D3.1 VISUAL ANALYSIS FOR REAL SENSING

Marios Krestenitis, Konstantinos Ioannidis, CERTH

@AshvinH2020

ASHVIN H2020 Project

www.ashvin.eu

ASHVIN has received funding from the European Union’s Horizon 2020 research and innovation programme under Grant Agreement No 958161. This document reflects only the author’s view and the Commission is not responsible for any use that may be made of the information it contains.

Project Title	Assistants for Healthy, Safe, and Productive Virtual Construction Design, Operation & Maintenance using a Digital Twin
Project Acronym	ASHVIN
Grant Agreement No	958161
Instrument	Research & Innovation Action
Topic	LC-EEB-08-2020 - Digital Building Twins
Start Date of Project	1st October 2020
Duration of Project	36 Months

Name of the deliverable	ASHVIN technology demonstration plan
Number of the deliverable	D3.1
Related WP number and name	WP 3 Visual analysis for real sensing
Related task number and name	T3.1
Deliverable dissemination level	PU
Deliverable due date	30-09-2022
Deliverable submission date	30-09-2022
Task leader/Main author	CERTH
Contributing partners	Marios Krestenitis (CERTH), Ioanna Mpouziona (CERTH), Konstantinos Ioannidis (CERTH), Stefanos Vrochidis (CERTH), Mary Lidiya Kalathiparambil Kennedy (TUB)
Reviewer(s)	Irina Stipanovic (INFCON), Sasa Klopanivic (MFL), Rahul Tomar (DDT)

ABSTRACT

D3.1 Visual analysis for real sensing [lead: CERTH; due: M24]. Deliverable D3.1 consolidates the progress achieved inT3.1 task. The main purpose of the document is to report all the algorithms that have been deployed for extracting features of the constructions and comprise the baseline for the higher level of implementations.

The report is divided into three distinct sections. First, it presents the developments made with respect to the 3D representation pipeline that were deployed and applied to demo sites #1, #4, #6 and #7, which included bridges and industrial buildings. These include tools for Structure from Motion and Dense 3D point cloud generation on images captured in ASHVIN demo sites. Furthermore, a single image 3D depth prediction pipeline is presented.

Secondly, the approach and implementation carried out to develop an AI-based defect detection service with pixel segmentation is presented. The aim was to detect and pixel segment different types of defects that are present in realistic inspection scenarios in demonstration site #3, which included airport operational areas. Convolutional neural network architectures were trained and validated.

Finally, the report presents the results of the training and implementation of a state-of-the-art object detection algorithm to detect objects at construction sites for monitoring the construction progress. The implemented model was applied to images obtained from demo site #4 (construction of industrial building) and is based on YOLO v5 detector.

KEYWORDS

Image-based 3D representation, Defect Detection, Object Detection, Semantic Segmentation.

REVISIONS

Version	Submission date	Comments	Author
V0.1	18/05/2022	Creation of ToC	Konstantinos Ioannidis
V0.2	29/07/2022	1st Draft Version	Ioanna Mpouziona, Marios Krestenitis
V0.3	10/08/2022	2nd Draft Version	CERTH
V0.4	30/08/2022	Results were integrated	CERTH
V0.5	10/09/2022	Section inserted	TUB
V0.6	23/09/2022	Internal Review	MFL, CERTH
V0.7	27/09/2022	Internal Review	DDT, CERTH
V0.8	30/09/2022	Internal Review	INFCON, CERTH

DISCLAIMER

This document is provided with no warranties whatsoever, including any warranty of merchantability, non-infringement, fitness for any particular purpose, or any other warranty concerning any information, result, proposal, specification or sample contained or referred to herein. Any liability, including liability for infringement of any proprietary rights, regarding the use of this document or any information contained herein is disclaimed. No license, express or implied, by estoppel or otherwise, to any intellectual property rights is granted by or in connection with this document. This document is subject to change without notice. ASHVIN has been financed with support from the European Commission. This document reflects only the view of the author(s) and the European Commission cannot be held responsible for any use which may be made of the information contained.

ACRONYMS & DEFINITIONS

2D	Two-dimensional
3D	Three-dimensional
DT	Digital Twin
DoA	Description of Action
CV	Computer Vision
DSP	Domain-Size Pooling
SfM	Structure From Motion
SLAM	Simultaneous Localisation and Mapping
MVS	Multi-View Stereo
RGB	Red Green Blue
ROI	Region Of Interest
SIFT	Scale-Invariant Feature Transform
UAV	Unmanned Aerial Vehicle
3DRI	3D representation from Images
FLANN	Fast Library for Approximate Nearest Neighbors
ANN	Approximate Nearest Neighbors
RANSAC	RANdom SAmple Consensus
BA	Bundle Adjustment
mAP	Mean Average Precision

ASHVIN PROJECT

ASHVIN aims at enabling the European construction industry to significantly improve its productivity, while reducing cost and ensuring absolutely safe work conditions, by providing a proposal for a European wide digital twin standard, an open-source digital twin platform integrating IoT and image technologies, and a set of tools and demonstrated procedures to apply the platform and the standard proven to guarantee specified productivity, cost, and safety improvements. The envisioned platform will provide a digital representation of the construction product at hand and allow to collect real-time digital data before, during, and after production of the product to continuously monitor changes in the environment and within the production process. Based on the platform, ASHVIN will develop and demonstrate applications that use the digital twin data. These applications will allow it to fully leverage the potential of the IoT based digital twin platform to reach the expected impacts (better scheduling forecast by 20%; better allocation of resources and optimization of equipment usage; reduced number of accidents; reduction of construction costs). The ASHVIN solutions will overcome worker protection and privacy issues that come with the tracking of construction activities, provide means to fuse video data and sensor data, integrate geo-monitoring data, provide multi-physics simulation methods for digital representing the behaviour of a product (not only its shape), provide evidence-based engineering methods to design for productivity and safety, provide 4D simulation and visualization methods of construction processes, and develop a lean planning process supported by real-time data. All innovations will be demonstrated on real-world construction projects across Europe. The ASHVIN consortium combines strong R&I players from 9 EU member states with strong expertise in construction and engineering management, digital twin technology, IoT, and data security / privacy.

TABLE OF CONTENTS

INDEX OF FIGURES

INDEX OF TABLES

1 INTRODUCTION

With the advent of affordable and high-quality smartphone cameras, fixed cameras, unmanned aerial and ground vehicles, there is a unique opportunity to massively record digitally and analyse the entire life cycle of construction environments.

Computer vision is an interdisciplinary scientific field that deals with how computers interpret and understand visual data. Examples of such applications range from self-driving cars, safety monitoring, to automating quality control and increasing production efficiency.

As a result of its growth (Chai, Zeng, Li, & Ngai, 2021) (Feng, Jiang, Yang, Du, & Li, 2019), computer vision has attracted research interest in construction in recent years on monitoring construction work, improving time-consuming and repetitive tasks and bringing the physical labor in digital twin technologies for construction. A variety of applications using computer vision were emerged in the field, such as automated progress monitoring and resource allocation by detecting progress deviation through a comparison to a BIM model, optimising equipment usage, improving quality assessment and granting safety monitoring, for instance, by examining the condition of concrete through a video or photographic images. The rapid growth of deep learning is opening a new era in digital applications for scaling-up and automating existing solutions as well as developing new ones based on the capabilities of automatically discovering and formulating features needed for classification (LeCun, Bengio, & Hinton, 2015).

In ASHVIN Task 3.1 the goal is to deploy computer-vision based algorithms for processing the collected visual data (images or/and videos) from demonstration construction sites and provide a higher level of understanding to feed specific tools of the ASHVIN toolkit. There are 3 main methods deployed in the framework of this task:

The first work aims to the deployment of the **3DRI** method. As quoted in D7.1 "*This method will introduce a pipeline for estimating 3D structures from 2D imagery. The depth information is calculated from 2D data using common information that is present in overlapping parts between different images or videos".*

The second task developed under T3.1 is a defect detection approach to be applied to images recorded on runways, which refers to the DDCV method. "*The AI-powered solution is used to detect damages, anomalies and objects on the runway surface and green areas around the runway. The aim is to integrate the automated damage detection into inspection and maintenance planning process*"[1]. The deployed service processes the visual input, acquired via UAV camera, and produces an annotated mask where the detected defects are segmented accordingly.

The third task related to computer vision and implemented under T3.1 concerns the monitoring of construction activities. An object detection algorithm was implemented to detect the mounting precast columns at the demonstration site #4. The service processes time lapse images to detect the duration of installation activities during the construction phase.

[1] D7.1 " ASHVIN technology demonstration plan"

1.1 Purpose and document structure

This deliverable report all the work and the algorithms deployed for extracting features and using visual data from demonstration sites within ASHVIN project to support as methods the higher-level tool components of ASHVIN project.

The document is structured into 3 main sections. Section 2 is about the 3D point cloud generation from images which refers to the main corpus of the work described in T3.1 of the DoA. Two additional sections derived from the interaction with the ASHVIN end users, to cover the needed requirements set by the implementation of the demo cases. Therefore Section 3 is describing the work around the deployment of the **DDCV** method, which aims to enable the condition monitoring of airport infrastructure in Zadar airport. Finally, Section 0 elaborates on the object detection algorithms that were applied to detect pile installation for demonstration site #4. The main objective of this work was to automate the process through a sequence of time-lapse images to extract the duration of pile mounting and feed the DES tool developed in T4.2.

1.2 Sensors

The quality of the collected visual data has the most impact on the performance of the system. For example, an efficiently designed and trained object detection module may significantly underperform if the input images are of poor quality (e.g., blurred or collected from awkward angles). Blurred images cause problems not only in scene recognition and spatio-temporal building and object techniques but also in photogrammetric 3D reconstruction, mainly because the amount of matched or tracked features drops dramatically during the feature matching process.

A visual sensor is a device composed of at least an RGB camera, a storage unit, an energy supply and a communication interface. Image resolution, brightness, contrast, compression factor and colour scheme are some of the characteristics that impact the performance of visual sensors, but such impact depends on the applications monitoring requirements. In the context of ASHVIN there were considered the below options to cover the needs of the data collection process in the programme demo sites. In addition, an evaluation of this sensing equipment is provided and their characteristics are listed and discussed in the following subsections.

1.2.1 Hand-held cameras (GoPro)

GoPro cameras have been considered as a candidate sensing element, see Figure 1, since it has a fish-eye lens which can capture a wide field of view and, in record mode, it has a high frequency rate (above 60Hz), allowing to record videos without blur. Therefore, some frames can be extracted as post-processing from these videos instead of continuously take pictures that can be time consuming. Generally, the following should be considered:

Figure 1: GoPro camera

- Capture images with good texture. Good texture provides unique features on the object surface, which are necessary for matching them to different images and estimating their 3D position.
- Capture images in similar lighting conditions. By avoiding high dynamic range scenes (e.g., pictures against the sun with shadows or pictures through doors/windows). Avoid specularities on shiny surfaces. The feature matching across images becomes significantly harder with strongly varying illumination and correspondences might not been found.
- Capture images with high visual overlap. Videos and image sequencing ensure that each object is shown in multiple frames – the more images the better. Overlap ensures that many visual features are available for matching features and subsequently estimating the relative camera motion.
- Capture images from different viewpoints. Do not take images from the same location by only rotating the camera, e.g., make a few steps after each shot. At the same time, try to have enough images from a relatively similar viewpoint. Different viewpoints are important to see the same surface several times to estimate feature locations in 3D and to reduce the number of unobserved areas due to occlusions.

1.2.2 Aerial Cameras as Drone payload

Drone recordings allows to apply incremental techniques for the 3D representation of the external geometry of the construction by post-processing of the acquired pictures.

The advantages of this technology are listed below:

- High quality and low-cost data: one can obtain recording of an area within a few minutes of flight, allowing the acquisition of high value information and high precision images.
- Metadata such as GPS location and IMU sensor recordings are also made available, improving the camera localization and speeding up the 3D representation process.
- Safety: Drone recordings and associated image data post-processing technology has brought to the construction industry a very powerful tool for data capturing and site survey, reducing the time spent collecting accurate data. By acquiring aerial imagery, it become easier to collect millions of high accuracy data points per flight also improving safety since there is no need to deploy personnel directly to hazardous or inapproachable areas.

Data collection process: the key element for a successful 3D representation from 2D input is the collection of the data set. A guidance document was delivered early in the project to the WP7 partners to guide them on how to collect image data. These guidelines for data collection include the following key points:

- High overlap between images;
- Image acquisition plan, which depends on the type of object to be reconstructed, which in turn depends on: Path planning, Flight height, Camera angle(s)
- Ground Control Points (GCPs) to improve and/or validate the accuracy of the georeferencing.

One type of drone that was considered for aerial scanning and could be useful for the construction industry is the Skydio drone [2] followed by the 3D scan toolkit. This drone offers adaptive scanning capabilities that allows for automated data capture tailored to the needs of 3D generation processes.

For demo sites #1,6,7 the drone used for the data capturing was the DJI Mavic Air 2.

Table 1: Air Mavic 2 Camera characteristics

Sensor	1/2" CMOS Effective Pixels: 12 MP and 48 MP
Lens	FOV: 84° Equivalent Focal Length: 24 mm Aperture: f/2.8 Focus Range: 1 m to ∞
ISO	Video:100-6400 Photo (12 MP): 100-3200 (Auto)
Max Resolution	48 MP 8000×6000 pixel
Photo Models	Single: 12 MP and 48 MP Burst: 12 MP, 3/5/7 frames

The instructions followed for camera settings have an impact on image quality and by extension lead to quality post-processing. For processing, the images should be sharp and have the least amount of noise. As a thumb rule, the following should be respected:

- Shutter speed should be fixed and set to medium speed (between 1/300 second and 1/800 second), but fast enough to not produce blurry images.
- ISO should be set as low as possible (minimum 100). High ISO settings introduce noise and reduce the quality of the results.
- Aperture depends on the lens and it is better to leave it on automatic.

The instructions given to end-users are set out on project's repository. Typical overlap percentage in such applications is around 80% in both directions (sometimes even higher percentage up to 90% may be applied) to ensure that all object points are depicted in multiple photos. This leads to better triangulation accuracies in 3D reconstruction. At the same time occlusions and hidden areas should be avoided.

1.2.3 Hi-Res Webcam.

Timelapse is probably one of the most used and efficient ways to immortalize several months of work on a construction site and to showcase the final result. Using timelapse

[2] https://www.skydio.com/

video, construction project monitoring becomes a powerful communication tool that can be shared on social media or broadcast. For Demo site #4 a timelapse video from a fixed camera is the only input of visual content information that is provided, constituting an important value for construction documentation. Moreover, one of the objectives in demo site #4 is to capture time-lapse high-resolution images from a fixed position to record the evolution of the construction process. And this was the only element to motivate our work for image processing to allow the 3D representation of the space and for real-time project progress tracking.

The estimation of 3D geometry from a single image is a special case of image-based 3D representation from several images, but is considerably more difficult since depth cannot be estimated from pixel correspondences due to great information loss from 3D to 2D. With the rise of neural networks and deep learning, neural networks have been deployed that could be trained to learn the three-dimensional structure of objects in a single image. There are no specific requirements set for this type of collected dataset other than the provision of high-quality images and their provision at regular time intervals.

Table 2: Camera characteristics options for the TimeLapse fixed camera installed in Demo site #4

Resolution	Image Quality	Reliability	Connections
23 Megapixel	Color reproduction	Self-sufficient	Cellular Broadband UMTS|LTE
35 Megapixel	Contrast	Maintenance-free	LAN / Wi-Fi
+60 Megapixel	Day + Night	Function monitoring	DSL / VDSL

Figure 2: Type of Hi-Res Webcam installed in demo site #4 [3]

[3] https://www.hi-res-cam.com/de/

2 3D POINT CLOUD GENERATION FROM VISUAL CONTENT

This section contains information about the solutions, workflows and algorithms used and developed in ASHVIN for the 3D representation of project demo cases using visual data. Details about the generation of 3D representations for the Pilot Demo Sites (PDS 1, 4, 6 & 7) are also given. 3D representation is performed on relevant visual data of the environment such as raw video footage, time lapse images, georeferenced imagery gathered by drones, that are collected during the data collection process performed by demo site leaders and still in process for some demo cases. The collection of data is described in the corresponding deliverables related to the Demo Sites. The main objective is the production of 3D point clouds of variant resolutions in ".ply" format depending on the needs of the 4DV-C construction monitoring tool (Figure 3). The generated 3D point clouds of outdoor and indoor spaces via imagery content will be one of the building elements of the Construction Monitoring tool needed for the Continuous monitoring of the as-build process. The 3D models (and the raw data) will be available:

During the last years, there has been a growing interest around the use of three-dimensional representation in various fields ranging from autonomous driving to augmented reality, which rely heavily upon accurate 3D reconstructions of the surrounding environment. The 3D reconstruction process captures the geometry and appearance of a single object or an entire scene.

Automated 3D point cloud generation from images has been one of the key problems of computer vision for years (Furukawa, 2014); for construction sites, it might be just the beginning to make use of these digital information. The goal of image-based 3D representations is **to infer the 3D geometry** and **spatial relationship from 2D images**. This long standing ill-posed problem is fundamental to many applications such as robot navigation, object recognition, scene understanding, 3D modelling and industrial control etc (Han, Laga, & Bennamoun, 2019).

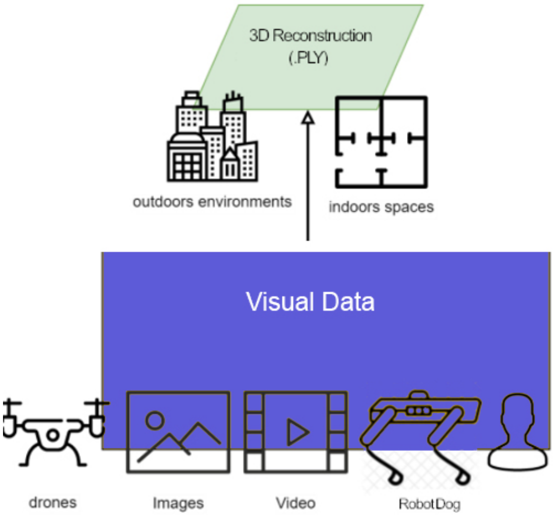


Figure 3:Schematic overview of the 3D representation method.

2.1 Objectives

The objectives of T3.1 (M1-M24) are aligned with the main goals that are described in the DoA and summarised as follows:

- **Analyze and evaluate the main equipment** that will be utilized as a means of acquisition. Digital Cameras, time lapse Hi-Res webcam, UAVs and GoPro cameras (that would be installed on the robot dog for Demo Site #5) were assessed to be integrated with the main system and comprise the basic means of sensing the real world. Moreover, a document with basic instructions related to data collection have been prepared early in the project and handed to WP7 leader to guide the data collection process.
- **Identify the computer vision algorithms** to be deployed for extracting accurate features from the 2D representations of the captured scenes. Descriptors well as higher level local descriptors were deployed to initially extract the required features.
- **Image enhancement algorithms** were evaluated as a pre-processing stage for optimal feature extraction and mitigating visual inaccuracies in any image/video caption.
- **Cameras' calibration** is needed as a prerequisite for optimal depth estimation.
- The objective of the task is to translate 2D images into their 3D reflectances.
- The process aims at achieving high accuracy initial data that will be forwarded to a higher level of implementations for producing the "Digital Twin" of the interested object.
- All image processing methods developed in this task will be integrated with the ASHVIN platform in close collaboration with T1.1.

As described in D7.1 "Ashvin technology demonstration plan", the 3DRI method will perform the following task:

Table 3: Plan for 3DRI method as described in D7.1.

<table>
<tr><th>ASHVIN tool/method</th><th>Name</th><th>How the method will be used on the project</th></tr>
<tr><td>3DRI (method)</td><td>3D Representation from Images</td><td>This method will introduce a pipeline for estimating 3D structures from 2D imagery. The depth information is calculated from 2D data using common information that is present in overlapping parts between different images or videos.</td></tr>
<tr><td colspan="2"><b>Data Input</b></td><td><b>Data output</b></td></tr>
<tr><td colspan="2">Image files from photo captures.</td><td rowspan="3">3D point clouds in .PLY format</td></tr>
<tr><td colspan="2">Geotagged Image sequences and Video from drones.</td></tr>
<tr><td colspan="2">Time lapsed images</td></tr>
</table>

As can be derived from Figure 4, the first step in a generic flow that involves the 3DRI method consists of new visual data being sent into to the 3D representation service. The new data can originate ideally from the IoT platform (T1.1) from recordings performed on demo sites by end-users.

Once a new piece of data is successfully ingested to the 3DRI service, the data is analysed using the Image-Based 3D representation pipeline that will be discussed in more detail on a following section. The inferred 3D point cloud in ".ply" format is the ready to be handed to the final receptor, the 4DV-C tool.

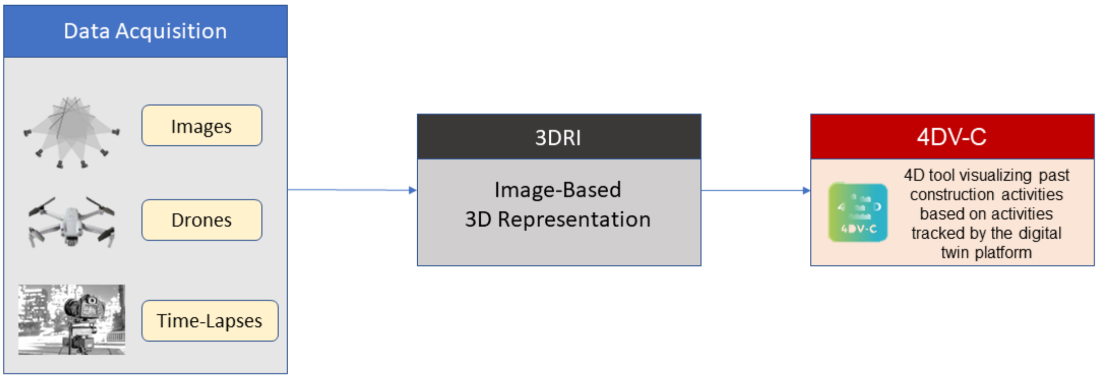

Data Acquisition
Images
Drones
Time-Lapses
3DRI
Image-Based
3D Representation
4DV-C
4D tool visualizing past construction activities based on activities tracked by the digital twin platform

Figure 4: High level view of visual data acquisition and data processing and the position of 3DRI method in ASHVIN project.

2.2 Related Technologies

The problem of reconstructing the 3D representations of the environment has drawn significant attention from many researchers over the past few decades. Reconstruction techniques are often consisted by algorithms that are fusing depth measurements from special sensors, such as LiDAR, RGB-D or structured light into 3D models. While these sensors can be extremely effective, they require special hardware making them more cumbersome and costly than systems that rely only on RGB cameras.

Most commonly found is the use of LiDAR scanners to reconstruct the 3D representation. The development of such devices has allowed the fast and accurate 3D recording of complex environments. However, 3D scanning is time consuming and the captured point clouds often contain noise due to failures of reflection of the laser beams, for example on shiny, metal or glass surfaces which presence is common in construction sites. Additionally, 3D scanning is still considered expensive due to the high cost of using and maintaining 3D laser scanner devices.

The image-based approach, which is the subject of the work developed under T3.1 and reported in this deliverable, is generally considered as a low-cost method, flexible, portable and capable of reconstructing 3D representations simply using images. In the last decades different solutions have become available for the automated processing of images and the derivation of 3D information and models. The processing mainly includes image orientation and dense 3D reconstruction with a large level of automation.

ASHVIN project integrates most of the above-mentioned capturing methodologies to create 3D representations. These include: (a) laser scanning suitable for detailed reconstruction of large-scale objects and (b) photogrammetry and computer vision techniques that exploit visual information using Simultaneous Localization and Mapping (SLAM) and Structure from Motion (SfM) techniques. The use of unmanned aerial vehicles (UAVs) was also foreseen for the 3D reconstruction of large-scale building environments (Demo site #6) and bridges infrastructures (Demo site #1 &#7).

Techniques for solving image-based problem come from both computer vision and robotic research communities by means of Structure from Motion (SfM) and visual

Simultaneous Localisation and Mapping (VSLAM). Standard SfM and VSLAM aim to simultaneously estimate the camera pose and 3D structure of the scene through a set of feature correspondences detected from multiple images.

Popular pipelines providing 3D image reconstruction are COLMAP (Schonberger & Frahm, 2016) , MeshRoom (MR) (Meshroom), OpenSfM, ODM and Capturing Reality (CR). The cornerstones of these pipelines are SfM and MVS, where the latter builds on the results of SfM. Moreover, many commercial solutions exist for undertaken this process: ArcGIS by ESRI, ContextCapture by Bentley Systems, Correlator3d by Simactive, Inpho by Trimble, iWitnessPRO by Photometrix, Metashape by Agisoft, Pix4DMapper by Pix4D, PFTrack by ThePixelFarm, RealityCapture by Epic Games, ReCap by Autodesk and Zephyr by 3DFlow and many more.

Despite the plethora of the existing frameworks for 3D representation, the majority of the current methods doesn't provide a complete pipeline including data pre-processing, image enhancement and an integrated process from video/image sequences to the creation of the final point cloud output.

In the following subsections we analyse in brief the open-source pipelines used to support the work deployed under T3.1 and act as a research basis:

2.2.1 COLMAP (ETH)

COLMAP (Schonberger & Frahm, 2016) is a general-purpose Structure-from-Motion (SfM) and Multi-View Stereo (MVS) pipeline with a graphical and command-line interface. It offers a wide range of features for reconstruction of ordered and unordered image collections. The software is licensed under the GNU General Public License (open-source), and is maintained by a member of ETH.

Related to the theories, Colmap detects keypoints in each image whose appearance is described by numerical descriptors. Pure appearance-based correspondences between keypoints/descriptors are defined by matches, while inlier matches are geometrically verified and used for the reconstruction procedure.

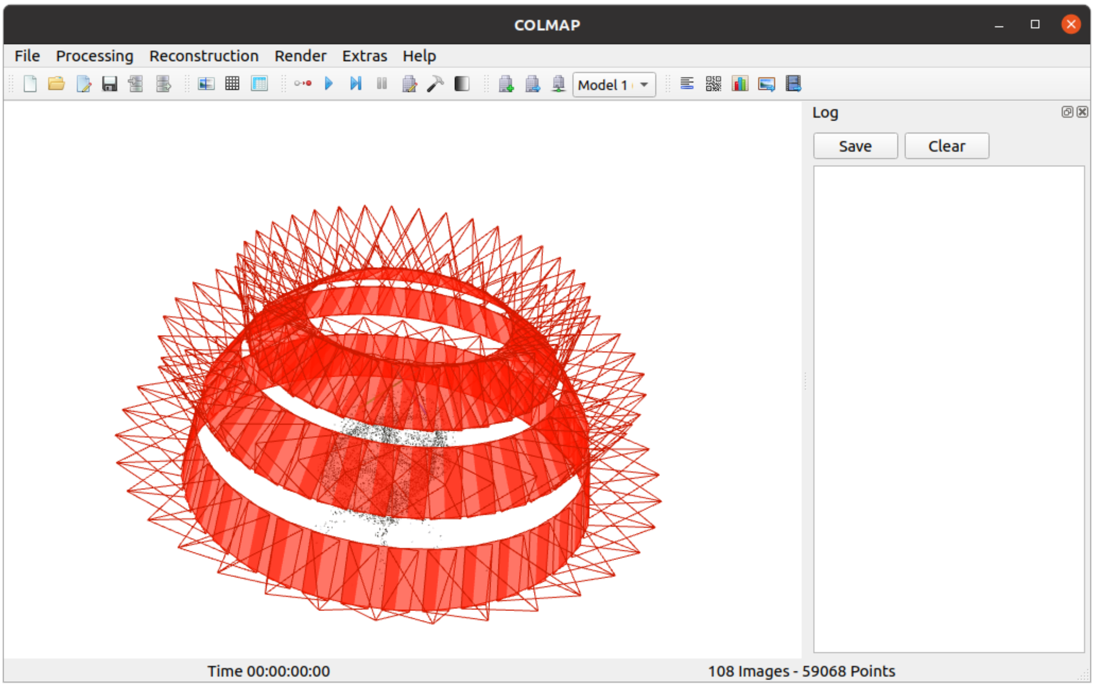


Figure 5: COLMAP interface

Colmap provides a command-line Interface, where more available options can be given by delivering diverse flags. Also, a scripting language interface for python is available to alter parameter or running additional custom fitting and matching algorithms.

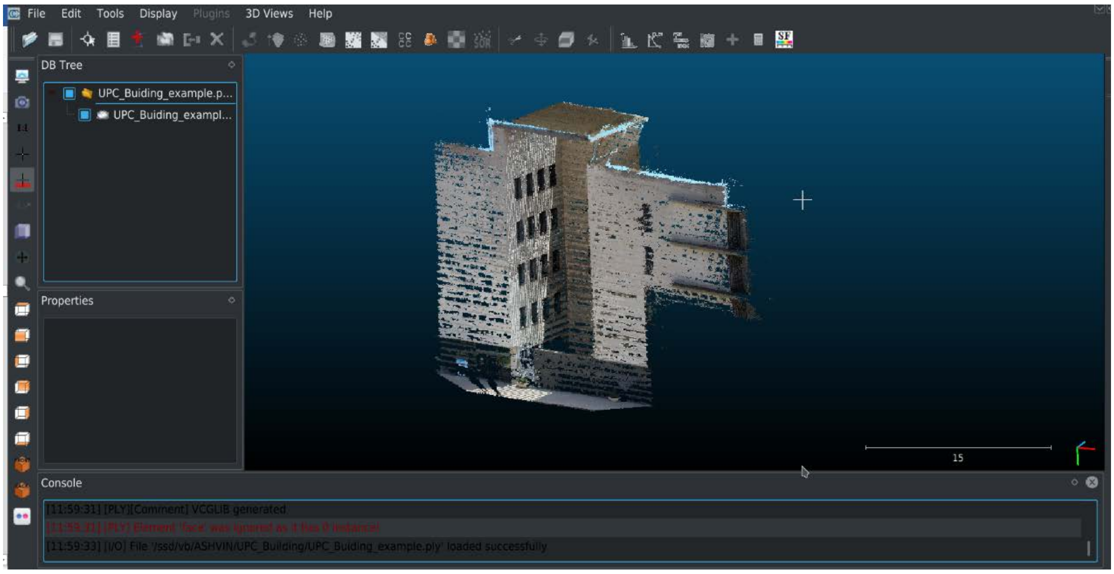
2.2.2 Hloc - the hierarchical localization toolbox

Hloc is a modular toolbox for state-of-the-art 6-DoF visual localization. It implements Hierarchical Localization (Sarlin, Cadena, Siegwart, & Dymczyk, 2019), leveraging image retrieval and feature matching, and is fast, accurate, and scalable. Hloc is working with learned SuperPoint features and SuperGlue (Sarlin, DeTone, Malisiewicz, & Rabinovich, 2020) to establish 2D-3D matches with a SfM model. The hloc toolbox also provides D2-Net (Dusmanu, et al., 2019) single CNN technique for feature detection and description.

The use of hloc was for running Structure-from-Motion with SuperPoint+SuperGlue and employing in the pipeline more updated feature matching techniques, more oriented to the challenging environments that are captured in construction sites, with low illumination, dynamic conditions for indoor and outdoor visual data collection.

As a basis, the hloc toolbox include multiple pipelines: one pipeline called pipeline_Aachen.ipynb, applied to the Aachen Day-Night dataset (Sattler, et al., 2018)and one called pipeline_InLoc.ipynb, applied to the InLoc dataset (Taira, et al., 2018). Both serve as examples of an application of the hloc pipeline, and to show its performance. The third pipeline is called SfM_pipeline. In the example, it is applied to the South_Building dataset [4],but it can easily be applied to another set of images. All three pipelines share some "backend" scripts, but the first two utilize some scripts specific to their respective datasets.

2.2.3 CloudCompare



```
[11:59:31] [PLY][Comment] VCGLIB generated
[11:59:31] [PLY] Element 'face' was ignored as it has 0 instance!
[11:59:33] [I/O] File '/ssd/vb/ASHVIN/UPC_Building/UPC_Buiding_example.ply' loaded successfully
```

Figure 6: The interface of CloudCompare

CloudCompare is an open-source 3D point cloud processing software, which can display large dense point-cloud smoothly and even perform comparison between two

[4] https://colmap.github.io/datasets.html

point-clouds. In this study, we mainly use this software to check the fused output from Colmap. Compared with another software MeshLab is CloudCompare in this study apparently faster and its optional rotation centre is very handy since the main content of point cloud locates usually not in the centre of the coordination due to some big outliers.

CloudCompare provides a set of basic tools for manually editing and rendering 3D points clouds. It also offers various advanced processing algorithms, among which methods for performing:

- projections (axis-based, cylinder or a cone unrolling)
- point cloud registration.
- distance computation (cloud-cloud or cloud-mesh the nearest neighbor distance)
- statistics computation (spatial Chi-squared test, etc.,

2.2.4 ODM

Web-ODM is a commercial-grade open-source software for drone image processing. Based on the open-source command line toolkit Open Drone Map (OpenDroneMap Authors, 2020), it can also be used across Linux, macOS, and Windows OS. The 3D representation of the area is saved in the server. The user can view the 3D model via the web-odm application. ODM software was used as a viable solution for processing visual datasets captured from demo site #3 (Section 3) and translating them into orthomosaics.

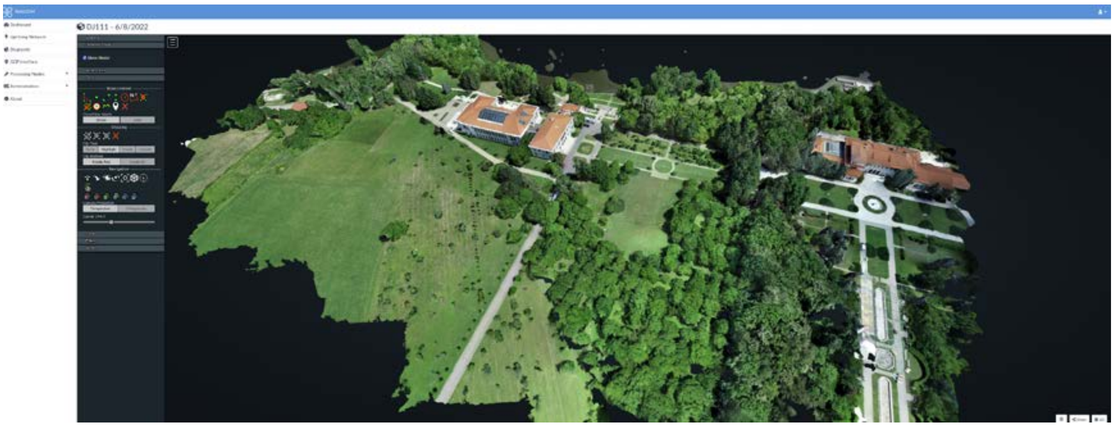


Figure 7 Offline 3D map on web application

2.3 Relation to ASHVIN demonstrations and motivation

ASHVIN project aims at the digitalisation of the construction industry. To achieve this goal, digital twin technology is one means to this end. One of the core developments of ASHVIN is to introduce important innovations in the process, enabling the ability to closely match the as-built information with the as-designed information.

Due to the simplicity and flexibility the image-based reconstruction can bring benefits to ASHVIN's goals. The 3D representation is considered as a core element for the digital twin since it offers an efficient and affordable way to bring the geometry of the physical world in a digital representation. Using images a dense point cloud can be calculated, which is similar to the point cloud measured directly using laser scanners

The following subsections outline the demonstration sites where the 3DRI method is currently engaged, as well as a description of the available data and the scope of the analysis performed. In the use cases below only monocular images/videos are recorded as source of visual information and there is also one case of time lapse images recorder (Demo Site #4) for which we follow a different strategy to develop the 3D representation workflow.

2.3.1 Demo Sites #1 & #7

High-definition cameras attached to drones can scan the construction sites and create a digital 3D model with real-time updates. This way construction project managers can have more visibility over the project without the hassle and risk of physical inspection.

The **demonstration sites #1 & #7** are particularly focused on maintenance and monitoring activities of bridges infrastructures. The challenge is to develop a Digital-twin enabled reproduction of the assets that generate an impact on cost reduction and safety at operational stages. In both demo sites, among other data, were also collected images and video footages from drone flights.

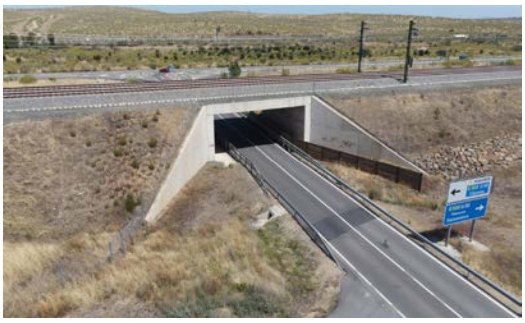


Figure 8: Demo site #1 : Underpass located at the 3 + 93 PK.

With respect to the 3DRI method developed under T3.1 the aspect examined in the context of **Demo Site #1** is the use of drones to scan bridges autonomously or with either reduced human supervision to produce and deliver digital 3D representations for the ASHVIN digital twin environment. An important drawback around the bridges of Demo Site #1 is that plans are only available in pdf format and assets is not easy to be digitalised and brought into a digital twin platform. Image-Based 3D representation could provide an affordable solution to this problem, although usually cannot generate the millimetre-level dense detail of a laser survey. It can also be more versatile than laser scanning. Unlike tripod-based laser scanners, digital cameras are compact and can be capture data easily in outdoor environments where laser scanners is difficult to be carried.

In **Demo site #7** the objective is the monitoring of the PR-04-B015 bridge which is located within the Metropolitan Area of Barcelona (Spain). The bridge connects two main road axes: the AP-7 Highway (heading North) and the A-2 Road (Heading West) and is consisting of two separated viaducts. Both viaducts are supported by 12 piers with varying span. In the context of ASHVIN we examine the aspect of using drones and photogrammetry methods to capture and reconstruct the 3D representation of the piers and introducing a cost affordable method for translating the constructed infrastructure into a digital format.

Figure 9: Bridge of Demo Site #7 construction.

With the mature of the Unmanned Aerial Vehicle (UAV), the cost of a drone reduces significantly, and real-time monitoring thus becomes much easier. Using drone to monitor construction site is a flexible approach with low manpower costs, and it can easily reach the site that is sometimes too hard for human. This working mode might replace even more traditional manpower with the development of technique and computation. The image-based reconstruction, as one of the various techniques, keeps updating by improving its accuracy and efficiency to help realize a more reliable digital construction.

2.3.2 Demo Sites #6

The **demonstration site #6** is focused on the monitoring of the construction activities. Demonstration buildings #6 are located in Barcelona (Spain) and they are part of project 22@, also known as 22@Barcelona and “Innovation district” (“Districte de la innovació”). The challenge is to develop and to propose digital solutions and digital twin implementations through the life cycle of the construction processes mainly around the concrete maturity based on readings from sensors.

Moreover, in the context of this demo is examined the aspect of replacing the traditional mapping and surveying methods by digital approaches and rely more on modern technology and digital solutions. Despite recent advances, the prevailing monitoring and management systems in the construction industry are still dominated by traditional approaches, including manual paper- based collection and recoding of on-site activities. With the use of photogrammetry and fully digitized data, less time can be spent in the field and highly representative data along with 3D visualization for the generation of interior and exterior representations.

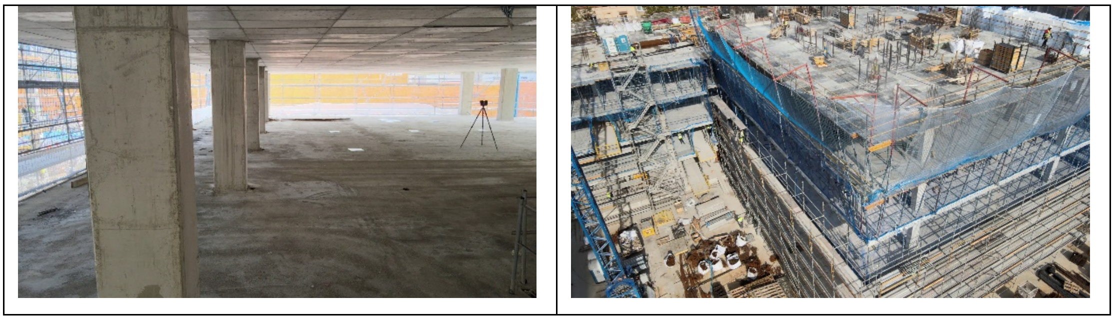


Figure 10: Demo Site #6 construction site a) indoor and b) outdoor drone recordings.

2.3.3 Demo Site 4

The demo site #4 is an industrial building located in Rinteln, Germany. Overall, the building has a size of nearly 30,000 m², while halls 1 and 2 utilise one third. The structure of the building consists mainly of prefabricated components such as precast concrete pillars, steel walls, and steel roof panels. A fixed Hi-res camera was installed on site to capture images of the construction progress every 10 minutes,

Again, the motivation for T3.1 and the **3DRI** method was the 3D representation from the given visual input. In contrast to multi-view algorithm, single-image 3D representation algorithms are not as accurate, since it is impossible to produce an accurate 3D reconstruction from a single 2D image due to the loss of information that occurs by projecting a 3D scene onto a 2D plane. Therefore, the problem of 3D reconstruction of a single view can be more accurately described as a 3D estimation or 3D prediction problem. A single image 3D depth prediction service was deployed integrating state-of-the-art neural network architectures.

2.4 Methodology

This section will introduce the key concepts that will help in understanding the technologies behind the 3D reconstruction methodology used to obtain the data described in this document. In task 3.1, CERTH is responsible to exploit visual data from different sources that are provided by the data collection performed by Demo Site leaders, in order to translate 2D images into their 3D reflectance.

Recovering the lost dimension from 2D images has been the goal of multiview stereo and Structure-from-X methods (Laga, 2019). The objective in these methods is to require matching features across images captured from slightly different viewing angles, and then use the triangulation principle to recover the 3D coordinates of the image pixels.

According to the publications, the process of generating point clouds from monocular images generally consists of seven steps, i.e., feature extraction, feature matching, camera localisation, sparse 3D reconstruction, model parameters correction, absolute scale recovery and dense 3D reconstruction. The combination of the above steps is called Structure from Motion (SfM) (Yang, Chao, Huang, Lu, & Chen, 2013). Each step needs to use the result obtained from the previous steps as the input.

Classical SfM pipelines (Sweeney, 2016) (Fuhrmann, Langguth, & Goesele, 2014) first extract and match sparse features. Usually, an initial transformation between pairs of cameras (essential matrix) is estimated with RANSAC. Given the initial camera transformations, a geometric verification stage evaluates photometric consistency between re-projected sparse features and excludes outliers. Starting from an initial two-view reconstruction, an incremental reconstruction is performed based on best view selection, triangulation, and bundle adjustment. Simultaneous Localization and Mapping (SLAM) methods also address the problem of joint camera estimation (ego-motion) and 3D scene reconstruction. However, SLAM techniques focus primarily on accurate egomotion estimation and real-time performance, typically sacrificing geometric accuracy.

Schonberger and Frahm (Schonberger & Frahm, 2016) proposed a structure-from-motion pipeline with better completeness and accuracy while better reducing drift in comparison to previous methods. They further propose a more robust best view selection and triangulation method, producing more complete structures. Finally, an iterative Bundle Adjustment, retriangulation, and outlier filtering step lead to significantly more complete and accurate 3D models.

The methodology used under the scope of T3.1 is based on the above technique. The aim is to estimate the camera positions corresponding to a set of input images and simultaneously recover a sparse 3D representation of the captured scene. Finally, the dense reconstruction step aims to recover the details of the scene, by using the images of a certain scene, the intrinsic and extrinsic parameters of cameras for each image and the sparse point cloud obtained from the previous step to generate a dense point cloud.

The reconstruction challenges arise in the presence of weakly textured areas, uniformly colored areas, scene transitions from dark to light (e.g., the light turns on or off or the camera aperture changes in the dark corridor), or reflective and repetitive surfaces. The mentioned problematic properties of the scene often directly lead to a **low number** of generated image matches. In addition, they cause poor match distribution in the image, as most of the challenging areas are almost matchless using methods such as a combination of SIFT and geometric verification using RANSAC-based model estimator. This lack of matches then negatively affects the accuracy of position estimates of registered cameras. The camera position inaccuracies then also decrease 3D point positions' precisions.

The inputs of the techniques for image-based 3D reconstruction usually are monocular images, stereo images or video frames, corresponding to monocular cameras, binocular cameras, video cameras respectively. Monocular images, stereo images and video frames have different characteristics from each other. Stereo images contain monocular images in pair, while video images contain a series of monocular images, or a series of stereo images depending on the recording setup. In our work, based on the visual data captured from the construction sites the process deployed regarded only input from monocular images. As stated in the previous section and is materialized for the deployment process, there are two different pipelines supporting the needs of the ASHVIN use cases. The first is dedicated to the visual data input coming from monocular videos and images sequences captured in most cases by drone assets and the second is a CNN technique deployed for the prediction of the depth dimension and eventually the point cloud generation based on single view images from a time-lapse fixed camera.

2.4.1 Monocular Images 3D representation pipeline

Below is presented a common framework for 3D representation from monocular images sequences or videos:

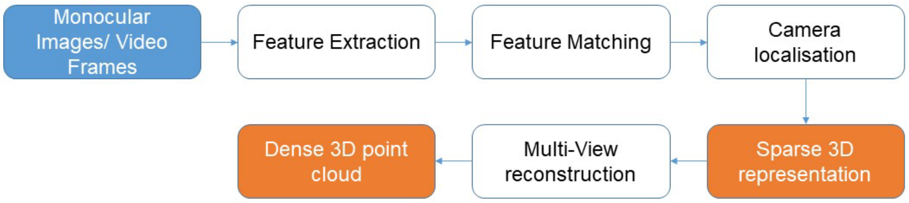


Figure 11: Generic Framework for 3D representation from monocular images or videos

The aim of this work is to create an automatic pipeline that can be used by practitioners on the construction sites to extract 3D representations from captured images that overcome the challenges that arise in construction sites. The basis of this work Is to explore the mechanism of classic algorithms in CV, such as Scale-Invariant Feature Transform (SIFT), Structure-from-Motion (SfM) and Multi-View Stereo (MVS).

In the section 2.6 is explained the work performed for each of the nodes indicated in the above workflow.

2.4.2 Single-View depth prediction pipeline

Complementary to the main 3D representation service described above another one was deployed as a means to generate quickly 3D content from single images, mainly of outdoor environments. This is separate branch of the 3DRI pipeline where a depth map and 3D coloured pointcloud can be obtained of an image. This is presented as a separate service and not as part of the main pipeline since it is not an accurate deterministic 3D representation method, but rather a way to “estimate” by using a trained network the underlying geometry depicted in an image. Further details for the work performed to exploit on different approaches published and the resulting of the final outcome is reported in more detail on a following section 2.7.

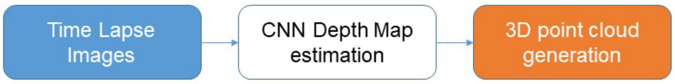


Figure 12: Overview of the single view depth map prediction workflow

2.5 Camera Calibration

Photogrammetric 3D representation is built on the principle of resection, in which the intersection between the projected rays from different viewpoints, depicted on 2D images, is computed. To implement these two pieces of information is needed, the camera parameters. The camera parameters are composed of extrinsic and intrinsic and the process to find them is called camera calibration.

Camera calibration is the analytical procedure of determining the camera’s internal parameters including the principal distance, format size, principal point, and lens distortion coefficients. Camera calibration is generally performed by means of coded targets or checker-boards, in order to achieve higher accuracy in the tie point identification and camera parameters estimation.

Typically, this means recovering two kinds of parameters

1. Internal parameters of the camera/lens system. E.g., focal length, optical center, and radial distortion coefficients of the lens.
2. External parameters: This refers to the orientation (rotation and translation) of the camera with respect to some world coordinate system.

The goal of the calibration process is to find the 3×3 matrix K, the 3×3 rotation matrix $\mathbf{R}$, and the 3×1 translation vector $\mathbf{t}$using a set of known 3D points (X_w, Y_w, Z_w) and their corresponding image coordinates (u, v). When we get the values of intrinsic and extrinsic parameters the camera is said to be calibrated.

2.5.1.1 Method

For the calibration step a simultaneous estimation of camera and projector calibration along with their relative orientation is performed. The implemented algorithm includes:

- The projection of a chessboard pattern onto a planar object and the recording of these projections by the camera. This is repeated for different successive orientations of the planar surface.
- Automatic detection of corners on the imaged chessboard patterns.
- Having determined image-to-pattern point correspondences and initial parameter values, an iterative bundle adjustment is carried out for estimating camera intrinsic matrix.

This process is a prerequisite for both pipelines that follows in sections 2.6 & 2.7. in the SfM approach we use the camera parameters for better results of the final product when this information is known. Otherwise, the process is based on focal guess.

2.6 Monocular Images pipeline

As discussed at the introductory section our strategy for this pipeline to optimize existing workflows adding automatic Image enhancement capabilities, filtering and masking of non-relevant classes, and exploit custom learning-based methods for feature extraction to deploy a pipeline using Colmap. The framework adopted to build our approach is the open-source software − Colmap. We perform the reconstruction separately in mainly two stages, i.e., sparse model and dense model, as divided by Colmap. Figure 13 shows the workflow followed in our study

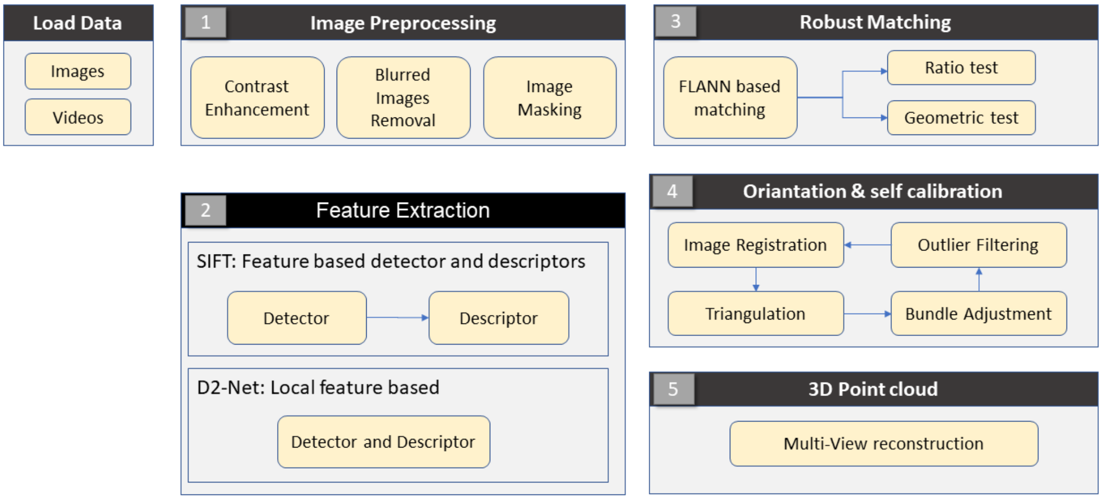


Figure 13: Overview of the image-based 3D representation process followed by 3DRI method in ASHVIN.

The visual content is introduced into the stream as a set of images or videos and the method sequentially analyses these videos or images into a complete set of frame instances. A script was written the ingestion process that also searches whether the corresponding videos, typically from drone shots are followed by metadata to parse them into corresponding EXIF fields of each analysed image.

Then each image is subjected to an image enhancement pre-processing which is detailed in the following section:

2.6.1 Image Pre-processing.

In T3.1 of ASHVIN project we focus on the 3D representation from various sources of visual data, videos and images. The photogrammetric pipeline is targeted towards the processing of images of the same scene. Optionally, these images should be taken from different viewpoints and should adhere to several important constraints that were listed and described in previous sections. Photogrammetric pipelines typically don't focus on processing video, or preprocessing the images or video frames. In video cases, the user is typically instructed to extract separate frames from the video and feed these to the photogrammetric package. There are some important issues with this strategy. The most important relates to the quality of the extracted frames, especially those taken in indoor spaces or with bad weather conditions. Also, the extraction of optimal frames (keyframes) for the photogrammetric processing is standing problem.

2.6.1.1 Image Enhancement

When video sequences are recorded, even using high-end drone technology and the aforementioned guidelines for image and video capturing, the possibility of blurred or under-illuminated frames cannot totally be excluded. Also, the dark construction environments lead to image color distortion and reduce the resolution and the contrast of the observed object in outdoor scene acquisition. **Image enhancement** can be defined as conversion of the image quality to a better and more understandable level for feature extraction or image interpretation.

2.6.1.1.1 Related work

In the deep learning era, several approaches have been introduced for image enhancement. In (Wang, et al., 2019), a convolutional neural network has been proposed where the authors introduce an illumination layer in their end-to-end neural network for under-exposed image enhancement, with the estimation of an image-to-illumination mapping for modeling multiple lighting conditions. The work of (Li, Guo, Porikli, & Pang, 2018) has proposed a trainable CNN for weak illumination image enhancement, called LightenNet, which learns to predict the relationship between illuminated image and the corresponding illumination map. A feature spatial pyramid model has been proposed in (Song, Huang, Cao, & Song, 2022) with a low-light enhancement network, in which the image is decomposed into a reflection and an illumination image and then are fused to obtain the enhanced image. A GLobal illumination-Aware and Detail-preserving Network (GLADNet) has been proposed in (Wang, Wei, Yang, & Liu, 2018). The architecture of the proposed network is split into two stages. For the global illumination prediction, the image is first downsampled to a fixed size and passes through an encoder-decoder network. The second step is a reconstruction stage, which helps to recover the detail that has been lost in the rescaling procedure. In contrast, the proposed approach is an end-to-end process and it is able to preserve the structure and texture information through a wavelet pooling transformation.

2.6.1.1.2 Method

In the framework of ASHVIN a novel method has been integrated deployed by CERTH to remove the noise from such images, sharpen edges and reveal details in textured regions. To achieve photorealism of the synthetic image, a model is expected to recover the structural information of a given image and enhance it effectively. To achieve this, a U-Net-based network is combined with wavelet transformations and Adaptive Instance Normalization (AdaIN). More specifically, the image recovery is addressed by employing wavelet pooling and unpooling, in parallel preserving the information of the content to the transfer network. Dense blocks are used to enhance the quality of feature transferring and skip connections in the transferring process. Intending to a natural stylization effect, the stylized features are inserted into the image reconstruction process.

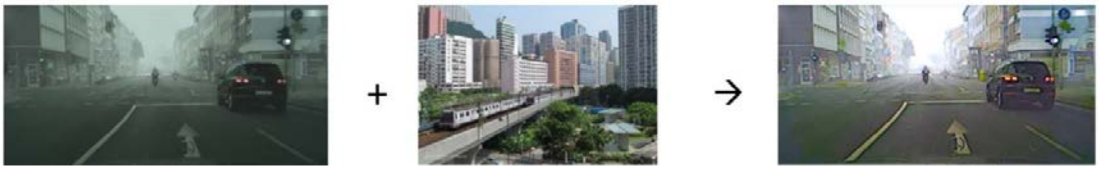


Figure 14: Example results of the enhancement method on a pair sample of the test set data..

The proposed under-exposed images enhancement achieves a natural stylization effect and similar numbers with SoA but with less resources. This work is under submission process, therefore specific details for the network architecture and the pipeline deployed cannot be reported in this deliverable.

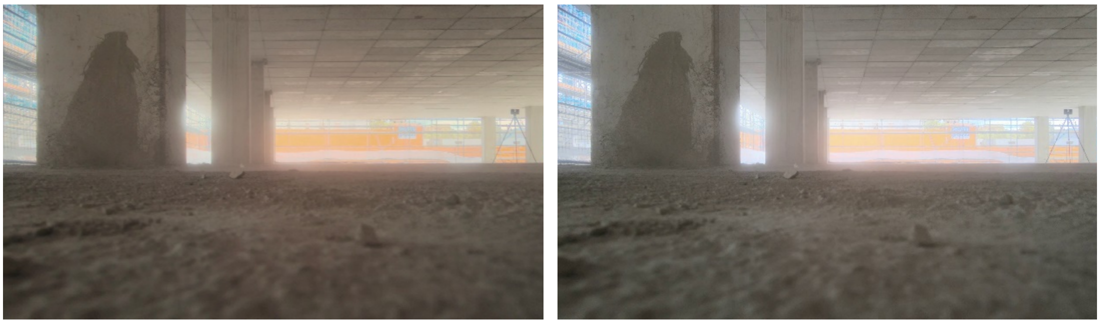


Figure 15: Image enhancement example on ASHVIN data recorded on Demo Site #6 by the use of Mavic Air 2 - Drone.

2.6.2 Image (not) Masking

In photogrammetry software there is the option to mask in or out the areas of interest or the non-relevant areas. The masking process is usually done manually by drawing the object area and saving the binary mask in TIF format. Masks define the areas to be processed in white and shorten the processing time by processing only the areas of interest. In ASHVIN we explore an automatic way of masking out the non-relevant classes, which usually cause noise in the generated final product.

2.6.2.1.1 Related work

Deep learning techniques have enormous success solving both image classification and segmentation problems. Image semantic segmentation has the goal to assign semantic labels to every pixel in the analysed image. Fully Convolutional Networks for Semantic Segmentation, presented by (Long, 2015), popularized the use of end-to-end convolutional networks and introduced skip connections from higher resolution feature maps. Another encoder-decoder architecture was proposed by (Peng, 2017) which includes very large kernels convolutions, but these large kernels convolutions are computationally expensive and they are adopted because networks tend to gather information from a smaller region. DeepLabV2 network (Chen L.-C. a., 2017) is an architecture for semantic segmentation that builds on DeepLab (Chen L.-C. , 2014) with an atrous spatial pyramid pooling scheme. New versions of it have been proposed, DeepLabV3 (Chen L.-C. e., 2017), which improves upon DeepLabv2 with several modifications, and DeepLabV3+ (Chen et al, 2018), which, in turn, extends the previous one.

2.6.2.2 Method

Based on DeepLabV3 architecture CERTH has deployed an algorithm to remove unwanted elements, such as “sky”, “people”, “electric poles”, from the collected datasets as a pre-processing step to facilitate the 3D reconstruction process that follows. To achieve this, we have deployed a semantic segmentation model pre-trained on the ADE20K dataset [5]. Each analysed image or video frame is properly masked to keep only the information needed." , see example in Figure 17.

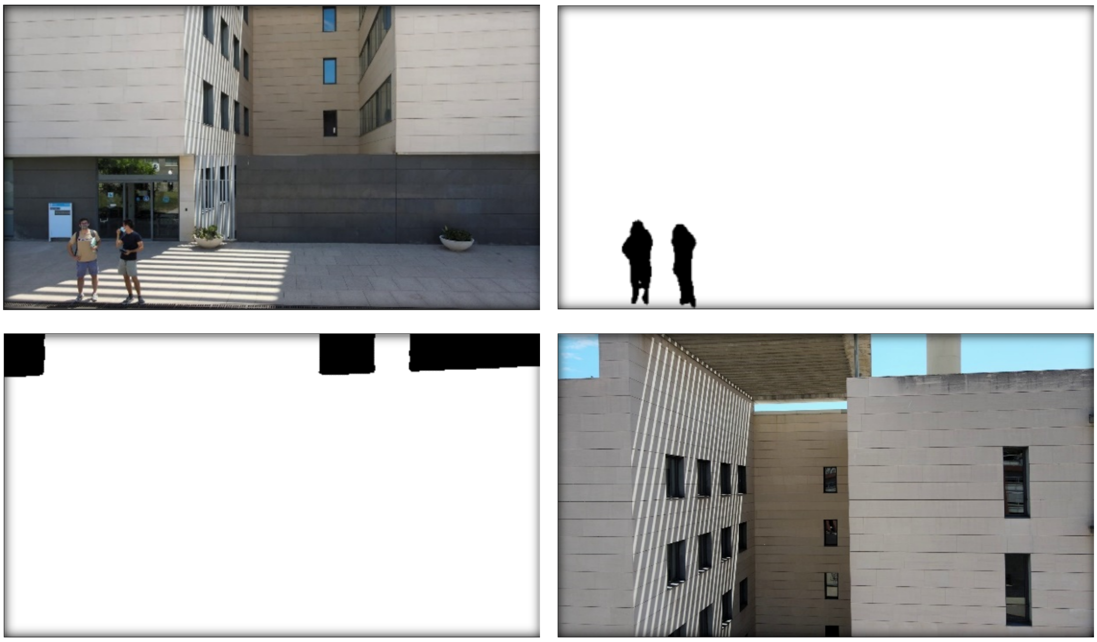


Figure 16: Masking results of the DeepLabV CNN algorithm: initial images, people and sky masks. The example is taken from the UPC building dataset which was captured for testing purposes.

The output masks can serve various purposes. First, they can be used to speed up the reconstruction since they limit the number of features used for the solving of the 3D representation problem. Secondly, they can be used to enhance the sparse and dense reconstruction by filtering outliers in the point clouds. For instance, the removal of faraway points on moving clouds, humans or electric poles can aid in the production of proper point cloud data. This in turn results in significantly better dense point clouds with less noise. Finally, the masking of people offer an extra level of security of people private info that may be present in the construction sites operating their daily tasks

2.6.3 Feature Descriptors from 2-D images

In this step, a structure from motion scheme was deployed which operates on successive image scales, to facilitate the use of a large number of high-resolution images. For the implementation COLMAP software libraries were used. Initially, stereo pairs are identified among the unordered set of images (either aerial from the drone or unstructured coming from raw photographic images). For this purpose, all images are subsampled to a low resolution; features are extracted and a matching scheme with outlier detection (RANSAC using fundamental matrix) is applied to all possible stereo image combinations. Valid stereopairs are defined based on the number of inliers, as well as the percentage of estimated outliers after RANSAC. In case the interior orientation of the camera is unknown, an initial estimation of a common camera constant may be computed as the median of all camera constant values extracted from the fundamental matrices of all valid stereopairs (assuming that the principal point coincides with the image centre) (Sturm, 2001).

[5] https://github.com/ayoolaolafenwa/PixelLib

Once stereopairs have been selected features are extracted at a higher image scale and matched via RANSAC based on the five point algorithm (Nistér, 2004) for the estimation of the essential matrix. Image matches are thus established across different stereopairs leading to multi-image point correspondences. A bucketing algorithm is then performed to reduce the number of tie points, without affecting their distribution on images.

For the initialization of all image exterior orientations, a stereopair is selected as reference; for every new stereopair, relative orientation is estimated from the essential matrix, tie points are reconstructed in 3D space through triangulation and a 3D similarity transformation allows inserting the current stereopair into the reference system. Local bundle adjustment solutions are held for every *N* successive images to ameliorate the exterior orientation accuracy, and a full self-calibrating bundle adjustment is performed among all available images.

Following the hierarchical scheme, new feature points are collected at successively higher image scales. Matching is restricted by the known image orientations (epipolar constraint) and by a rough 3D reconstruction of the object surface that is obtained from the tie points of previous image scale. This approach is repeated up to the full image resolution, leading to final bundle adjustment.

After image orientation, dense point clouds are generated by means of dense stereo (Hirschmüller, 2005) and multi-image matching algorithms (Multiple View Stereo - MVS), followed by a triangulation in object space. These methods take advantage of the epipolar geometry derived from the exterior orientation information and determine a pixel-to-pixel correspondence between images for every image pixel, instead of distinct features only. Each pixel corresponds to a viewing ray to the object. By intersecting all viewing rays for a common, matched object point, a 3D point can be determined. By increasing the number of rays, the accuracy and reliability of the point cloud is increased. To achieve this, acquired stereo depth maps are combined with respect to their spatial resolution and their distribution in space.

In order to apply image-based 3D reconstruction techniques, the computer processes an image by analysing its mathematical features and captures all the features as a result of its understanding. Feature extraction has a great influence on the performance and success of the image-based 3D reconstruction pipeline. The **feature extraction** process refers to image feature detection and matching, which aims to identify the same features across images and then build feature tracks.

Within T3.1 of WP3, we defined, implemented and conducted a study of the state-of-the-art local descriptors, descriptor compression schemes and local binary descriptors. The feature descriptors are an important element in the 3D representation from images process to track optical features across several images or image frames. In this process, a single camera can be used or multiple cameras with a known baseline difference to achieve the required accuracy of the depth triangulation process. Feature extraction, description, and matching are being regularly improved (Miksik & Mikolajczyk, 2012). Besides new hand-crafted feature extractors and matchers, in recent years, these updates also include learnable neural networks (NN) (DeTone, Malisiewicz, & Rabinovich, 2018) (Sarlin, DeTone, Malisiewicz, & Rabinovich, 2020)

1. SIFT: Scale-Invariant Feature Transform

SIFT, proposed by David Lowe in (Low, 2004), has four main steps which are feature point detection, localization, orientation assignment, and feature descriptor generation.

2. SURF: Speeded Up Robust features

A major disadvantage of SIFT is that it is slow. SURF added a lot of features to improve the speed of the SIFT algorithm in every step. SURF is good at handling images with blurring and rotation but not good at handling viewpoint change and illumination change. SURF is better than SIFT in rotation invariant, blur, and warp transform. SIFT is better than SURF in different scale images. SURF is three times faster than SIFT because of the use of integral image and box filters.

3. AKAZE: Accelerated KAZE

The feature description performed by AKAZE is based on a Modified Local Difference Binary that uses a gradient to intensity information. This makes the descriptors robust to changes in scale. ORB is faster to compute than AKAZE and the processing time of AKAZE quickly rises with increasing image resolution. However, after filtering the matches and removing outliers, AKAZE presents a more significant number of correct matches when compared with ORB. AKAZE shows a better compromise between speed and performance than ORB for images with low resolution.

4. D2-Net

Traditional feature extractors can be replaced by a convolutional neural network (CNN), since CNN’s have a strong ability to extract complex features that express the image in much more detail, learn the task specific features and are much more efficient. D2-Net is deep learning approach based on a single convolutional neural network that is both a dense feature descriptor and a feature detector.

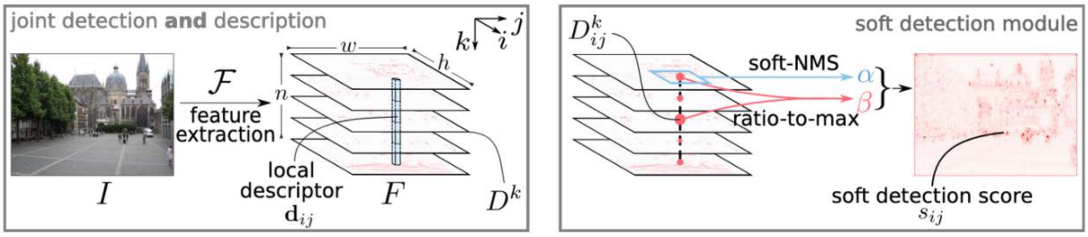


Figure 17: In D2-Net approach a feature extraction CNN is used to extract feature maps and plays a dual role, to extract both the local descriptors by traversing all the feature maps at a spatial position and the detections are obtained by performing a non-local-maximum suppression on a feature map followed by a non-maximum suppression across each descriptor

D2-Net avoids the explicit definition of interest points by training a CNN to jointly describe and detect local features. The network computes a feature map of dimension D with a resolution of ¼ of the image resolution. Interest points are defined as local maxima in the feature space. The final keypoint position is then refined comparably to SIFT and descriptors are linearly interpolated at these positions.

In our work we included the integration of D2-Net features (Dusmanu, et al., 2019) with predefined parameters including the pre-calibration of the principal distance (camera constant) into the open-source SfM-routine COLMAP.

In structured datasets captured in good lighting conditions D2-Net obtains a comparable performance with respect to SIFT. However, in indoor datasets, with bad lighting conditions or unstructured datasets, D2-Net outperforms SIFM while requiring less memory in pose estimation process.

Table 4: Evaluation on the Local Feature Evaluation Benchmark

Dataset	Method	#Images	Sparse Points	Dense points
Madrid Metropolis (datasets, n.d.)	SIFT D2-Net	500 500	116K 84K	1.82M 1.46M
Tower of London (datasets, n.d.)	SIFT D2-Net	804 785	239K 180K	3.05M 2.73M

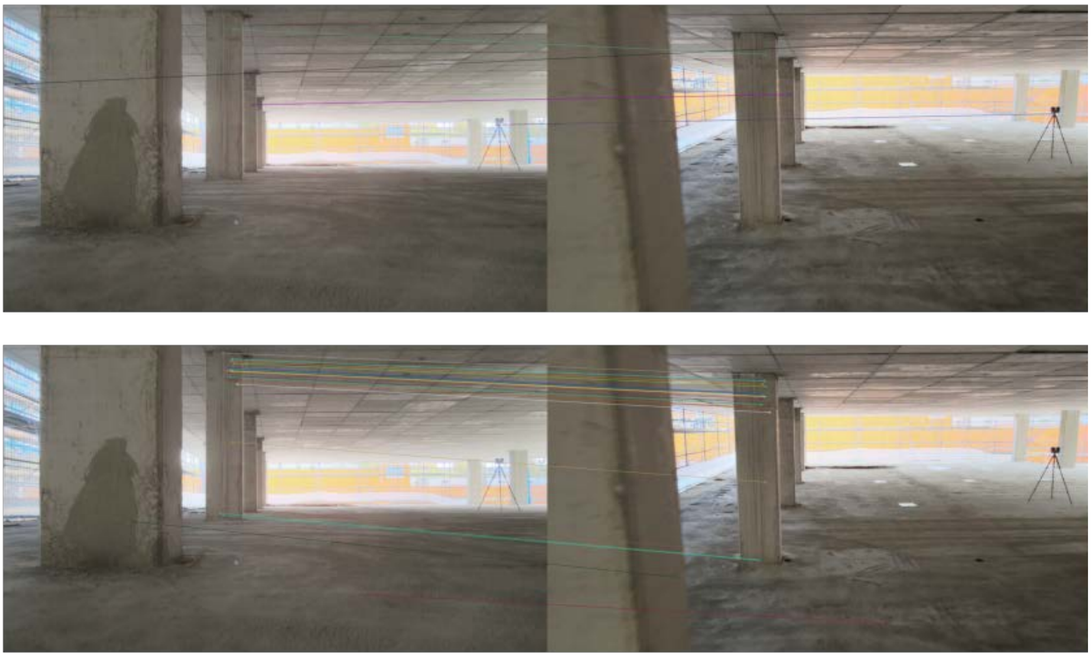


Figure 18: Example of SIFT and D2-Net feature matching in two images with poor overlap from Demo Site #6 indoor dataset a) at the top is presented the SIFT matching with 314 raw matches and 9 inliers detected for the matching, b) at the bottom image the D2-Net performs better with 1627 raw matches and 23 inliers for the matching.

The features are matched between image pairs added incrementally to the reconstructed scene. A sparse point cloud is created using bundle adjustment, following the COLMAP pipeline.

2.6.4 Feature Matching and triangulation

Feature matching stage provides a comparison of descriptors across given images. Feature matching is closely aligned to the feature extraction and involves as process the correct correspondence for as many detected features as possible. Features are matched between images and the fundamental matrix for each image is obtained through the multi-view geometry solution. The relative camera motion between a set of images will be determined with the use of corresponding features. Therefore, the matching strategy should be selected carefully according to the data collection process. For example, standard exhaustive matching approaches attempt to match every image against every other image and is appropriate to be applied to image sets that are randomly collected from a scene. Since in this approach the number of matching candidates increases quadratically with the image count, exhaustive matching is only viable with a relatively low number of images and yet these images should be carefully selected so as to include all the needed information for the final reconstruction product. In any case it is beneficial to create a feature space database and apply approximate nearest neighbour search algorithms to speed up feature matching in this space. For ordered images sets with consecutively captured images, sequential incremental approaches are used. Prior knowledge when metadata is provided, is also used, such as GPS coordinates in EXIF data.

2.6.5 Dense Matching

The result of the previous steps consists of the camera calibration and a sparse point cloud, containing the 3D reconstruction of the matched feature points. This point set is limited by design and is not a detailed or convincing representation of the filmed scene. Once a sparse representation of the scene has been completed, however, denser scene geometry may be recovered by matching as many pixels between images as possible. This process, called dense matching, is the most time-consuming part of the entire photogrammetric pipeline but can be sped-up by employing graphical processing hardware and parallel processing. Typical dense reconstruction pipelines produce depth maps from stereo pairs for all registered images. This relies on accurate exterior and interior camera parameters and epipolar geometry between images to constrain the search for matches. Depth maps are subsequently fused into a dense point cloud. It is important to note that the information contained in the depth maps is often redundant, as SfM-compliant images are usually taken with large overlap.

In sequence, fusion of the depth and normal maps of multiple images in 3D produces a dense point cloud of the scene. The method performs Multi-View Stereo (MVS) with pixelwise view selection for depth/normal estimation and fusion. As Colmap infers the best depth and normal based on both photometric and geometric consistency in multiple views, it generates `image_name.photometric.bin` and image_name.geometric.bin` under `stereo/depth_maps` corresponding to each image by default.

2.7 Single-View Depth Estimation

The previous described method focusses on understanding and formalizing the 3D to 2D projection process, with the aim to devise solutions to the ill-posed inverse problem. Quality results typically require multiple images, captured using accurately calibrated cameras. Although the SfM based techniques can achieve remarkable results, they are still limited in many aspects. For instance, they are not suitable when dealing with occlusions, featureless regions, or highly textured regions with repetitive features. The avenue of deep learning techniques, and more importantly, the increasing availability of large training data sets, have led to a new generation of methods that are able to recover the lost dimension even from a single image.

As mentioned on a previous section, the reconstruction of the 3D space from a single image is an ill posed problem since there are geometrically, infinite different 3D spaces that could have generated any specific image. The prediction of 3D information is based on state-of-the-art neural network pre-trained models, such as MegaDepth, AdaBins and MiDas combined with image filtering, 2D semantic segmentation to identify and remove the sky from images (as described in section 2.6.2) and 3D projection to convert depth predictions to 3D pointclouds. A variety of publicly available datasets for learning single image depth prediction, such as MegaDepth and KITT were exploited. The results of the service are presented in Section 2.8.1.4

Deep Learning (DL) based depth map extraction approaches refer to techniques that aim to extract 2D depth maps from input images using a data-driven learning approach. The main advantage of these methods is that they operate under a level of technical abstraction, in which heuristics are minimized and low-level technical details are left to the underlying algorithm to figure out. However, most learning-based algorithms require a significant amount of data relevant to the problem at hand to reach a satisfactory level of understanding that will lead to a robust and highly general solution.

For the depth map prediction task in particular, extracting a significant amount of tagged training data can prove to be a resource-intensive task, requiring a huge investment of time and expensive equipment (e.g. 3D scanners) as a minimum for the collection process. In ASHVIN, in order to be able to implement a personalized learning-based approach, we had to create a custom dataset containing RGB and depth image pairs. Unfortunately, we didn’t have the possibility to scan and create such datasets with depth and 2D visual information from the same point of view, therefore we were limited in using pre-trained datasets for our deployment.

Furthermore, depending on the nature of the dataset (e.g., indoor or outdoor scene dataset), the data collection process can easily lead to immensely noisy results. Due to the complex nature of the depth estimation problem, it is essential to pay attention to these data-specific limitations and handle them efficiently.

2.7.1 Datasets

To address the problem of predicting the depth map through machine learning, publicly available datasets were explored based on the needs of the project. These datasets are presented below:

MegaDepth (Li & Snavely 2018) is a large-scale dataset providing depth information for 196 locations worldwide, reconstructed using the COLMAP Sfm/MVS software (Sayab et al. 2017). The dataset contains more than 128 thousand unique frame-depth pairs and is the largest publicly available dataset for the depth estimation task. The depth maps provided in this dataset contain non-normalized values spanning over a wide variety of ranges.

KITTI dataset contains street-view (outdoor) depth estimation data pairs captured by a 3D scanner mounted on a moving car. It contains over 93 thousand depth maps along with their corresponding raw LiDaR scans and RGB images. The depth maps contain information up to 80 meters.

2.7.2 Pre-trained models

In an attempt to explore and get a deeper understanding of the state-of-art approaches we performed tests with publicly available pre-trained models. Details about these models are presented below:

MegaDepth pre-trained model follows the guidelines and research approach introduced in the dataset’s original paper. Due to the dataset's non-normalized nature, the training algorithm is designed very carefully and huge attention is paid to the loss function. In particular, a composite loss function is developed that contains various components aiming to preserve object ordinality, smooth depth transitions and sky separation from the scene objects.

The **AdaBins** (Bhat et al. 2021) pre-trained model focuses on object continuity and aims to extract gradual depth values for each object by first calculating the depth value of each object's centre and then interpolating all other values using the calculated centre. The algorithm uses a learnable bucketing system in which depth buckets are extracted dynamically for each input image, clustering and guiding the depth distribution around the bucket centres. Pre-trained checkpoints exist for both NYU and KITTI datasets.

The **MiDas** (Rantfl et al. 2019) pre-trained model aims to achieve high generalization by using ten different datasets as its learning base. The algorithm focuses on utilizing learning objectives that are invariant to specific dataset peculiarities. Thus, the training procedure can process various training data and learn meaningful information from them.

The final deployed method is based on MegaDepth which presented better results for images of outdoor spaces.

2.8 3D point cloud demonstration capabilities

This subsection presents the final results of the 3DRI service in respect to the data collected from the demo sites until M23. The processing pipeline of visual material consists of the following steps:

- Shot detection
- Keyframe extraction
- Sparse alignment
- Dense alignment

2.8.1.1 Demo Site 1

During the course of ASHVIN, 3 corresponding sets of videos from drone recordings were collected for demonstration site #1, for the 1) Underpass bridge, 2) Valdelinares and 3) PlataBridge. The purpose, as discussed in Section 2.3.1, was to record and digitize the infrastructure of the bridges. This is achieved by taking highly overlapping images and videos that are then processed by means of photogrammetry and 3D computer vision algorithms. Images need to be clean, in focus and sharp. Typical overlap percentage in such applications is around 80% in both directions (sometimes even higher percentage up to 90% may be applied) to ensure that all object points are depicted in multiple photos. This leads to better triangulation accuracies in 3D reconstruction. At the same time occlusions and hidden areas should be avoided. Usually, to perform such recordings an autopilot software is used that allows the planning of the drone mission by setting the basic parameters such as the area that needs to be captured, the flight elevation, the camera direction, the overlap percentage, the take-off and landing points. However, the flights in this use case were performed manually and the capturing is arbitrary due to the operator’s training exercise. In the following images are presented the results for the underpass bridge.

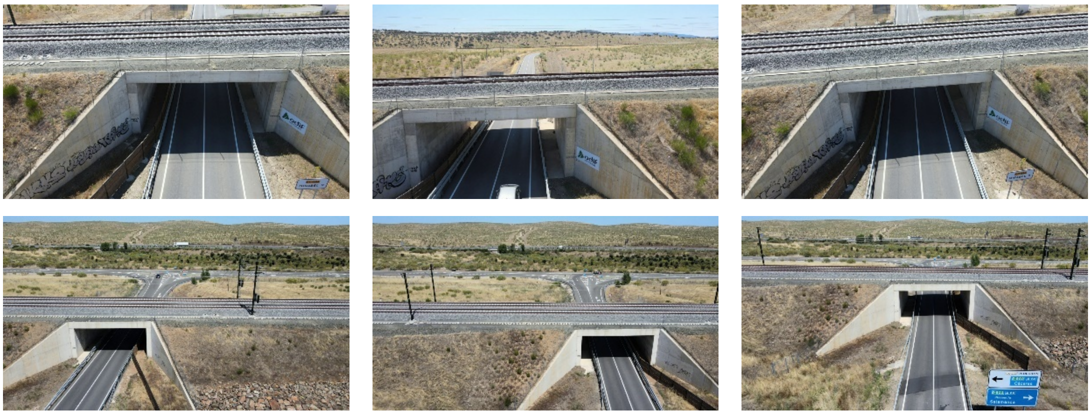


Figure 19: Sample of images extracted by the recording reffered to the Underpass Bridge in Demo Site #1.

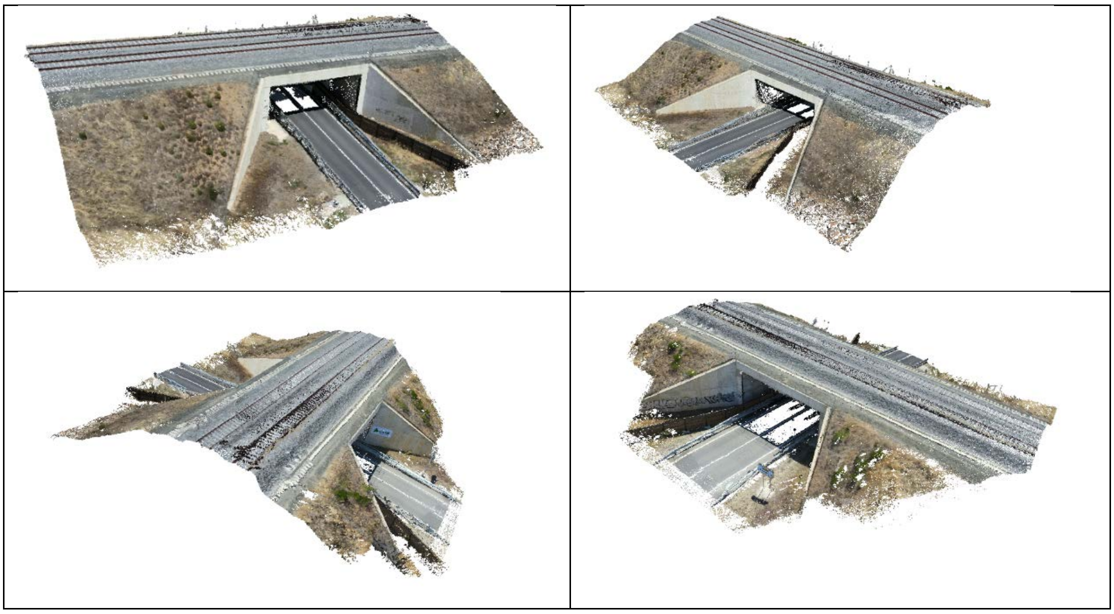


Figure 20: Results of the image-based 3D representation performed by 3DRI method.

2.8.1.2 Demo Site 6

The dataset collected consists of two sets of videos obtained from the air mavic drone, namely the sets DJI_067 and DJI_0270. Two more videos were captured but with main focus on the dissemination activities, therefore the coverage is irregular and mainly targets the people participants and not to the construction structure. The captured videos were followed. The SRT file contains the flight trajectory metadata information including gps location, size (in pixels), aperture value, exposure speed, ISO value etc., that are used for the pairing of matches and the extraction of the camera intrinsic matrix.

DJI_067 is a 1.01-minute drone flight at 25fps recorded indoors on the 2nd floor of site #6 by the UPC team who is the partner responsible for the managing of the Demo Site. 149 frames were extracted from this video. Although it is provided an SRT file with the metadata, the recorded GPS positions are empty since the gps signal didn't localise in the indoor environment.

DJI_0270 is 3.14 minute in length captured again at 25fps from the ourdoors of site #6. 299 frames were extracted from it.

In both cases, the overlap is inherently high. No flightpath is followed for the data capturing since the flight performed manually in both cases.

The results of the reconstruction are presented in the figures below.

Outdoor

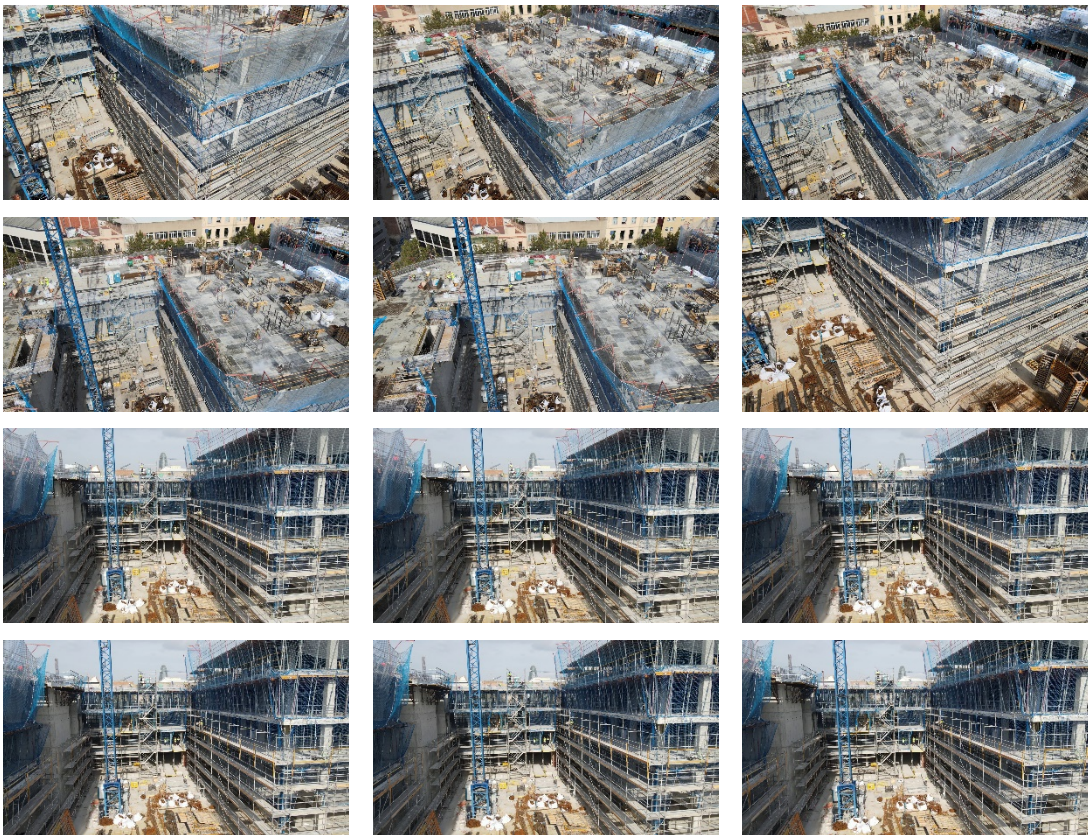


Figure 21 Sample of images extracted by the outdoor drone recording at Demo Site #6.

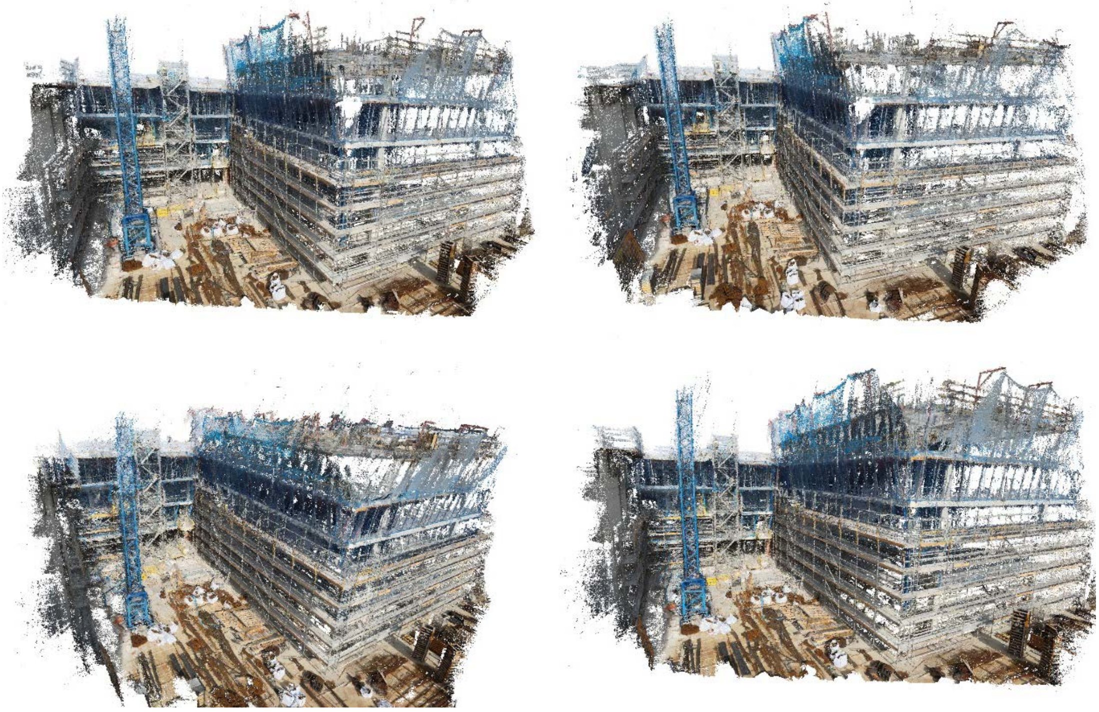


Figure 22 Resulting pointcloud from the image-based 3D representation technique performed using the 3DRI method for the outdoor dataset at Demo Site #6.

Indoor

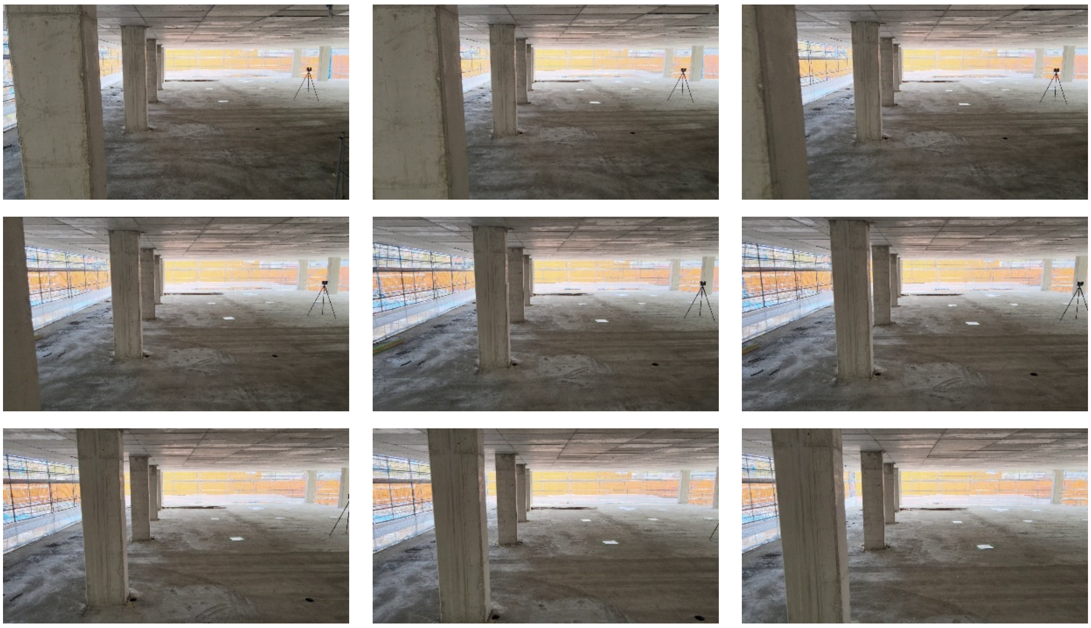


Figure 23 Sample of sequential images extracted from indoor drone recording at demonstration site #6.

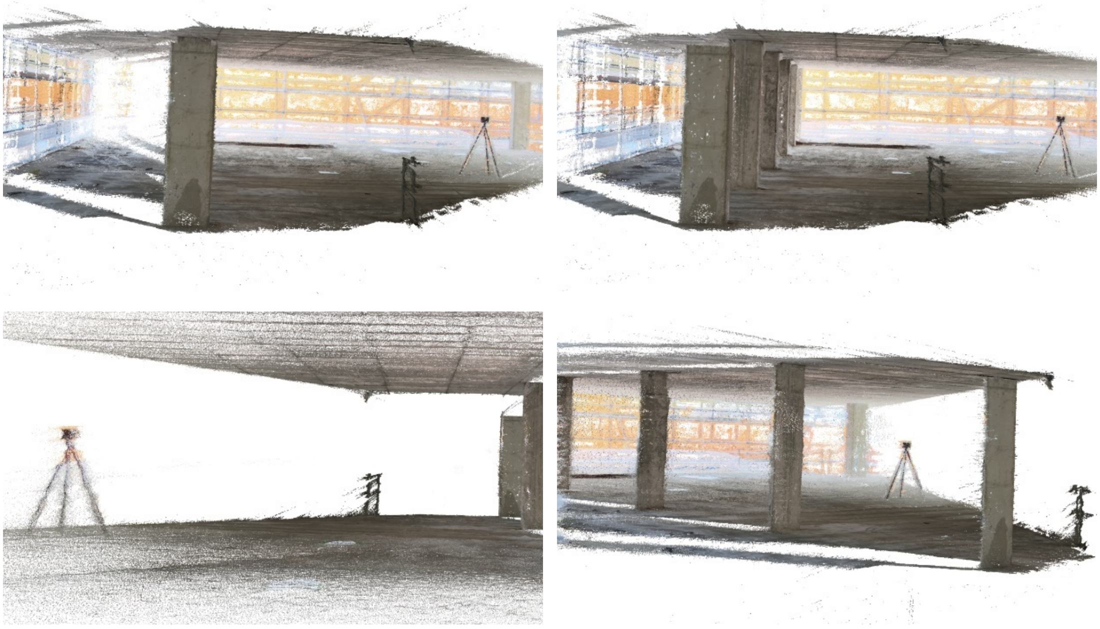


Figure 24: Point cloud resulting from the image-based 3D imaging technique performed with the 3DRI method for the indoor dataset at Demo Site #6.

2.8.1.3 Demo Site 7

For demo site #7 only one single shot video was recorded until the reporting period that served as a sample to identify the limitation of the method and schedule more carefully the collection procedure. In the specific shot the camera rotates away from the pile and performs a panorama motion, unsuitable for 3D reconstruction. Moreover, scene exhibits symmetries and duplicated structures. To overcome the issue, we performed a filtering process by reimplementing the ideas from Distinguishing the indistinguishable: Exploring structural ambiguities via geodesic context by (Yan, Yang, Zhang, & Xiao, 2017). (CVPR 2017).

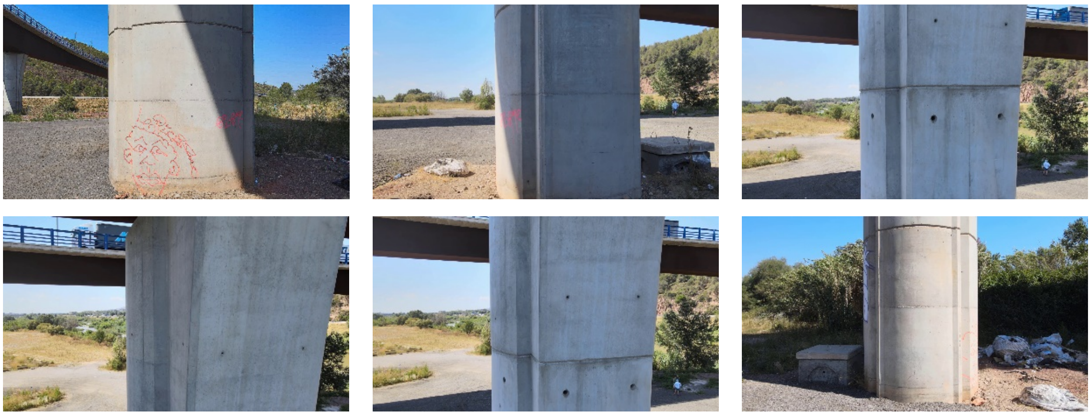


Figure 25: Sample of sequential images from drone recording at demonstration site #7.

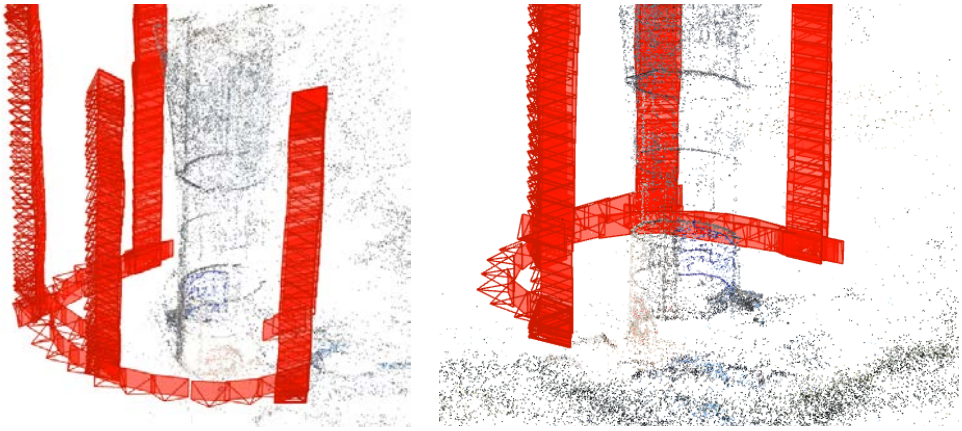


Figure 26: Unfiltered sparce point cloud and camera localisation around the bridge pile.

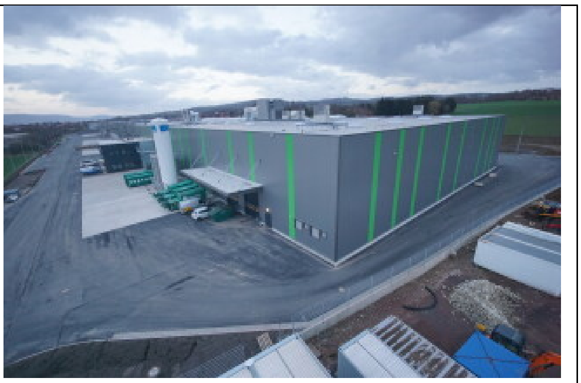
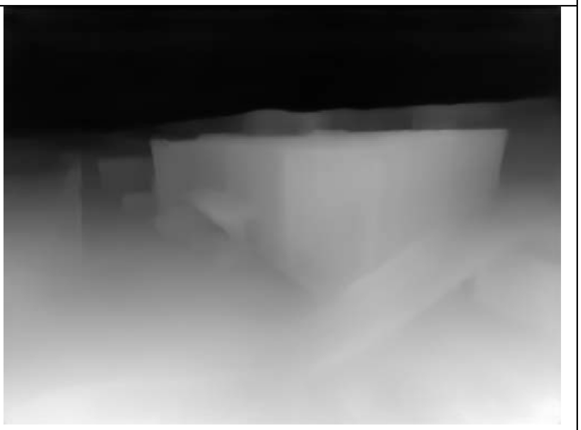
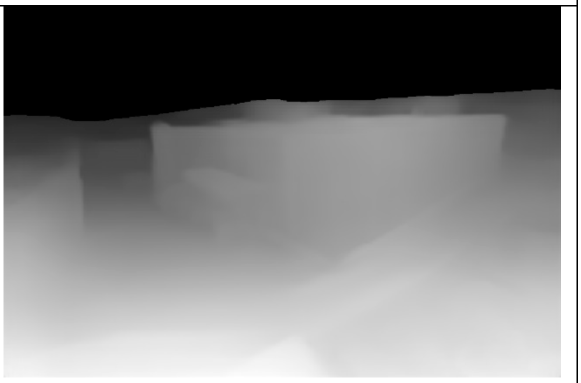
The data collection process for this demonstration site will take place over the next few months, where based on the findings and limitations of the method developed, new instructions were communicated to the demonstration site manager to perform the data collection.

2.8.1.4 Demo Site 4

The data obtained from Demo Site #4 consisted of a set of time lapse images taken from a fixed camera at a defined position. Since the images were taken using a camera from a fixed position, it caused occlusion of certain regions of the site.

For this set of images, the single image 3D depth prediction approach was applied to predict the depth maps from single images. The original size of the images acquired was 6000×4000 pixels, the images were reduced close to the training resolution.

Semantic sky segmentation was used to remove the inconsistencies in predicted sky regions, boosting the edges of the appearing objects. As detailed in section 2.6.2, the sky segmentation detects the sky pixels on the input image and nullifies their predicted depth values, minimizing unnecessary noise and prediction. An example of the sky segmentation post-processing technique, applied in the MegaDepth pre-trained model, are presented below:

original image	
without sky segmentation filtering	
with sky segmentation filtering	

Sky segmentation masking proved to aid the most in sharpening the edges of the depicted objects. Moreover, the sky segmentation technique is a robust method to crop out the sky pixels when projecting the image into a 3D format. Thus, the resulting 3D objects appear more realistic without cluttered sky information.

After extracting the estimated depth maps, the next step in the 3D estimation pipeline is to project the input image in three dimensions. Various mathematical techniques exist to solve the projection problem, all of which revolve around generating three-dimensional points by grouping input image coordinates with the corresponding values of the depth map.

A variant of the LeRes 3D projection algorithm was used to extract three-dimensional points from the input image and project them into a coloured point cloud. Point cloud format was chosen against other three-dimensional structures due to its simple representation, which aids in reducing visual noise generated by view occlusion.

Examples of the resulting 3D projection are depicted below:

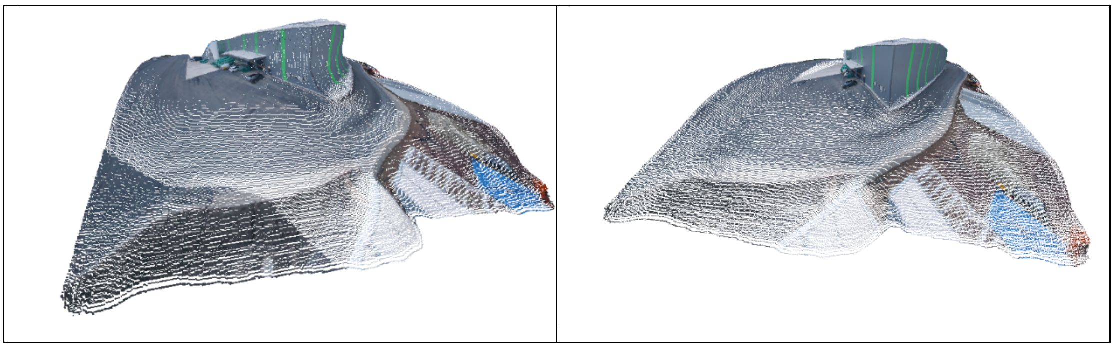


Figure 27: Examples of the 3D projection algorithm for single view depth estimation for Demo site #4.

2.9 Additional Work

Image matching and the data association are still open research areas in the fields of computer vision and robotic vision respectively. The detector and the descriptor chosen directly affect the performance of the system to track the salient features, recognize areas previously seen, build a consistent model of the environment, and work in real time.

To date, there are no standards for evaluating and comparing the general efficiency and effectiveness of a complete visual 3D representation systems based on SfM or VSLAM. Nonetheless, there are several indicators that may characterize their performance, such as the degree of human intervention, accuracy of location, reconstruction consistency, and the control of computational cost that arises among others.

As future work within ASHVIN there are still some steps to be taken with a main focus on the application of 3D reconstruction techniques for the new collection to be carried out for demonstration site #7 and the integration of the method into the ASHVIN platform. We are also targeting to explore the potential of using neural radiance fields that recently proposed by (Mildenhall, et al., 2021) for accurate depth estimation. Therefore, our future work includes optimization for reconstruction based on the improved geometric structure in the learned neural radiance fields.

Finally, further effort will be devoted for the integration of the 3RDI method into the digital twin platform and link it to the 4DV-C tool. The results will be reported in D4.6 “Visualizing and dashboard construction activities based on digital twin data” deliverable due on M30.

3 Defect Detection

Demonstration site #3 refers to Zadar airport, one of nine airports located in Republic Croatia. Although extending and reconstructing the airport is planned, due to COVID crisis the expansion is currently postponed. Thus, in the context of ASHVIN the demo project objective was changed from construction monitoring to maintenance of the existing operational areas. Through the discussions with end-users arose the requirement for a tool to monitor the condition and support the maintenance of the airport's runway. CERTH, committed to the consortium, supported this task by deploying a framework for quality control via defect detection on visual data.

Quality control is a crucial aspect in the construction industry. Depending on the method employed to identify a defect on a structure a surface or a component, quality control strategies can be classified as destructive or non-destructive. Non-destructive testing methods (NDT) are intended to monitor and evaluate the integrity of a component or structure to detect defects without extracting samples from it, destroying it or removing ts suitability for service (Czimmermann, et al., 2020).

Among them, the visual-based approach for defect detection is one of the most common procedures. Currently, with the aforementioned advances in the field of computer vision, many researchers have dedicated their efforts to develop image-based automatic NDT methods for contactless or even remote defect detection systems. Furthermore, in the last decade, numerous research and practice efforts have been made to implement computer-vision approaches combined with UAV technology to monitor and inspect infrastructure (Bukhsh, Anžlin, & Stipanović, 2021), (Žnidarič, Kreslin, Anžlin, & Krivic, 2020). Drone-enabled inspections coupled with computer vision technology has the potential to serve as a more economical and safer alternative to conventional inspection and monitoring practices.

3.1 Related Technologies

Research community has been significantly active in the task of automating the defect detection process. Computer vision researchers have also been enabled in this field aiming to provide image-based solutions. The main objective in this case is processing the acquired visual data and identifying the depicted surface crack instances. Visual recognition of surface cracks is a quite challenging task due to their irregular shape and size, as well as, their essential similarity to the background texture. Furthermore, the background texture can vary significantly case-to-case, adding further limitations into the effort of deriving a solution that can be generalized in most structures and construction materials.

The majority of the initial approaches were based on mature image processing techniques in order to detect the depicted crack and discriminate it from the background. Tree structures (Zou, Cao, Li, Mao, & Wang, 2012) and genetic algorithms (Nishikawa, Yoshida, Sugiyama, & Fujino, 2012) have been utilized towards this direction. A significant number of presented methods have been focused on image filtering, aiming to enhance the distinction of the crack instance from the background. Salman et al. (Salman, Mathavan, Kamal, & Rahman, 2013) proposed a method utilizing Gabon filters to detect cracks on varying-texture images of pavements. Authors in (Fujita & Hamamoto, 2011) proposed a method combining image filtering to reduce noise and enhance crack-related features, with a probabilistic model to detect the depicted cracks in the processed image. In (Yeum & Dyke, 2015) an edge detection-oriented method was presented, aiming to identify crack-like edges through Hessian matrix–based filtering of the image.

The advent of machine learning and Convolutional Neural Networks (CNNs) was also reflected in the research field of crack detection. Deep learning architecture, named GoogleNet, was utilized in (Ni, Zhang, & Chen, 2019) to classify surface crack in high-resolution images. Similarly, authors in (Zhang, Nateghinia, Miranda-Moreno, & Sun, 2022) exploited a custom-built dataset to train a CNN model capable to detect cracks on pavement images. CrackNet was proposed in (Zhang, Yang, Zhang, & Zhu, 2016), a CNN network capable to preserve the spatial dimensions of the input image in order to detect in pixel-level the depicted cracks. Liu et al. (Liu, Yao, Lu, Xie, & Li, 2019) presented DeepCrack, a deep leaning architecture based on SegNet (Badrinarayanan, Kendall, & Cipolla, 2017), which semantically segments the input image to crack and background. Similar approaches based on semantic segmentation were also followed in (Yang, et al., 2018), (Bang, Park, Kim, & Kim, 2019) in order to detect crack instances in road and pavement surfaces.

A set of works has been presented focused on detection techniques utilizing UAV-based visual data. Authors in (Lei, Wang, Xu, & Song, 2018) deployed a crack detection method for bridge inspection via UAVs. The approach was based on a sophisticated image processing framework in order to enhance the contrast among crack and background, leading to efficient detections. A decision-making tool for UAV building inspection was presented in (Kucuksubasi & Sorguc, 2018), where CNNs were fine-tuned to the task of crack detection. Choi et al. (Choi, Bell, Kim, & Kim, 2021) developed a CNN-based framework that processes images acquired via UAV and classifies them as crack or not, while by employing other sensing modalities it provided information regarding the location of the detected cracks.

Despite the interesting results reported in the literature, the majority of the presented methods is focused only on a specific type of defect, such as cracks. Furthermore, the evaluation process is usually based on simplified cases, where the crack instances are captured from a close distance with a homogenous background. Realistic inspection scenarios imply the existence of different types of defects, usually mixed in varying and changing background, under different illumination conditions and textures. Thus, providing a robust method capable to meet these challenging requirements remains an active field of research.

3.2 Relation to User Requirements and ASHVIN demonstrations

Under the scope of ASHVIN platform, a module is required enabling the condition monitoring of airport infrastructure and specifically runway. According to D7.1: “Ashvin technology demonstration plan” the required method, titled Defect Detection using Computer Vision (DDCV), should perform the following task:

Towards this direction, Zadar airport is selected as the demo site #3 to implement its digital twin. Zadar runway is exposed to high load and due to the defined high safety standards, condition monitoring of runway and other operational areas is required. Runway surface defects such as cracks or accumulated tyre marks are considered as the main deterioration threats, since they may critically affect the surface friction and thus, comprise a significant safety threat.

Table 5: Plan for DDCV method as described in D7.1.

ASHVIN tool/method	Name	How the method will be used on the project
DDCV (method)	Defect Detection using Computer Vision	The AI-powered solution is used to detect damages, anomalies and objects on the runway surface and green areas around the runway. The aim is to integrate the automated damage detection into inspection and maintenance planning process, as part of RISA tool.

The main vision in this case is to employ Unmanned Aerial Vehicles (UAVs) in order to automate and improve the inspection and monitoring process, by collecting, storing and analysing structural features (i.e. cracks, joints, tyre marks) in an objective, repetitive and efficient way, which is currently conducted in non-digital way. Furthermore, a machine-based inspection can enable the ability to detect defects on their early stage, which cannot be possible through visual inspection conducted by humans. Thus, the employment of the specific module for automatic inspection will decrease the overall required time and workload, while improving its efficiency and objectivity. Moreover, the collected UAV data can be assessed by the corresponding GIS model to improve the overall inspection and maintenance decision making process.

Based on the aforementioned user requirements and defined KPIs for maintenance, an AI-based defect detection module has been deployed for the ASHVIN platform. The main objective in this case is to process UAV images acquired from airport runway and provide crucial information regarding the runway's condition and the detected defect types.

3.3 Methodology

In this section is described the designed and deployed solution, named DDCV according to D7.1. In specific, an AI-powered approach has been developed focused on image semantic segmentation. The deployed module processes the visual input, acquired via UAV camera, and produces an annotated mask where the detected defects are segmented accordingly. The core element of the specific framework is a deep-learning model based on UNet architecture, which semantically segments the RGB input images. The model was trained and evaluated on data collected from Zadar airport demo site #3, where different types of defects are depicted.

In Figure 29 the overall pipeline of the developed DDCV module is presented. The deployed CNN model, trained and evaluated on the collected data from the Zadar airport, is enclosed in the overall framework, capable to provide a set of valuable results in order to enhance the inspection process. In specific, apart from detecting defect instances, the collected visual data can be exploited to create the orthomosaic representation of the scanned runway, which will be used for development of the digital model of the infrastructure. Furthermore, information regarding the location, the number and the type of detected defects can be extracted and combined in order to estimate the severity level of the inspected damages, which enables objective and digitally tracked evolution of damages over time. This will provide the information for the development of performance prediction model and optimization of the maintenance planning, implemented in RISA model. Overall, the designed framework enables the automation of the inspection process and operates as a decision support system to schedule targeted maintenance operations, which will reduce costs and environmental impacts for airports.

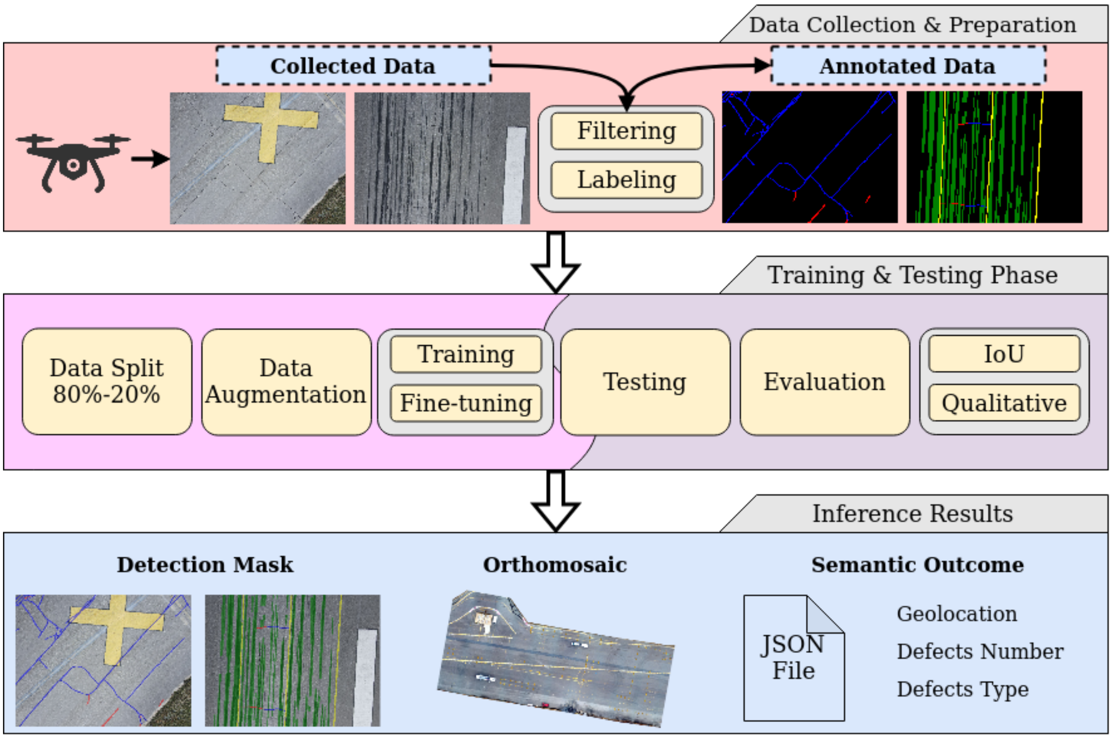


Figure 28 Operational pipeline of the deployed DDCV module

3.3.1 Custom-built Dataset

The developed dataset contains RGB images collected through two different phases. In the first phase, hand-held cameras were utilized to capture a wide variety of defect instances observed on the surface on the surface of the airport's runway. The collected data were annotated accordingly, using Make Sense tool, by the partners and field-experts of Infraplan, leading to a set of 301 images. The specific set contains 4 different classes, namely crack, joint (repaired cracks), tire marks and background. In the second phase, a camera mounted on a UAV was exploited to collect visual data from a wider area of the runway. Following a similar approach, the collected images were annotated accordingly for the semantic segmentation task. Based on some initial results derived the necessity to discriminate joints of repaired cracks from the construction joints. Thus, an extra class was added containing the construction joints leading to a set of 5 classes. In total, 386 high resolution images were collected yet, since labelling is a time-consuming process that requires a heavy amount of workforce, a smaller part of 100 images is currently annotated. Nevertheless, the images collected through the two aforementioned phases were joint into a pluralistic dataset containing 401 images enclosing a wide range of defect instances under varying capturing conditions. The deployed dataset was utilized to train and test the designed model. In specific, 20% of the drone images was utilized for testing while the rest 80%, combined with the data of the first phase, were employed for training.

3.3.2 Detection Model

The developed AI model is based on UNet (Ronneberger, Fischer, & Brox, 2015) architecture, a well-known network for image semantic segmentation. The specific architecture is based on two main components, the encoder and the decoder. The aim of the encoder is to enclose the input image in a compact representation yet containing high-level information. To achieve that, a set of consecutive convolutional layers is utilized to gradually decrease the spatial dimensions of the input while increasing its number of channels (depth) in order to increase the information content of the encoded vector. The objective of decoder is to extract the segmented outcome from the encoded representation. Towards this direction, the encoded vector is upsampled by combining interpolation with convolutional layers, in order to meet the original spatial dimensions while its depth is gradually decreased. The aforementioned operation leads to the segmented outcome, where each one of the image pixels is assigned to one of the defined classes. Aiming to increase the model efficiency, skip connections are employed among the layers of the encoder and the decoder. This design secures seamless backpropagation of the information to the initial layers of the network, while feeding the latest layers with crucial low-level features.

For the encoding stage it was utilized EfficientNet (Tan & Le, 2019), a state-of-the-art architecture, pretrained on ImageNet (Deng, et al., 2009), aiming to extract robust high-level features. In the decoding stage, attention blocks were added before the upsampling layers in order to allow to the model to focus on parts of the images with high semantic content and thus, enhance its efficiency. The model was trained for 2000 epochs with batch size 16. A set of image processing techniques was employed for data augmentation. In specific, input images are randomly (with probability 50%) flipped horizontal and vertical, rescaled and modified in terms of brightness. Finally, a patch of 256 X 256 pixels is randomly cropped from the processed image. Training was conducted with Adam solver and learning rate equal to 10^{-3}.

3.4 Evaluation

In order to validate the efficiency of the deployed defect detection method, the segmentation accuracy is measures in terms of Intersection-over-Union (IoU). IoU is a well-known, sophisticated, metric extensively employed in evaluating semantic segmentation methods. Since the original resolution of testing images is quite high, tiles of 512 X 512 pixels are fed to the trained model and recomposed to the segmented outcome. IoU is measured for the whole testing dataset of the segmented images and reported for each class in Table 6. The mean IoU (mIoU) averaged over the 5 classes is also reported.

Table 6: Evaluation results for each of the classes

Crack	Joint	Construction Joint	Tyre Marks	Background	mIoU
18.72%	35.00%	49.80%	68.22%	96.68%	

Although the developed model can effectively detect the majority of classes, results imply the challenging nature of the problem. Regarding classes crack and joints the developed can detect the depicted instances yet, it presents lower accuracy into identifying in pixel-level their exact shape and size. This remains a challenging task, due to their complex shape and the ambiguity of their outlines of these classes. Regarding the rest classes, the model reports higher performance due to the higher number of samples, especially in case of tyre marks and background. Especially in case of construction joints, the model can effectively distinguish them from joint instances, although these two classes differ only in shape (construction joints are usually straight long lines), while texture characteristics are the same for both of them. In total, it should be noted that the developed model can provide adequate information regarding the detected defects, since its main objective is fullfield by to identify the main “skeleton” of the captured defects. The above analysis is also reflected in the qualitative results provided in the following section.

3.5 Defect Detection examples

A set of qualitative results extracted via the deployed method is presented in Figure 27.

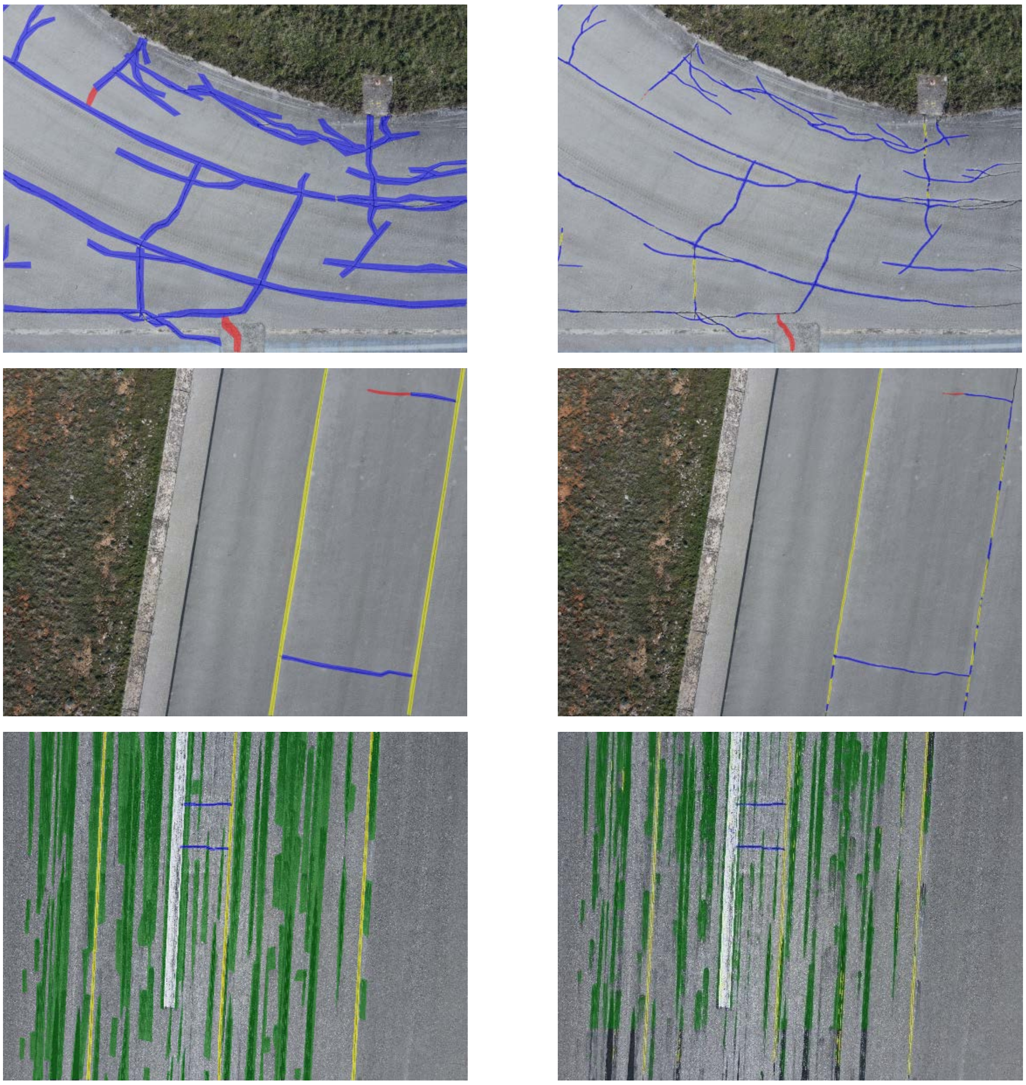


Figure 27 : Results of the deployed DDCV method. Cracks are highlighted with red, joints with blue, construction joints with yellow and tyre marks with green color, respectively.

Results imply that in most cases the deployed model can efficiently detect the captured defects. Segmentation is more accurate for classes such tyre marks and construction joints, since there are plenty training samples due to their natural shape and size.

Regarding joints class, the model can efficiently detect relevant instances, especially in cases where it clearly distinct from the background. Yet there are cases where construction joints are mislabelled as joints and vice versa. Note that the texture of these two classes is identical, yet they differ in terms of shape and size. Thus, it leads to a challenging issue that cannot be fully tackled by the developed model. Considering cracks class, although the depicted instances are adequately recognized, yet there is some room for improvement.

3.6 Disscusion

In the context of the specific task, a robust AI-based method, named DCCV, for defect detection was employed. Its core element is an efficient CNN model, trained and evaluated on real-world data collected from the demonstration site #3 of Zadar airpot. Evaluation results implied that the deployed module is capable to detect 4 different type of defects adequately, yet highlighted the challenging nature of the defect detection task. More specifically, although the deployed method can detect the depicted crack instances, their shape cannot be perceived in its full complexity. The efficiency of the deployed model is expected to be increased significantly by employing higher number of training data. Towards in this direction, the remaining collected data will be annotated accordingly to create an extended dataset, covering more defect cases. Furthermore, possessing a dataset with higher number of samples, will enable the ability to train deeper architectures which can lead to performance improvements.

4 Construction Site Object Recognition

4.1 Introduction:

Site progress monitoring is a vital aspect for the successful completion of a construction project since it offers critical information for management to make timely and informed decisions (Son, 2010). Conventional construction monitoring practices are manual, time-consuming, and prone to human error. Inadequate dynamic progress monitoring results in losing control of the project's success, resulting in time and cost overruns (Varun Kumar Reja, 2022). Therefore, automation of the progress can highly improve the efficacy of the process, making it easier and error-free.

Several studies have pointed out that a systematic and regular inspection of the sites can help identify the deficiencies at the site in an early stage to prevent any impending losses. This (McCulloch) study also shows that construction managers, on average, spend 30 to 50 percent of their working hours recording and analyzing the data manually. Eventually, this leads to distraction from other important tasks.

The construction industry faces many challenges, including lower productivity, safety, and cost overruns. However, as a result of the fast advancements in computer vision technology, it is now feasible to dynamically monitor tasks at construction sites that cannot be accomplished by a human vision system thereby effectively enhancing the safety management, productivity tracking, quality, and cost control of the projects (Aritra Pal) (Suman Paneru, 2021). The recent development in the field of deep learning object detection algorithms has enabled the application of computer vision technologies to different use cases in construction to be more effective and expedient in terms of speed, accuracy, and feature extraction (Zhang Y. , 2021) (Shrey Srivastava, 2021). A critical aspect of applying these DL object detection algorithms is the availability of sufficient image data for training the algorithms.

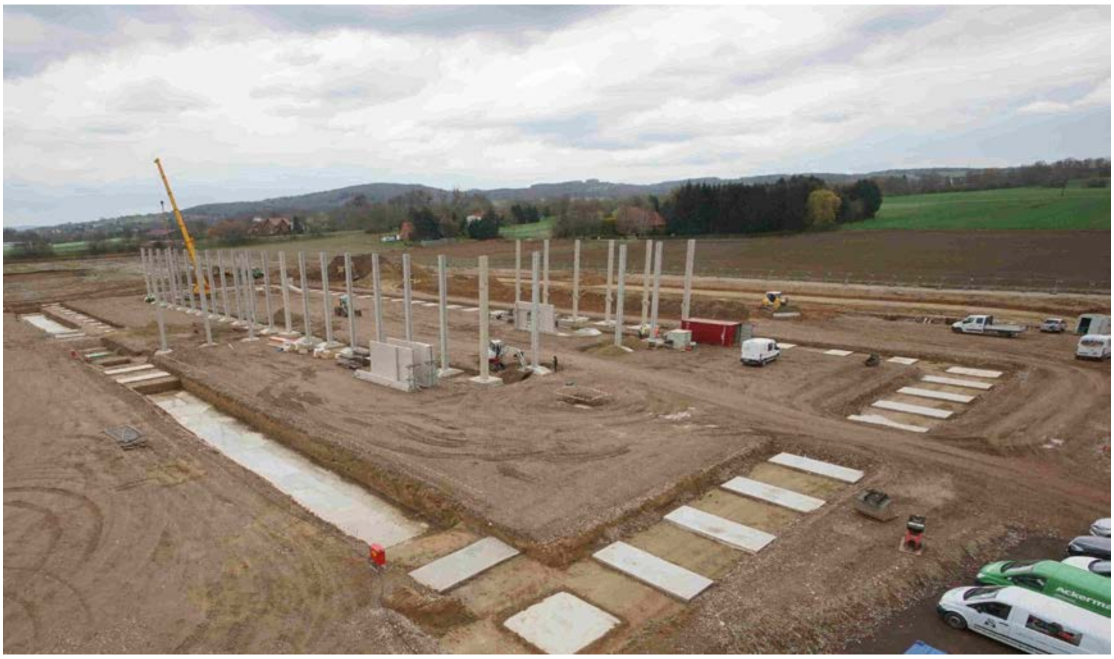


Figure 29 Rinteln demo site

Images are usually collected in different ways. The camera could either be monocular or stereo. In this subtask, the main objective is to monitor the site activities at the Rinteln demo site and asses the productivity rates. The first step towards achieving this objective is the recognition of objects at the site. The demo site at Rinteln adopted a modular construction methodology with precast structural elements largely being used for construction. To achieve the objective, the image data set was collected from the Rinteln site which consists of the images of the site taken by a camera at a fixed position. Figure 30 shows one of the raw images taken using the camera.

Around 300 images were handpicked from the dataset, pre-processed, and annotated. The dataset was then tested on two object detection algorithms. Moreover, the study also aims to suggest researchers select appropriate algorithms for various applications in the field of construction site monitoring.

4.2 Related Technologies:

4.2.1 Computer vision in construction site monitoring:

There have been several studies and demonstrations on applying deep learning-based computer vision techniques in the construction industry, specifically in construction safety or risk management, production control, and personnel management (W. Fang).In his study, Kong et al. (Ting Kong, 2021) combined computer vision with LSTM to predict unsafe behaviour on construction sites. In their research, Nath et al. (Nipun D. Nath, 2020) also presented three DL models built on YOLO architecture to detect the PPE compliance of workers. Braun et al. (Alex Braun, 2020) proposed a DL-based object detection approach that supports construction progress monitoring by verifying element categories compared to expected data from the BIM model. Also, Fang et al. (Weili Fang, 2019) in his study developed a Mask R-CNN model that can detect people that traverse concrete/ steel supports during construction which can be used for identifying unsafe behaviour in construction sites. Furthermore, Koch et al. (Christian Koch, 2015) presented a review study on state-of-the-art computer vision-based defect detection and condition assessment related to concrete and asphalt civil infrastructure. Beckman et al. (G.H. Beckman, 2019) also proposed a method that can be used for automatic volumetric quantification of concrete spalling employing a depth camera and faster region-based CNN.

There have been several studies and demonstrations on applying deep learning-based computer vision techniques in the construction industry, specifically in construction safety or risk management, production control, and personnel management (W. Fang).In his study, Kong et al. (Ting Kong, 2021) combined computer vision with LSTM to predict unsafe behaviour on construction sites. In their research, Nath et al. (Nipun D. Nath, 2020) also presented three DL models built on YOLO architecture to detect the PPE compliance of workers. Braun et al. (Alex Braun, 2020) proposed a DL-based object detection approach that supports construction progress monitoring by verifying element categories compared to expected data from the BIM model. Also, Fang et al. (Weili Fang, 2019) in his study developed a Mask R-CNN model that can detect people that traverse concrete/ steel supports during construction which can be used for identifying unsafe behaviour in construction sites. Furthermore, Koch et al. (Christian Koch, 2015) presented a review study on state-of-the-art computer vision-based defect detection and condition assessment related to concrete and asphalt civil infrastructure. Beckman et al. (G.H. Beckman, 2019) also proposed a method that can be used for automatic volumetric quantification of concrete spalling employing a depth camera and faster region-based CNN.

In the field of activity recognition, Luo et al. (Xiaochun Luo & Dongping Cao) he proposed a two-step method to recognize diverse construction activities from the site images in a fully automatic way. Also, Pan et al. developed a DL-based -prediction model to estimate tunnel boring machine performance during deep excavation operations under complex underground environments. From the literature review, it has been found that several studies have successfully applied DL-based object detection algorithms to enhance construction safety, productivity and to, asses the workforce, and monitor progress at sites. Also, this emphasizes the need for developing and enriching a data set focuses more on construction site objects.

4.2.2 Deep learning-based object detection algorithms:

Object detection entails both recognition (e.g., "object categorization") and localization (e.g., "location regression") tasks. An object detector must accurately localize each object instance and correctly predict the category label for each object to identify objects of certain target classes from backgrounds in the picture. These target object instances are intended to be localized using bounding boxes or pixel masks (Xiongwei Wua) .

The state-of-the-art object detectors based on deep learning are divided into two main categories two stage detection algorithms and one stage detection algorithms. The detection task is divided into two steps by two-stage detectors: (i) generating proposals, and (ii) creating predictions for these proposals. The detector will look for regions in the picture during the proposal generation stage that might possibly represent object regions. The purpose is to propose regions with a high recall, such that at least one of these regions comprises all of the objects in the image. In the second stage, these proposals are classified with the right category labels using a deep-learning-based model (Xiongwei Wua). Either a background or an object from one of the predefined class labels could form up this region. However, the proposal generation stage is not a separate phase in one-stage detectors. Usually, they treat each region on the picture as any potential object and categorize each area of interest as either the backdrop or the intended target (Xiongwei Wua).

R-CNN is the first implemented two stage object detectors. SPP-net, Fast R-CNN, Faster R-CNN, and Mask R-CNN are some most commonly used improvised R-CNN-based algorithms. A typical and most commonly used one-stage detector is YOLO owing to its high accuracy and the ability to run real-time (Xiongwei Wua). In recent years, several versions of YOLO have been released namely YOLO V2, YOLO V3, YOLO v4, and YOLO V5 including some revised limited versions (Peiyuan Jiang, 2021).

4.3 Relation to User Requirements and ASHVIN demonstration:

The object detection algorithm implemented in this task would enable object detection at the Rinteln demo site. Recognition of objects at the site is the first step towards implementing a tool for activity recognition at the site which in turn can help calculate the productivity of any construction project; which is one of the KPIs of ASHVIN.

This tool can be further extended to predict the time required for each activity at the site for example in the case of Rinteln demo site, it would help predict the time taken for mounting precast columns at the site, the time taken for unloading columns from the transportation trucks etc. Object detection lays the foundation stone to achieve this goal. Furthermore, as the demo case at Rinteln had adopted a modular construction method, measurement of productivity rates and cost is of utmost importance in assessing the project success. Moreover, the enrichment of the dataset made to support the object detection also makes it possible to use it in future tasks that requires labelled data from the sites.

4.4 Methodology:

The following Figure 31 shows the methodology adopted in this study. The study consists of four main stages namely data preparation, pre-processing the image data, selection of appropriate algorithms for object detection, training, testing and comparison between the two implemented models.

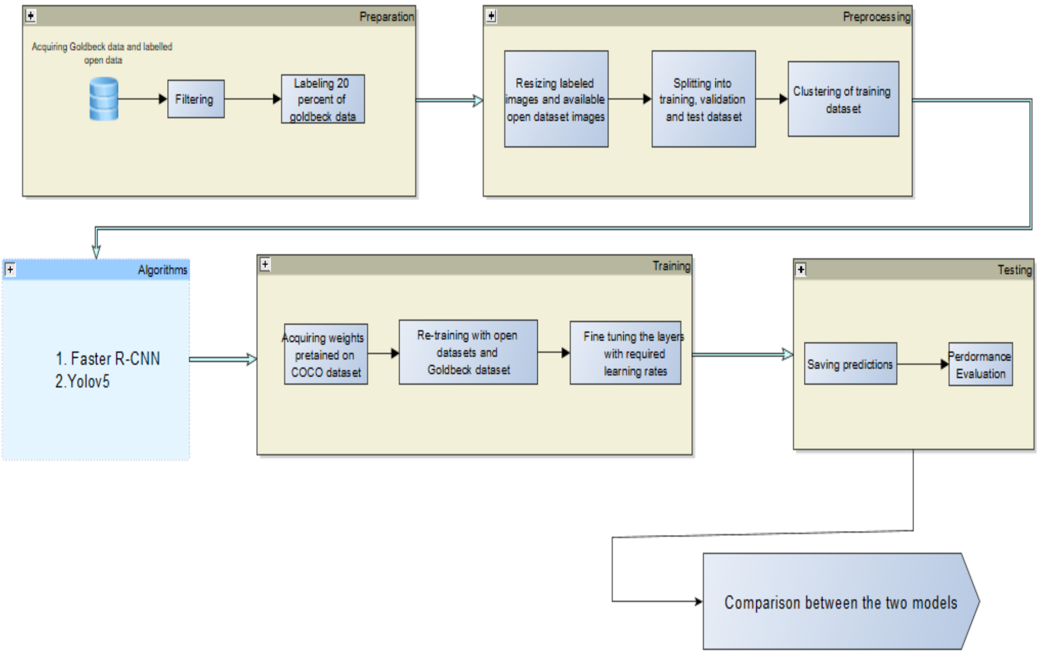

Preparation
Acquiring Goldbeck data and labelled open data
Filtering
Labeling 20 percent of goldbeck data
Preprocessing
Resizing labeled images and available open dataset images
Splitting into training, validation and test dataset
Clustering of training dataset
Algorithms
1. Faster R-CNN
2.Yolov5
Training
Acquiring weights pretained on COCO dataset
Re-training with open datasets and Goldbeck dataset
Fine tuning the layers with required learning rates
Testing
Saving predictions
Perdormance Evaluation
Comparison between the two models

Figure 30 Methodology

4.4.1 Target Selection:

The target objects intended for detection at the Rinteln demo are persons, precast columns, cranes and transportation trucks.

The following table shows the categories and label for each of the intended targets.

Table 7: Categories and label for each of the intended targets

Category	Label
Machine	crane, truck
Person	person
Structural elements	precast_column

4.4.2 Data Acquisition:

The data obtained from Goldbeck consisted of 39539 images taken over 11 months. The images were taken using a single camera fixed at a defined position. From the pool of images, around 300 images were chosen. The selection of images was made in such a way that, it consisted of images taken during when column mounting for the Rinteln demo site was carried out. Since the images were taken from quite high point, the images can be considered as a panoramic view of the construction site. This had both advantages and disadvantages. Panoramic view can be regarded as good for capturing large machinery, however smaller objects like person were quite hard to label. Since the images were taken using a camera from a fixed position, it caused occlusion of certain regions of the site.

4.4.3 Data Pre-processing

Before annotation of images, the images were pre-processed for getting optimal detection accuracy from the data. The main objectives for pre-processing the images are as follows:

- Removal of objects that are not intended for detection to a certain extend from the frame.
- Elimination of vague data; images that are blurred or are anomalous due to some extreme weather condition. This is quite an important step, as it can have significant impact on the training (Rui Duan, 2022).
- Reduction of original image size in terms of file size and dimensions in order to make further processing of the task easier and reduce the need for high computational power. The main objective here would be to choose an optimal size so as to get the maximum possible accurate predictions.

The original size of the images acquired was 6000×4000 pixels, the images were initially cropped to remove the non-target objects and were further reduced to 416×416, and 1280×1280.

4.4.4 Data annotation

The pre-processed data was annotated for the target objects using a data labelling platform Label box (Labelbox, n.d.). Figure 32 shows the percentage share of each target object in the annotated images.

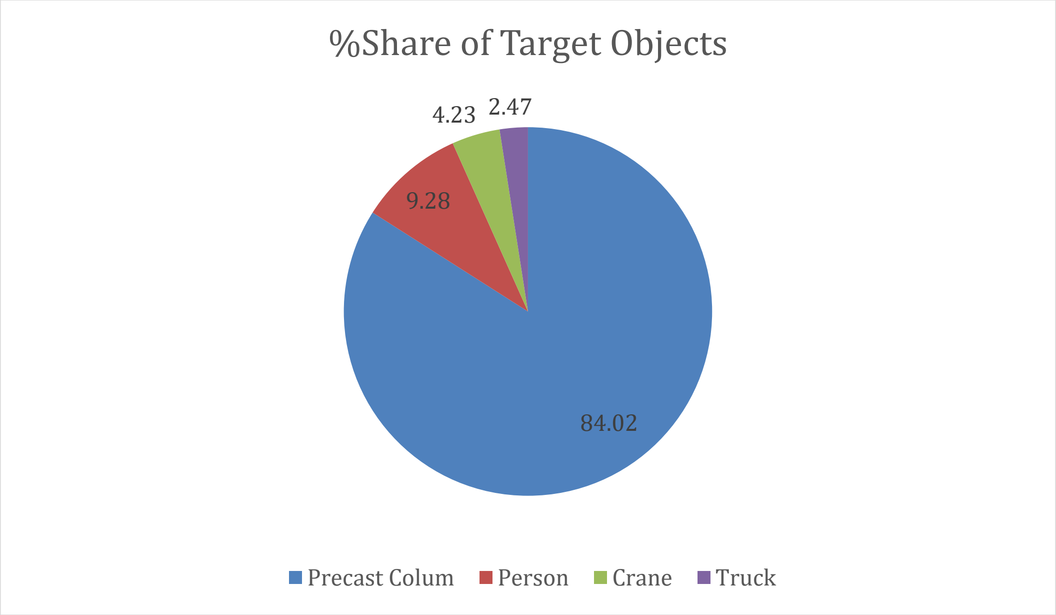


Figure 31: Percentage share of target objects

The annotated data was exported from the platform in JSON format.

4.4.5 Algorithms, Training and Testing

For object detection, two different algorithms were chosen so as to suggest an appropriate algorithm for similar tasks in the project and to extend it to activity recognition. One algorithm was chosen from family of one stage detectors and the other one from two-stage detectors.

Among two-stage detectors, Faster RCNN is chosen and among the one stage detectors, YOLO v5 is used owing to its high speed of detection and accuracy. However, no one architecture can be suggested as the best one, it is highly dependent on the use cases and data available (Shrey Srivastava, 2021). YOLO v5 was implemented using PyTorch framework and Faster R-CNN TensorFlow 2.0 API in python language.

The dataset for both the cases were divided into 70% training set, 20% validation set and 10% testing set. Mean average precision (mAP) is the metric chosen to assess the two models. mAP encapsulates submatrices including confusion matrix, Intersection over Union, recall and precision.

For YOLO v5 600 epochs for chosen for training, but after 476 epochs the model failed to show any improvement. The weights were initialized using the pretrained weights on MS COCO dataset. The model achieved an overall mAP of 83.5%.

Fig. 4 shows the training results.

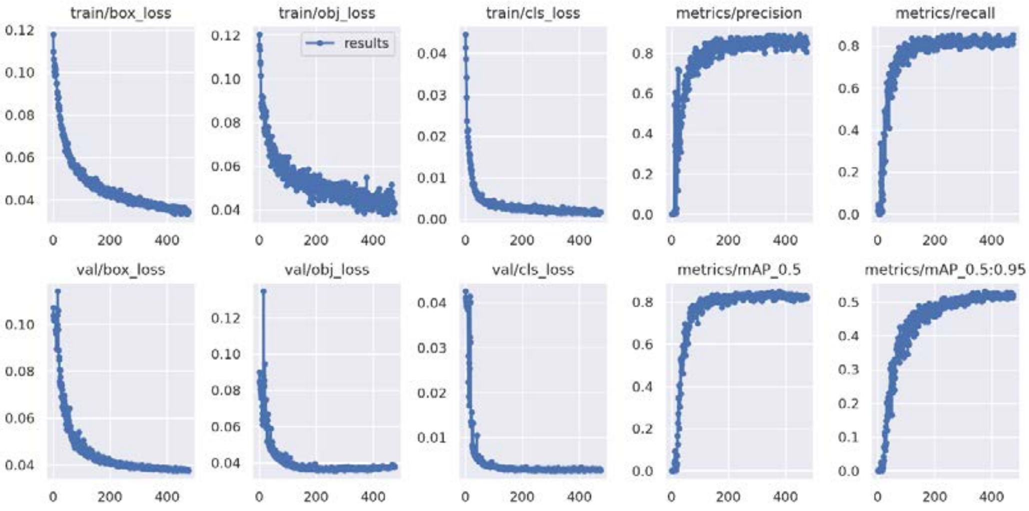


Figure 32: Results- YOLO v5

Table 8: Performance of YOLO v5 on Goldbeck data set

Class	Precision	Recall	mAP
Overall	0.868	0.824	0.835
Crane	0.968	0.92	0.959
Person	0.705	.552	.517
Precast Column	0.967	.936	.984
Truck	0.833	.889	.881

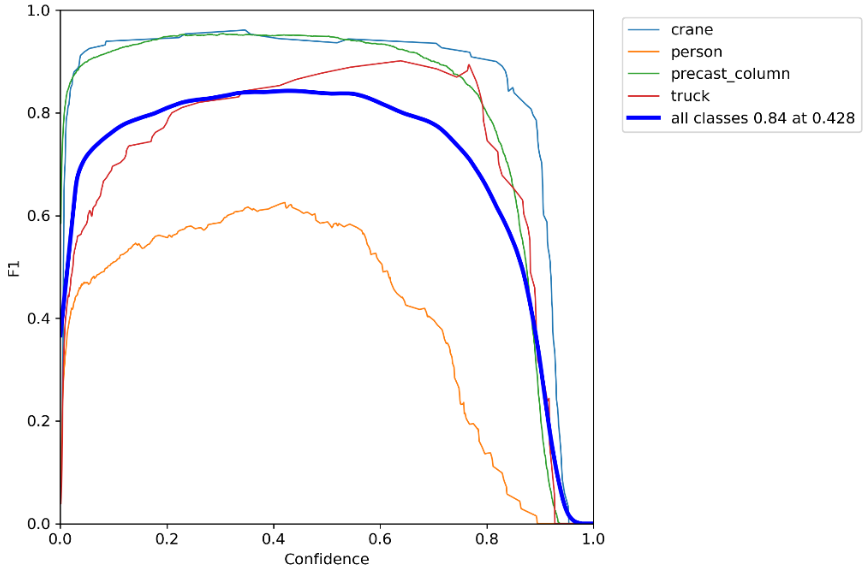


Figure 33 : F1(YOLO v5)

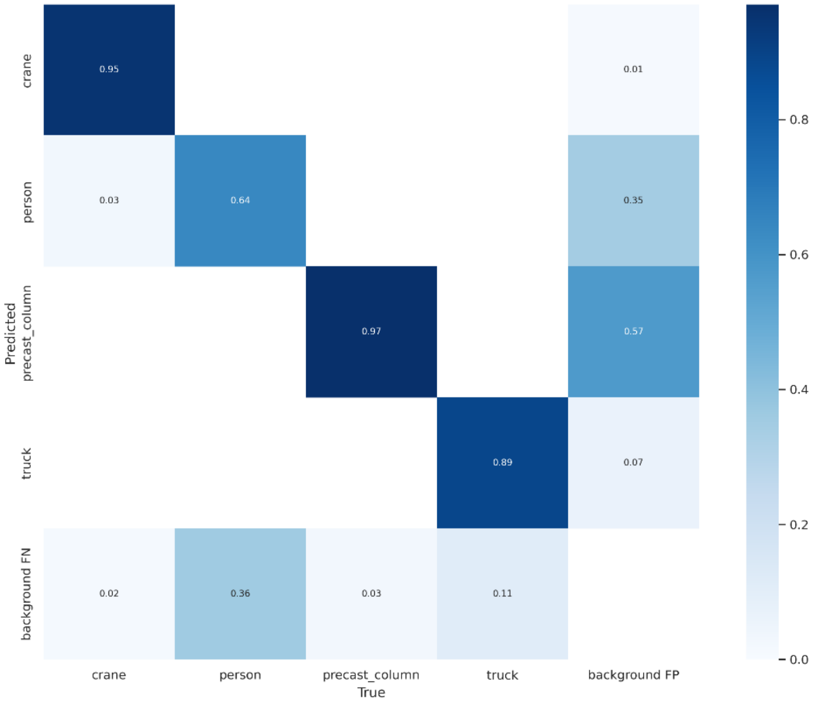


Figure 34: Confusion matrix(YOLO v5)

For Faster R-CNN 50000 steps were chosen for training. The model achieved an overall mAP of 58.0%. And the final loss was logged as 0.512. The training time was quite high as compared to YOLO v5.

4.5 Object Recognition Examples:

The following images shows object recognition examples at Rinteln demo case using YOLO v5 architecture.

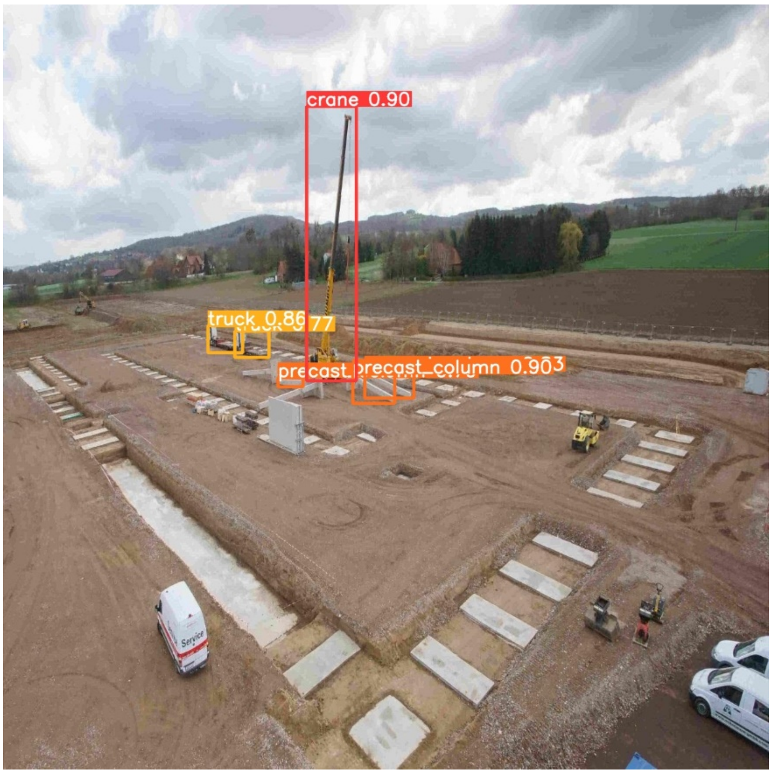


Figure 35: Site object detection example 1

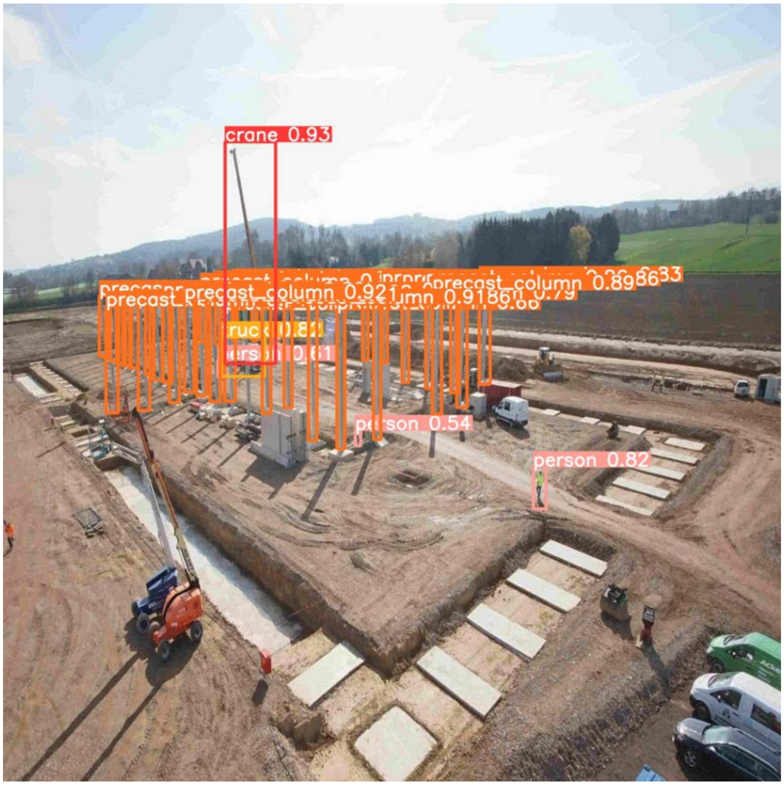


Figure 36: Site object detection example 2

4.6 Discussion

This study was able to successfully employ the state-of-the-art object detection algorithms to detect objects at construction sites. The implemented model was successfully applied to images obtained from Rinteln demo site. From the experiments conducted on the data set, it can be concluded the YOLO v5 architecture outperforms

Faster R-CNN in terms of both detection speed and accuracy and requires less computational resources. However, one particular model cannot be considered as the best, as it largely depends on the use cases.

It can also be observed from the results that, it was quite hard to detect persons in the site owing to the fact that they were constituted of comparatively small number of pixels than other target objects. The model could benefit from employing multiple shooting methods for photography. Also, the data set used is for training is quite small, enrichment of the data set is necessary for better results. YOLO v5 based detector could make predictions in comparatively shorter time than Faster R-CNN. Therefore, YOLO v5 based architecture can be used to extend the current work for activity recognition in future

5 Conclusion

This deliverable presents the progress attained in the scope of the ASHVIN project with regard to the task T3.1. The aim of the work performed within this task was to analyse and develop algorithms that will be deployed for extracting features, identified as KPIs, of the constructions and comprise the base data for the higher level of implementations.

Construction site images, as instant records of the state of the construction site, contain rich information, which makes them natural spaces for automatic construction process monitoring. On the one hand, the popularity of built-in camera equipment makes it feasible to obtain massive free images from the construction site. On the other hand, advanced software techniques provide powerful tools for extracting useful information from daily images.

In D3.1 we have presented three different approaches for visual data processing on construction sites, a 3D representation for digital twin augmentation, an AI-based drone-based defect detection approach with pixel segmentation, and a mixture of object detection for activity recognition. In practice, monitoring the progress of a construction project may require a combination of multiple methods. Therefore, this report combines different relevant technologies and methods into a comprehensive technology path, always in regards to ASHVIN demonstration cases.

The report provides a set of methodologies and solutions using several image-based technologies for monitoring and inspection of different types of structures at different stages of their life cycle. Using digital technologies, we have proposed methods that include performance indicators which could be adopted for construction and maintenance processes of structures and implemented in digital twins. The usage of proposed methodologies will may contribute to result in less resources (human, material and machine), lower costs for performing construction monitoring tasks and more energy efficient practices.

6 REFERENCES

Alex Braun, S. T. (2020). Improving progress monitoring by fusing point clouds, semantic data and computer vision. *Automation in Construction*.

Aritra Pal, S.-H. H. (n.d.). *VISION BASED CONSTRUCTION SITE MONITORING: A REVIEW.*

Badrinarayanan, V., Kendall, A., & Cipolla, R. (2017). Segnet: A deep convolutional encoder-decoder architecture for image segmentation. *IEEE transactions on pattern analysis and machine intelligence, 39*, 2481–2495.

Bang, S., Park, S., Kim, H., & Kim, H. (2019). Encoder–decoder network for pixel-level road crack detection in black-box images. *Computer-Aided Civil and Infrastructure Engineering, 34*, 713–727.

Bukhsh, Z. A., Anžlin, A., & Stipanović, I. (2021). BiNet: Bridge Visual Inspection Dataset and Approach for Damage Detection. *International Conference of the European Association on Quality Control of Bridges and Structures*, (pp. 1027-1034).

Chai, J., Zeng, H., Li, A., & Ngai, E. W. (2021). Deep learning in computer vision: A critical review of emerging techniques and application scenarios. *Machine Learning with Applications, 6*, 100134.

Chen, L.-C. (2014). Semantic image segmentation with deep convolutional nets and fully connected crfs. *arXiv preprint arXiv:1412.7062.*

Chen, L.-C. a. (2017). Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs. *pattern analysis and machine intelligence*, 834--848.

Chen, L.-C. e. (2017). Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587 .*

Choi, D., Bell, W., Kim, D., & Kim, J. (2021). UAV-Driven Structural Crack Detection and Location Determination Using Convolutional Neural Networks. *Sensors, 21*, 2650.

Christian Koch, K. G. (2015). A review on computer vision based defect detection and condition assessment of concrete and asphalt civil infrastructure. *Advanced Engineering Informatics*.

Czimmermann, T., Ciuti, G., Milazzo, M., Chiurazzi, M., Roccella, S., Oddo, C. M., & Dario, P. (2020). Visual-based defect detection and classification approaches for industrial applications—a survey. *Sensors, 20*, 1459.

Deng, J., Dong, W., Socher, R., Li, L.-J., Li, K., & Fei-Fei, L. (2009). Imagenet: A large-scale hierarchical image database. *2009 IEEE conference on computer vision and pattern recognition*, (pp. 248–255).

DeTone, D., Malisiewicz, T., & Rabinovich, A. (2018, June). SuperPoint: Self-Supervised Interest Point Detection and Description. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops.*

Dusmanu, M., Rocco, I., Pajdla, T., Pollefeys, M., Sivic, J., Torii, A., & Sattler, T. (2019). D2-net: A trainable cnn for joint detection and description of local features. *arXiv preprint arXiv:1905.03561*.

Fassi, F., Fregonese, L., Ackermann, S., & De Troia, V. (2013). Comparison between laser scanning and automated 3d modelling techniques to reconstruct complex and extensive cultural heritage areas. *International archives of the photogrammetry, remote sensing and spatial information sciences, 5*, W1.

Feng, X., Jiang, Y., Yang, X., Du, M., & Li, X. (2019). Computer vision algorithms and hardware implementations: A survey. *Integration, 69*, 309-320.

Frahm, J.-M., Fite-Georgel, P., Gallup, D., Johnson, T., Raguram, R., Wu, C., . . . others. (2010). Building rome on a cloudless day. *European conference on computer vision*, (pp. 368–381).

Frahm, J.-M., Pollefeys, M., Lazebnik, S., Gallup, D., Clipp, B., Raguram, R., . . . Johnson, T. (2010). Fast robust large-scale mapping from video and internet photo collections. *ISPRS Journal of Photogrammetry and Remote Sensing, 65*, 538–549.

Fuhrmann, S., Langguth, F., & Goesele, M. (2014). Mve-a multi-view reconstruction environment. *GCH*, (pp. 11–18).

Fujita, Y., & Hamamoto, Y. (2011). A robust automatic crack detection method from noisy concrete surfaces. *Machine Vision and Applications, 22*, 245–254.

Furukawa, Y. (2014). Photo-Consistency. In K. Ikeuchi (Ed.), *Computer Vision: A Reference Guide* (pp. 595-597). Boston, MA: Springer US. doi:10.1007/978-0-387-31439-6_204

G.H. Beckman, D. P.-J. (2019). Deep learning-based automatic volumetric damage quantification using depth camera. *Automation in Construction*.

Han, X.-F., Laga, H., & Bennamoun, M. (2019). Image-based 3D object reconstruction: State-of-the-art and trends in the deep learning era. *IEEE transactions on pattern analysis and machine intelligence, 43*, 1578-1604.

Hirschmüller, H. (2005). Accurate and efficient stereo processing by semi-global matching and mutua information. *Proceedings - 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2005, II*(2), 807-814.

Kucuksubasi, F., & Sorguc, A. (2018). Transfer learning-based crack detection by autonomous UAVs. *arXiv preprint arXiv:1807.11785*.

Labelbox. (n.d.). Retrieved from https://labelbox.com/

Laga, H. (2019). A survey on deep learning architectures for image-based depth reconstruction. *arXiv preprint arXiv:1906.06113*.

LeCun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *nature, 521*, 436-444.

Lei, B., Wang, N., Xu, P., & Song, G. (2018). New crack detection method for bridge inspection using UAV incorporating image processing. *Journal of Aerospace Engineering, 31*, 04018058.

Leonardis, A., Bischof, H., Pinz, A., Bay, H., Tuytelaars, T., & Van Gool, L. (2006). Computer Vision – ECCV 2006 SURF: Speeded Up Robust Features. *Computer Vision – ECCV 2006, 3951*(July 2006), 404-417-417.

Li, C., Guo, J., Porikli, F., & Pang, Y. (2018). LightenNet: A convolutional neural network for weakly illuminated image enhancement. *Pattern recognition letters, 104*, 15-22.

Liu, Y., Yao, J., Lu, X., Xie, R., & Li, L. (2019). DeepCrack: A deep hierarchical feature learning architecture for crack segmentation. *Neurocomputing, 338*, 139–153.

Long, J. a. (2015). Fully convolutional networks for semantic segmentation. *computer vision and pattern recognition* (pp. 3431--3440). IEEE.

Low, D. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 91-110.

McCulloch, B. (n.d.). Automating Field Data Collection in Construction Organizations. *Construction Congress V: Managing Engineered Construction in Expanding Global Markets.*

Meshroom, A. (n.d.). 3D Reconstruction Software.

Moulon, P., Monasse, P., Marlet, R., & others. (2014). Openmvg. an open multiple view geometry library. *Openmvg. an open multiple view geometry library*.

Ni, F., Zhang, J., & Chen, Z. (2019). Pixel-level crack delineation in images with convolutional feature fusion. *Structural Control and Health Monitoring, 26*, e2286.

Nipun D. Nath, A. H. (2020). Deep learning for site safety: Real-time detection of personal protective equipment. *Automation in Construction*.

Nishikawa, T., Yoshida, J., Sugiyama, T., & Fujino, Y. (2012). Concrete crack detection by multiple sequential image filtering. *Computer-Aided Civil and Infrastructure Engineering, 27*, 29–47.

Nistér, D. (2004). An efficient solution to the five-point relative pose problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 26*(6), 756-770.

OpenDroneMap Authors, O. D. (2020). A command line toolkit to generate maps, point clouds, 3D models and DEMs from drone, balloon or kite images. *OpenDroneMap/ODM GitHub*, 2020.

Peiyuan Jiang, D. E. (2021). A Review of Yolo Algorithm Developments. *The 8th International Conference on Information Technology and Quantitative Management .*

Peng, C. a. (2017). Large Kernel Matters--Improve Semantic Segmentation by Global Convolutional Network. *computer vision and pattern recognition* (pp. 4353--4361). IEEE .

Ronneberger, O., Fischer, P., & Brox, T. (2015). U-net: Convolutional networks for biomedical image segmentation. *International Conference on Medical image computing and computer-assisted intervention*, (pp. 234–241).

Rui Duan, H. D. (2022). *SODA: Site Object Detection dAtaset for Deep Learning in Construction.*

Salman, M., Mathavan, S., Kamal, K., & Rahman, M. (2013). Pavement crack detection using the Gabor filter. *16th international IEEE conference on intelligent transportation systems (ITSC 2013)*, (pp. 2039–2044).

Sarlin, P.-E., Cadena, C., Siegwart, R., & Dymczyk, M. (2019). From coarse to fine: Robust hierarchical localization at large scale. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, (pp. 12716–12725).

Sarlin, P.-E., DeTone, D., Malisiewicz, T., & Rabinovich, A. (2020). Superglue: Learning feature matching with graph neural networks. *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, (pp. 4938-4947).

Sattler, T., Maddern, W., Toft, C., Torii, A., Hammarstrand, L., Stenborg, E., . . . Pajdla, T. (2018, June). Benchmarking 6DOF Outdoor Visual Localization in Changing Conditions. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR).*

Schonberger, J. L., & Frahm, J.-M. (2016, jun). Structure-from-motion revisited. *Proceedings of the IEEE conference on computer vision and pattern recognition* (pp. 4104-4113). Las Vegas, NV, USA: IEEE. doi:10.1109/CVPR.2016.445

Shrey Srivastava, A. V. (2021). Comparative analysis of deep learning image detection algorithms. *Journal of Big Data*.

Son, C. K. (2010). 3D structural component recognition and modeling method using color and 3D data for construction progress monitoring. *Automation in Construction*, 844-854.

Song, X., Huang, J., Cao, J., & Song, D. (2022). Feature spatial pyramid network for low-light image enhancement. *The Visual Computer*, 1-11.

Sturm, P. (2001). On focal length calibration from two views. *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2*, 145-150.

Suman Paneru, I. J. (2021). Computer vision applications in construction: Current state, opportunities & challenges. *Automation in Construction*.

Sweeney, C. (2016). Theia multiview geometry library. *Tutorial & reference*.

Taira, H., Okutomi, M., Sattler, T., Cimpoi, M., Pollefeys, M., Sivic, J., . . . Torii, A. (2018). InLoc: Indoor visual localization with dense matching and view synthesis. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, (pp. 7199–7209).

Tan, M., & Le, Q. (2019). Efficientnet: Rethinking model scaling for convolutional neural networks. *International conference on machine learning*, (pp. 6105–6114).

Ting Kong, W. F. (2021). Computer vision and long short-term memory: Learning to predict unsafe behaviour in construction. *Advanced Engineering Informatics*.

Varun Kumar Reja, K. V. (2022). Computer vision-based construction progress monitoring. *Automation in Construction*.

W. Fang, P. E. (n.d.). Computer Vision and Deep Learning to Manage Safety in Construction: Matching Images of Unsafe Behavior and Semantic Rules. *IEEE Transactions on Engineering Management*.

Wang, R., Zhang, Q., Fu, C.-W., Shen, X., Zheng, W.-S., & Jia, J. (2019). Underexposed photo enhancement using deep illumination estimation.

Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, (pp. 6849-6857).

Wang, W., Wei, C., Yang, W., & Liu, J. (2018). Gladnet: Low-light enhancement network with global awareness. *2018 13th IEEE international conference on automatic face \& gesture recognition (FG 2018)*, (pp. 751-755).

Weili Fang, B. Z. (2019). A deep learning-based approach for mitigating falls from height with computer vision: Convolutional neural network. *Advanced Engineeering Informatics*.

Wu, C. (2011). VisualSFM: A visual structure from motion system. *http://www. cs. washington. edu/homes/ccwu/vsfm*.

Xiaochun Luo, H. L., & Dongping Cao, F. D. (n.d.). Recognizing Diverse Construction Activities in Site Images via Relevance Networks of Construction Related Objects Detected by Convolutional Neural Networks.

Xiongwei Wua, D. S. (n.d.). Recent Advances in Deep Learning for Object Detection.

Yan, Q., Yang, L., Zhang, L., & Xiao, C. (2017). Distinguishing the indistinguishable: Exploring structural ambiguities via geodesic context. *Proceedings of the IEEE conference on computer vision and pattern recognition*, (pp. 3836–3844).

Yang, M.-D., Chao, C.-F., Huang, K.-S., Lu, L.-Y., & Chen, Y.-P. (2013). Image-based 3D scene reconstruction and exploration in augmented reality. *Automation in Construction, 33*, 48–60.

Yang, X., Li, H., Yu, Y., Luo, X., Huang, T., & Yang, X. (2018). Automatic pixel-level crack detection and measurement using fully convolutional network. *Computer-Aided Civil and Infrastructure Engineering, 33*, 1090–1109.

Yeum, C. M., & Dyke, S. J. (2015). Vision-based automated crack detection for bridge inspection. *Computer-Aided Civil and Infrastructure Engineering, 30*, 759–770.

Zhang, C., Nateghinia, E., Miranda-Moreno, L. F., & Sun, L. (2022). Pavement distress detection using convolutional neural network (CNN): A case study in Montreal, Canada. *International Journal of Transportation Science and Technology, 11*, 298–309.

Zhang, L., Yang, F., Zhang, Y. D., & Zhu, Y. J. (2016). Road crack detection using deep convolutional neural network. *2016 IEEE international conference on image processing (ICIP)*, (pp. 3708–3712).

Zhang, Y. (2021). Safety Management of Civil Engineering Construction Based on Artificial Intelligence and Machine Vision Technology. *Artificial Intelligence and Internet of Things (IoT) in Civil Engineering*.

Žnidarič, A., Kreslin, M., Anžlin, A., & Krivic, A. (2020). Detection of Delaminated and Cracked Concrete with Unmanned Aerial Vehicles. *Routes/Roads, 386*.

Zou, Q., Cao, Y., Li, Q., Mao, Q., & Wang, S. (2012). CrackTree: Automatic crack detection from pavement images. *Pattern Recognition Letters, 33*, 227–238.