and fast simulations to rely on the objective function itself instead of heuristics. These are the main components that make MCTS scalable to highly complex problems. We can rely on the same components when developing algorithms that are more suitable to our problems.
- The optimization in MCTS proceeds by evaluating "moves". With [10], we realized that these "moves" do not have to be the moves in a game, but could be for example, the selection of a proposal. More generally, these moves can be of different nature for the same problem. By introducing new types of moves, we can combine perception and control in a unified manner.

We detail below how we plan to exploit these general observations for our goal.

1. Object proposal generation and representation. To fulfill our goal, we need to handle objects that can be unknown and articulated or deformable, which are still very challenging problems. The solution we consider is based on "2D proposals" extracted from the images, corresponding to the 2D reprojections of the objects.

While working on object discovery [8], we noticed than an object detector trained on a large variety of object categories tends to generalize *to some extend* to unknown objects. In other words, we can obtain the silhouettes of unknown objects in images, but some silhouettes will be incorrect and others will be missing. This observation however is a very good starting point when put together with the first key aspect mentioned above.

One limitation is that some objects can still be missed by the detector. To avoid this, one solution would be to add new proposals if something goes wrong in the proposal matching. This could be detected for example by considering which parts of the input are not explained by any good proposal, and using this feedback to try and detect more proposals in these parts.

We also want to infer the 3D geometry of the unknown, moving objects, including of objects that can be moving or articulated. This is also still a challenging problem in the case of real images and unknown categories. In our case however, we have as input a video sequence to exploit rather than a single frame. Moreover, we can generate multiple proposals and let our filtering algorithm select the correct one. We can also introduce a continuous optimization step to refine the predicted shapes. We will investigate all these directions.

2. Novel filtering algorithms. We will develop an algorithm that matches the 2D proposals that correspond to the same objects across images while rejecting the incorrect ones. This algorithm will be inspired by the components of MCTS that make it scalable, however it will depart from the standard MCTS. For example, MCTS was designed to run on trees. Such structure is required in games as moves are performed in a sequence but irrelevant for proposal matching. Instead of a tree with a fixed structure, we could use a list of matches sorted according to how promising each proposal is.

It will also be important to study the complexity of the algorithm we will develop. The factors that influence this complexity include the percentage of false positives in the pool of proposals, and the constraints between proposals. The required properties of the objective function should also be studied, in terms of continuity for example, as was done in [13].

3. Automatic data capture. The two first points (Proposal Detection and Representation and Filtering Algorithms) can be developed and evaluated offline on precaptured video sequences. To control a robotic platform to capture the video sequences, we need to define and formalize a criterion that should be optimized by moving the platform. For example, we could aim to minimize ambiguities between

possible solutions of the filtering algorithm. To evaluate ambiguities, a principle way is to estimate a distribution over these possible solutions. We can notice that by exploring several possible solutions, MCTS and our future algorithms will produce a distribution over the solution space: Each evaluated solution becomes a sample and its objective function value tells us how likely this sample is. From this, it should be possible to identify which motion should relax the ambiguities. A straightforward solution would be to move to viewpoints that are very different from those that generate the ambiguities, but more sophisticated options could be identified, maybe with reinforcement learning.

Moreover, we should integrate planning and perception, as there will be feedback between the motions taken by the platform and new data that will keep coming up. We will extend our algorithm developed for offline videos into an algorithm that optimizes our criterion that combines "perception moves" for matching proposals and "control moves" for moving the platform.

The research direction described above should be developed first on a ground-based mobile platform because such platforms come with more computational power—and break less easily than flying drones. But flying drones are still very interesting as they are a good solution for capturing data efficiently outdoor. Can we use our approach not just for generating 3D labeled data, but also for generating training data for learning to move and capture data efficiently?

4 High-risk / high-gain nature of the project

We identified the following risks:

- Dealing with unknown objects in real images is still extremely challenging. To generate proposals, we need to detect them, segment them in images but also predict their 3D geometry from images. This is even more difficult for articulated or deformable objects. This is however the conditions to tackle open worlds. This risk is maybe mitigated as discussed at the beginning of Section 3 by the light requirements on this procedure, which is part of our strategy. This is thanks to the filtering step, however this step has its own risks, see next point.
- About the filtering step, it is possible that the complexity of the problem becomes too high. We will have to deal with large, complex, dynamic scenes and if too many proposals are incorrect, it is possible that our algorithms will not be able to identify the correct ones. We will need to develop powerful algorithms, maybe by taking into account the 3D nature of the problem.
- The active capture is the most risky part. Active capture of training data has never been attempted before, and will require to combine perception and planning.
- Our lab has a worldwide recognition in computer vision, however it is not a robotic lab while a large part of the project relies on robotics. In fact, there are not so many labs that possess top-level expertise in both robotics and computer vision. Fortunately, software environments of robotics platforms are getting better and better, and make development relatively easy even for non-robotics experts. Members of our lab have already a direct experience in flying drones for SLAM [4]. An important part of the future of computer vision will most likely be about combining robotics and computer vision, and this is the right time to start working on it.

If successful, the explorer project will have a strong impact:

- We will introduce a novel research problem, the active capture of training data, which is crucial for large-scale Digital Twins;
- We will introduce a novel methodology to solve this problem;
- Beyond Digital Twins, our general approach and the algorithms we will develop are original and should inspire other researchers for their own problems;
- Our project will make the creation and maintenance of Digital Twins, which are particularly important for the future of industrial development, significantly easier.

5 Resources

I will dedicate 70% of my time to the explorer project. Moreover, two permanent researchers from the Imagine lab at ENPC will also be part of the project: Drs David Picard and Pascal Monasse, both at the level of 20% of their time. Dr Picard is an expert in reinforcement learning and MCTS; Pascal Monasse is an expert in 3D capture, in particular from UAVs. In France, by law, PhD students' contracts have a fixed duration of 3 years. We request 5 PhD students in total. Two will work on the development of the offline algorithms, and 3 will work on the active capture. In addition, three postdocs (3 years each) will help me supervise the PhD students along the project. We also request resources from paying the annotation company involved in the creation of the dataset, and for buying a robotic platform.

References

[1] Digital Twin Market by Technology, Type (Product, Process, and System), Application (predictive maintenance), Industry (Aerospace & Defense, Automotive & Transportation, Healthcare), and Geography – Global Forecast to 2026. https://www.marketsandmarkets.com/Market-Reports/digital-twin-market-225269522.html. 1

[2] Tesla. , 2016. 4

[3] Armen Avetisyan, Manuel Dahnert, Angela Dai, Manolis Savva, Angel X. Chang, and Matthias Nießner. Scan2CAD: Learning CAD Model Alignment in RGB-D Scans. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 1

[4] Thomas Belos, Pascal Monasse, and Eva Dokladalova. MOD SLAM: Mixed Method for a More Robust SLAM Without Loop Closing. In *International Joint Conference on Computer Vision, Imaging and Computer Graphics Theory and Applications (VISAPP)*, 2022. 5

[5] Cameron B. Browne, Edward Powley, Daniel Whitehouse, Simon M. Lucas, Peter I. Cowling, Philipp Rohlfshagen, Stephen Tavener, Diego Perez, Spyridon Samothrakis, and Simon Colton. A Survey of Monte Carlo Tree Search Methods. *IEEE Transactions on Robotics and Automation*, 2012. 3

[6] Yixin Chen, Siyuan Huang, Tao Yuan, Siyuan Qi, Yixin Zhu, and Song-Chun Zhu. Holistic++ Scene Understanding: Single-View 3D Holistic Scene Parsing and Human Pose Estimation with Human-Object Interaction and Physical Commonsense. In *International Conference on Computer Vision*, 2019. 1

[7] Rémi Coulom. Efficient Selectivity and Backup Operators in Monte-Carlo Tree Search. In *International Conference on Computers and Games*, 2006. 3

[8] Yuming Du, Yang Xiao, and Vincent Lepetit. Learning to Better Segment Objects from Unseen Classes with Unlabeled Videos. In *International Conference on Computer Vision*, 2021. 4

[9] Alexander Grabner, Peter M. Roth, and Vincent Lepetit. 3D Pose Estimation and 3D Model Retrieval for Objects in the Wild. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018. 1

[10] Shreyas Hampali, Sinisa Stekovic, Sayan Deb Sarkar, Chetan S. Kumar, Friedrich Fraundorfer, and Vincent Lepetit. Monte Carlo Scene Search for 3D Scene Understanding. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2021. 1, 2, 3, 4

[11] Hamid Izadinia, Qi Shan, and Steven M. Seitz. Im2cad. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017. 3

[12] Hema Swetha Koppula, Abhishek Anand, Thorsten Joachims, and Ashutosh Saxena. Semantic Labeling of 3D Point Clouds for Indoor Scenes. In *Advances in Neural Information Processing Systems*, 2011. 3

[13] Rémi Munos. Optimistic Optimization of Deterministic Functions without the Knowledge of its Smoothness. In *Advances in Neural Information Processing Systems*, 2011. 4

[14] Claudio Mura, Oliver Mattausch, and Renato Pajarola. Piecewise-Planar Reconstruction of Multi-Room Interiors with Arbitrary Wall Arrangements. In *Computer Graphics Forum*, 2016. 3

[15] Yinyu Nie, Xiaoguang Han, Shihui Guo, Yujian Zheng, Jian Chang, and Jian Jun Zhang. Total3DUnderstanding: Joint Layout, Object Pose and Mesh Reconstruction for Indoor Scenes from a Single Image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 1

[16] Mike Roberts and Nathan Paczan. Hypersim: A Photorealistic Synthetic Dataset for Holistic Indoor Scene Understanding. In *arXiv Preprint*, 2020. 1

[17] David Silver, Aja Huang, Chris J. Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, and Others. Mastering the Game of Go with Deep Neural Networks and Tree Search. *Nature*, 529(7587), 2016. 3

[18] David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, Timothy P. Lillicrap, Karen Simonyan, and Demis Hassabis. A General Reinforcement Learning Algorithm That Masters Chess, Shogi, and Go through Self-Play. *Science*, 2018. 3

[19] Sinisa Stekovic, Mahdi Rad, Friedrich Fraundorfer, and Vincent Lepetit. MonteFloor: Extending MCTS for Reconstructing Accurate Large-Scale Floor Plans. In *International Conference on Computer Vision*, 2021.

[20] Andrea Vedaldi. http://www.vlfeat.org/~vedaldi/code/siftpp.html. 2

[21] Shenlong Wang, Sanja Fidler, and Raquel Urtasun. Holistic 3D Scene Understanding from a Single Geo-Tagged Image. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015. 3

[22] Sergey Zakharov, Wadim Kehl, Arjun Bhargava, and Adrien Gaidon. Autolabeling 3D Objects with Differentiable Rendering of SDF Shape Priors. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2020. 3

[23] Chuhang Zou, Ruiqi Guo, Zhizhong Li, and Derek Hoiem. Complete 3D Scene Parsing from an RGBD Image. *International Journal of Computer Vision*, 2019. 3