xR4DRAMA

Extended Reality For DisasteR management And Media planning

H2020-952133

D4.2

Multi-sourced 3D reconstruction of outdoors spaces

Dissemination level:	Public
Contractual date of delivery:	Month 25, 30 November 2022
Actual date of delivery:	Month 27, 16 January 2023
Work package:	WP4 Interactive AR/VR solutions for media production and hazard management
Task:	T4.4 3D reconstruction of the area
Type:	Demonstrator
Approval Status:	Final version
Version:	1.0
Number of pages:	52
Filename:	D4.2_xR4Drama_Multi_sourced_3D_ 20230116_v1.0 .pdf

Abstract

This deliverable documents on the image-based 3D reconstruction service of the xR4DRAMA platform, which is developed as a tool for the users in the office – pre-emergency users of PUC1 and journalists and production managers of PUC2 to build photorealistic 3D models from aerial images. The service also works for carefully taken ground level images. The 3D models with photographic texture are optimized for use in the Authoring tool and the Virtual Reality tool. The models are simplified, georeferenced, and automatically imported in a Unity server to create asset bundles for maximum performance in VR.

co-funded by the European Union

History

Version	Date	Reason	Revised by
0.1	18/08/2022	Initialisation of the document	Christos Stentoumis
0.2	07/09/2022	Main text	Monika Sara-Kawka
0.3	10/10/2022	up2metric internal review	Christos Stentoumis
0.4	25/10/2022	2nd version	Monika Sara-Kawka
0.5	22/12/2022	Issue draft version for internal review by CERTH	Ilias Kalisperakis
1.0	16/01/2023	Final revised version	Ilias Kalisperakis

Author list

Organization	Name	Contact Information
up2metric	Christos Stentoumis	christos@up2metric.com
up2metric	Panagiotis Bikiris	panagiotis.bikiris@up2metric.com
up2metric	Monika Sara-Kawka	monika-sara.kawka@up2metric.com
up2metric	Ilias Kalisperakis	ilias@up2metric.com
CERTH	Theodora Pistola	tpistola@iti.gr
CERTH	Konstantinos Chatzistavros	konschat@iti.gr
CERTH	Stefanos Vrochidis	stefanos@iti.gr

Executive Summary

This deliverable reports the results of the activities carried out in the frame of WP4, regarding the 3D reconstruction of outdoors areas from image or satellite data. It goes with the completion of task T4.4 and documents the workflows, the algorithms and the online service that was developed in order to generate high fidelity 3D models of urban and country areas.

Its first part is dedicated on image-based 3D reconstruction by means of a fully automated Structure-From-Motion (SfM) approach. The second part presents the combination of Satellite images and available DEMs for the rapid extraction of the landscape of larger areas. All 3D models are complemented with photographic texture and are optimized for use in the Authoring tool and the Virtual Reality tool.

All terrain/ 3D models are geo-referenced and added to layers of a 3D GIS (T5.3), to fuel the AR geo-localization algorithm for visual and GIS-assisted navigation and the VR environment in WP4.

Abbreviations and Acronyms

API	Application Programming Interface
AR	Augmented Reality
DEM	Digital Elevation Model
GIS	Geographic Information System
GPS	Global Positioning System
MVS	Multi-View Stereo
PUC	Pilot Use Case
RANSAC	RANdom SAmple Consensus
SLAM	Simultaneous Localization And Mapping
SfM	Structure from Motion
UAV	Unmanned Aerial Vehicle
VR	Virtual Reality

Table of Contents

List of Figures

List of tables

1 INTRODUCTION

The xR4DRAMA 3D reconstruction service consists of two separate web services. The first service is responsible to generate photorealistic 3D mesh models from a series of overlapping images, typically captured from drones but it can also handle carefully taken ground level images from a handheld camera or a mobile phone (Task T2.2 – Space sensing using drones and cameras). The second can be conceived as an extension of the Remote Sensing Service of the xR4DRAMA platform (T2.3) which allows the request of satellite images and elevation data (DEMs). These data are combined and further processed to generate 3D mesh models of large areas.

The deliverable starts with an introduction (Section 1) where the connection of the 3D reconstruction service with the xR4DRMA user and technical requirements is described. Section 2 provides a review of 3D reconstruction techniques. In Section 3 the basic steps of image-based 3D reconstruction are analyzed. Section 4 is dedicated in the description of the xR4DRAMA 3D reconstruction services. Examples of the services are also presented. General conclusions are drawn is Section 5.

1.1 User & technical requirements

An analysis regarding the user and technical requirements and their connection to the 3D reconstruction service is essential, since it plays a significant role for the definition of the data types needed, the functionalities of the service and the expected data quality. In this section we describe how multi-sourced 3D reconstruction of outdoors spaces, i.e. the generation of photorealistic 3D models from different data sources, with the use of drones and cameras in the field, or satellite data from relevant online repositories and services, is related to technical and user requirements for each PUC.

The relevant user and technical requirements were defined in deliverable D5.2 and finalized in D6.2 (Table 1 and Table 2).

Req-ID	Name	Description
SYS-6	Immersive visual representation	A functionality that visualizes the location and additional information to enhance situation awareness (e.g. VR, AR)
SYS-7	Initial (Level 1) situation awareness for control room staff	System can present available information in a spatial view (Initial scene view) • Immersive • Non-immersive • PoV • Bird's Eye

Table 1: User requirements relevant to 3D reconstruction service

TR NO.	Technical Requirements	Related components	User Requirements
TR_SM_01	Creates a new job for 3d reconstruction	Space modelling	SYS-6
TR_SM_02	Creates a simplification job.	Space modelling	SYS-6
TR_SM_03	Execute the 3D reconstruction process	Space modelling	SYS-6
TR_SM_04	Return the 3D model as an obj and texture maps	Data storage	SYS-6
TR_SM_05	Consider the visual analytics semantic info in the reconstruction process	Visual analysis	SYS-6
TR_SM_06	Fuse geographic data from the satellite data	GIS/Geoserver	SYS-6

Table 2: Technical requirements for space modelling component

In the following figure, the position of the 3D reconstruction service in the 3D models pipeline is showcased. This is a user triggered pipeline to generate 3D models from data captured by various cameras mounted on drones, as well as hand-held cameras. This helps in layering the status of the area of a project on top of the data captured from the online map sources. The pipeline includes the Space sensors to capture the location, which is then analysed, and 3D reconstruction is done. The data is finally saved in the platform's data storage and can be displayed in the 3 major visualisation tools by the different users.

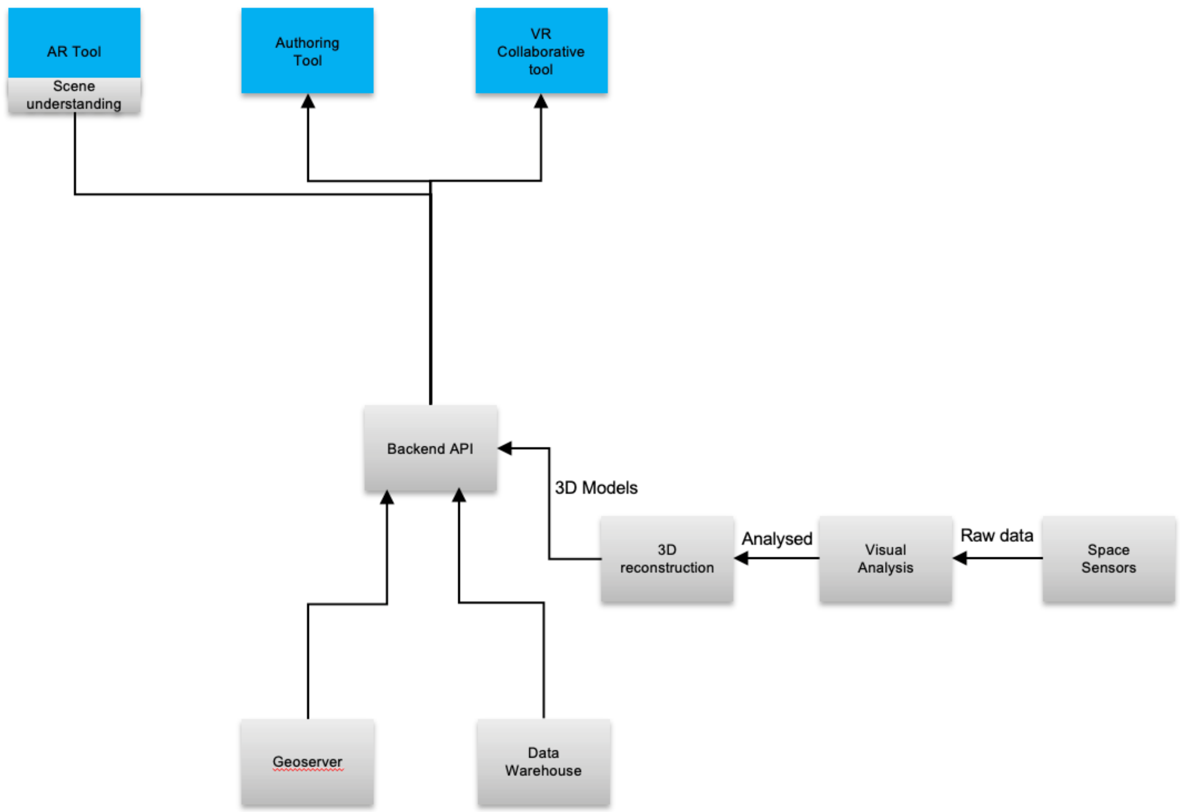

Figure 1: 3D model's pipeline

xR4DRAMA project envisions 2 pilot use cases, disaster management and media production planning.

1.2 PUC1 - Disaster management

The Disaster management use case is oriented to flood prevention and preparedness in Vicenza City (Italy), a highly populated and urbanized area, with extremely complex drainage and irrigation networks and important economic activities, ecological and cultural assets, characterized by high flood risk. The aim of the use case is to demonstrate the use of a VR environment and of extended reality to improve first responders' management before and during a flood event.

In a **pre-emergency management** phase, the xR4DRAMA platform will carry out an initial query on the expected flooding scenario in Vicenza. Detailed 3D models from drones visualized using multiple VR devices inside a single environment are expected to advance the situation awareness in the control room. In an **emergency** phase, the control room will be able to update its situation awareness context by verifying whether the real/current in situ conditions coincide with the forecast of the expected event. Additionally, a 3D representation allows the control room to have a realistic view of the place where the emergency is occurring (e.g. real flooding conditions, position of elements at risk) to plan the intervention. At the same time, it allows the first responders to be guided in the action in situ in safe conditions following indications on the best path (to avoid dangerous areas) and identify important information about the intervention to be performed.

Table 3 and Table 4 include the specific requirements of the space modelling module as they were identified in deliverables D6.1 and D6.2 respectively.

Req Id	Name	Description	Priority 1=highest 3=lowest
DM7	Drone analysis	To analyse video and images from drone to update/enhance the VR scenarios	1
DM8	Satellite images analysis	To analyse satellite images to update/enhance the VR scenarios	1

Table 3: Specific Disaster management requirements for space modelling component

Info-ID	Category	Name	Description	Possible source of information or data	Priority (1=high 4=low)
PUC1-15	Geography, Surroundings	Land use change, past flood events' extent	Information derived by satellite images analysis	LEVEL 1/2/3: Information from COPERNICUS satellites	3

Table 4: Information-related requirements for space modelling component (Disaster management)

1.3 PUC2 - Media production planning

The media production planning use case is oriented in exploiting tools to perform various simulations such as examining a chosen location, collaborate on a virtual representation of it and in general to supervize and manage the production process from a remote position. The space modelling module will enable the production management team in the control room to make creative decisions regarding camera angles and movements or whether they should utilize special equipment like cranes, lights, drones etc. Besides allowing all stakeholders to get a *spatial sense* for a candidate location, higher level of situation awareness (level 3) is expected to be achieved, where users will be able to "see" the visual effect of certain shots through the "camera's eye", to try out longer and complex camera movements and tracking shots, to interact with other team members within the virtual environment and to test different lighting concepts.

Corfu was chosen as a location for the media production planning use case, since it is easily accessible for all consortium partners, weather conditions potentially allow for media production throughout the year and it is more straight-forward for the Athens-based consortium partner up2metric to obtain UAV flight permissions and to organise the data collection, given the COVID-19 travel restrictions.

Table 5 and Table 6 include the specific requirements of the space sensing module as they were identified in deliverables D6.1 and D6.2 respectively.

Req Id	Name	Description	Priority (1=highest 3=lowest)	Comment
MP4	Advanced (Level 3) situation awareness	The system provides (advanced) functionalities that further increase situation awareness and that can be utilized by control room staff at will	1	• Possibility to define camera positions • Possibility to simulate camera movements • Add simulations of time (daylight, night) or specific weather situations Relevant for UC_3

Table 5: Specific Media production planning requirements for space modelling component

Info-ID	Category	Name	Description	Possible source of information or data	Priority (1=high 4=low)
PUC2-11	Facilities	Props&Gear	Possibility to put props/decoration/etc. in the environment	Import of existing 3D Models	2

Table 6: Information-related requirements for space modelling component (Media production planning)

2 REVIEW

2.1 Introduction

Photogrammetric, or image-based, three-dimensional (3D) reconstruction is an image processing technique that allows the generation of digital 3D models from a series of two-dimensional images (Stentoumis, 2018; Rickard et al., 2020) (Figure 2). It is a mature but still active scientific topic in both photogrammetric and computer vision literature, where a series of articles propose good practices and innovative methods to improve efficiency, geometric accuracy, and visual quality, illustrating its popularity as well as its potential to produce high-quality spatial data. The following literature review describes various SFM (Structure from Motion) and MVS (Multi-View Stereo) algorithms that can be used to accomplish each step of the 3D reconstruction pipeline. Additional factors, such as the vision setup, texture, and size of the observed object, are also considered part of the analysis (Ham et al., 2019).

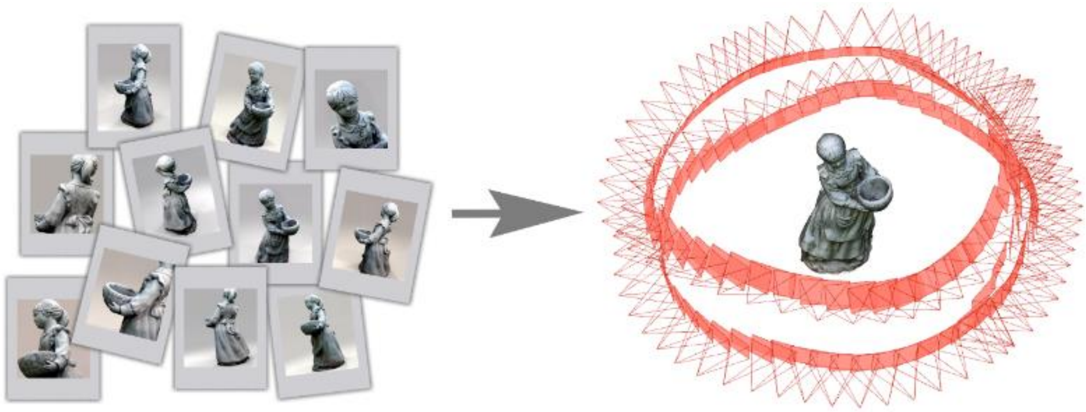

Figure 2: 3D reconstruction using Structure from Motion [Bianco et al., 2018]

To capture reality in 3D, a set of overlapping images, typically taken with hand-held cameras or unmanned aerial vehicles (UAVs), is required (Figure 3). Following a Structure from Motion (SfM) pipeline, the camera parameters and a sparse point cloud of the depicted 3D space are determined. MVS (Multi-View Stereo) algorithms are subsequently applied to densify the sparse point cloud, which is then triangulated into a 3D digital model called a mesh. Lastly, the mesh is textured using the images. Concisely, a typical image-based 3D reconstruction pipeline includes the following steps:

- → Feature detection
- → Keypoints Matching
- → Bundle Adjustment
- → Multi-View Stereo Matching
- → Model Fitting
- → Texture Mapping

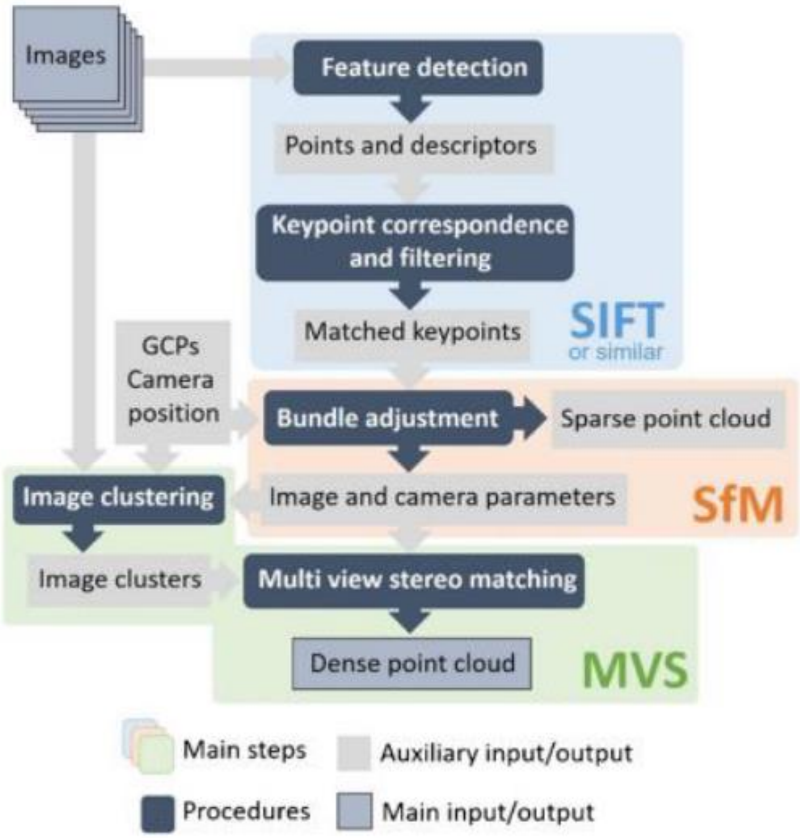

Figure 3: Schematic workflow of the SfM-MVS process resulting [Iglhaut et al., 2019]

3D reconstruction of a scene or object provides information that is currently utilized by a variety of applications, including robot navigation (Ann et al., 2016), object tracking (Ma et al., 2019), pavement distress analysis (Inzerillo et al., 2018), estimation of human poses in 3D (Tome et al., 2017) or estimation of food volumes (Dehais et al., 2017).

In the past decade, Structure-from-Motion (SfM) combined with Unmanned Aerial Vehicles (UAV) footage has become increasingly popular among professionals and amateurs for capturing reality in 3D, due to the advancements in the field that make the whole process automatic and easy to apply for inexperienced users (James et al., 2019).

It is imperative, however, to investigate the potential and limitations of SfM/UAV techniques under different practical application conditions, as well as current good practices and innovative solutions to the most common problems encountered.

Obstructions or occluded areas can complicate the 3D reconstruction process (Stathopoulou et al., 2019). In addition, many tools adopt different terminology and offer to users the ability to interact with different parameters of the pipeline (Remondino et al., 2017). Consequently, new methods for improving SfM/UAV results are constantly being proposed at every stage of the typical SfM process. Among these are the use of learned descriptors (Jiang et al., 2021) in place of handcrafted ones or the adjustment of the image's environmental values (Moon et al., 2021). Moreover, new factors that affect the accuracy of 3D data in various applications are constantly being analysed. Zeybek & Biçici (2021) evaluate two factors affecting the three-dimensional model's accuracy. The first was the flight altitude of the UAV, and the second was the software used for the three-dimensional model reconstruction. According to Li et al. (2021), another factor is the actual terrain which especially in the case of vertical imagery, can reduce the mapping accuracy. However, this accuracy can be increased by a more significant overlap of the images or the use of oblique images.

2.2 Existing 3D reconstruction tools

Many commercial software solutions for image-based 3D reconstruction exist and are suitable for professional and engineering applications. Agisoft Metashape[1], ContextCapture[2], Pix4Dmapper[3], Reality Capture[4] are examples of such software suites. The 3D models generated using these software suites can vary depending on the 3D object (e.g., model quality or processing time). According to reviews found in the literature, Pix4DMapper, in terms of processing time, is faster than Agisoft Metashape (Zarnowski et al., 2015; Georgopoulos et al., 2016) and gives more accurate results (Georgopoulos et al., 2016; Burns & Delparte, 2017). Reality Capture, on the other hand, offers fully automatic processes and is widely adopted in the game development industry (Remondino et al., 2017; Gabara & Sawicki, 2018; Luhmann et al., 2019; Kingsland, 2020). Becker et al. (2018) compare Agisoft Metashape, ContextCapture, and Pix4Dmapper in comparable 3D reconstruction scenarios and suggest that ContextCapture produces 3D models of higher quality at about the same time as Pix4Dmapper.

In parallel to commercial solutions, there exist several open-source and freeware alternatives. 3DF Zephyr Free[5], COLMAP[6], Meshroom[7], OpenMVG[8], Regard3D[9], Visual SFM[10] are among the most widely used. Meshroom is based on the AliceVision 3D computer vision framework (Griwodz et al., 2021) and offers a comprehensive photogrammetric workflow combined with an easy-to-use graphical interface. Thus, it can be considered in many situations a valid alternative to commercial software (Đuric et al., 2021) requiring, however, longer processing times (Reljić et al., 2021). Meshroom has been used in numerous industries, including manufacturing, medicine (Collins et al., 2021), cultural heritage (e Sá et al., 2019), archaeology (Milàn et al., 2020; Lallensack et al., 2020), biology (Chowdhury et al., 2021), and surveillance (Wallner et al., 2021). As a general remark, open-source solutions often offer more parameterization and interchangeability between pipelines, allowing full control of the implemented functions. Moon et al. (2021) and Jiang et al. (2021) performed evaluations of different SfM software tools.

[1] https://www.agisoft.com/
[2] https://www.bentley.com/pl/products/brands/contextcapture
[3] https://www.pix4d.com/
[4] https://www.capturingreality.com/
[5] https://www.3dflow.net/
[6] https://demuc.de/colmap/
[7] https://alicevision.org/#meshroom
[8] https://github.com/openMVG/openMVG
[9] http://www.regard3d.org/
[10] http://ccwu.me/vsfm/index.html

Besides commercial and open-source/freeware stand-alone software tools lately there exist web-based applications, where one can upload images and obtain 3D mesh models. A first web-based 3D reconstruction service has been developed by Vergauwen and van Gool (2006) to meet the needs of the cultural heritage field. The service involves a pipeline that begins with the user uploading images of the object or scene(s) he wishes to reconstruct. The automatic reconstruction process computes the depth maps for the images, as well as the camera calibration, using a server connected to a cluster of computers. Likewise, Tefera et al. (2018) present a web-based 3D imaging pipeline, 3Dnow, that can be used by uploading a set of images through a web interface. As well as producing sparse and dense point clouds, 3Dnow can also generate mesh models. Embedded visualization interfaces allow users to preview directly on the web browser or download 3D models in standard formats.

2.3 Advances in Structure from Motion

New SfM algorithms are being proposed that promise new possibilities and improved results. Incremental SfM is a widely used algorithm (Pollefeys et al., 2004; Snavely et al., 2006; Agarwal et al., 2011; Bianco et al., 2018). As a result, most photogrammetric software utilizes this algorithm. This algorithm finds the correspondences between images and performs an iterative, incremental reconstruction. Below is a description of the algorithms used in each step of this algorithm.

SIFT (Lowe, 2004), SURF (Bay et al., 2008), ORB (Rublee et al., 2011), BRISK (Leutenegger et al., 2011), KAZE (Alcantarilla et al., 2012), AKAZE (Alcantarilla et al., 2013) are some of the most widely used algorithms for Feature Extraction. They are all invariant to changes in scale, rotation, and to a limited extent, affine deformations. Among these, SIFT and SURF are the most widely known; however, SURF offers a computationally less expensive alternative to SIFT (Tareen & Saleem, 2018).

The Feature Matching step can be performed using a variety of methods. When comparing two feature points, the distance of their descriptors is the simplest way to determine whether they are similar. The Hamming distance is measured for binary descriptors (such as ORB and BRISK), whereas the Euclidean distance is measured for gradient-based descriptors (such as SIFT, SURF, etc.). This method is known as Brute-Force Matcher. Despite its simplicity, its major drawback is that it compares every feature point of one image with every feature point of the other. A similar but faster method is FLANN (Fast Library for Approximate Nearest Neighbors) (Muja & Lowe, 2009). As an alternative to comparing a feature point with all feature points of the other image, the comparison is made only with the approximate nearest neighbors in the high-dimensional space. Cascade hashing (Cheng et al., 2014) is another method used. For each feature of an image, the Cascade Hashing algorithm consists of three steps. First, a coarse search is conducted using the LSH method, known as hashing looking in the first step. In this step, buckets described by binary code are created using the LSH method, and the tables of buckets are generated. Each table contains one bucket containing the searched feature point. The Hamming distances are calculated in the second step after remapping the buckets containing the feature point into the higher dimensional Hamming space. As a final step, the two buckets with the shortest Hamming distance are selected, and the Brute-Force search is applied to the feature points of these buckets.

The following algorithms allow the geometric verification of the correspondences of the previous algorithms. As the verification is geometric, the choice of algorithm depends on the process of capturing the images. Specifically, if the object photographed is planar, the images are related through homography. The homography matrix has eight degrees of freedom, and each point has two equations, so four points suffice to solve the matrix (Hartley & Zisserman, 2004). The epipolar geometry and epipolar constraint, whereby a point in one image corresponds to a line in the other image, are exploited when the object is not planar. Whether the intrinsic parameters of a camera are known or unknown, this constraint can be described by the essential matrix E or the fundamental matrix F. In terms of the essential matrix, five degrees of freedom represent the camera's extrinsic parameters. In order to solve this problem, at least five correspondences are required, which is accomplished through the 5-point algorithm (Nister, 2004; Stewénius et al., 2006). Alternatively, the fundamental matrix has seven degrees of freedom, including intrinsic and extrinsic parameters. Therefore, a 7-point or normalized 8-point algorithm with a linear solution can be applied to the solution of this problem (Hartley & Zisserman, 2004).

Concisely, the previous algorithms involved finding the relative orientation of two images and generating 3D points from their correspondences. Then, all images that have a sufficient number of correspondences with the 3D points of these two images are selected. Each new image added in each iteration has six degrees of freedom corresponding to its pose. Based on the solution of the Perspective-n-Point problem, these data can be obtained. The solution to this problem requires at least three correspondences with known 3D points. It is a P3P method (Xiao-Shan Gao et al., 2003) that provides four geometrically feasible solutions to the problem when the minimum correspondences are used. The RANSAC algorithm is typically used to identify a unique solution. Besides P3P, P4P (Bujnak et al., 2008) and P5P (Kukelova et al., 2013) are other related methods. There is also an EPnP (Lepetit et al., 2008) method that requires at least four correspondences in order to solve the problem. In this case, the n points are expressed as a weighted sum of four virtual control points. The final camera pose is solved using the control points as unknowns. Additionally, DLS (Direct Least Squares) (Hesch & Roumeliotis, 2011) calculates all PnP problem solutions as the minima of a nonlinear least-squares cost function. In the previous methods, the cameras were assumed to be calibrated. In addition to estimating extrinsic parameters, DLT (Direct Linear Transform) (Hartley & Zisserman, 2004) also attempts to estimate intrinsic parameters. This method requires at least six correspondences.

Through the application of the prior algorithms, a sparse point cloud can be made denser through triangulation in which new points are found that appear in at least two images. As a result, the corresponding 3D points can be calculated by knowing the camera poses and the correspondences between the images. These points can be determined using two-view or multiple-view geometry (N-view). According to its name, the Mid-point method determines the midpoint between two back-projected rays by finding the shortest line between them. Another option is to calculate a 3D coordinate using the DLT method, which solves an unknown matrix using the SVD method.

Optimizing both the parameters of the camera poses and the points in the resulting sparse point cloud is necessary to eliminate any possible inaccuracies in the previous process. In the last step, Bundle Adjustment is performed using the Levenberg-Marquardt algorithm, which performs optimization based on nonlinear least squares. Two widely known algorithms for solving the nonlinear least squares problem are Multicore Bundle Adjustment (Wu et al., 2011) and Ceres Solver (Agarwal & Mierle, 2012).

As part of the previous steps (algorithms) of geometric verification and the solution to the PnP problem, possible outliers must be removed. RANSAC algorithms (Fischler & Bolles, 1981) or variants are (Chum et al., 2003; Chum & Matas, 2005; Raguram et al., 2008; Moisan et al., 2012; Fragoso et al., 2013) used for this purpose. In brief, the RANSAC (RANdom Sampling Consensus) algorithm calculates the parameters of a mathematical-geometric model based on random combinations of minimum data points. After that, it uses a threshold to separate inliers and outliers. Generally, the solution (calculated values of parameters) chosen is the one with the most inliers (the most consistent solution).

Apart from the incremental SfM algorithm, other algorithms have been proposed. In recent years, interest has shifted to the global SfM algorithm (Cui & Tan, 2015; Zhu et al., 2018; Barath et al., 2021) because of its higher computational efficiency and reconstruction accuracy. Instead of registering each image one at a time, as in incremental SfM, global SfM resolves all camera poses simultaneously (Zhu et al., 2018).

In order to use the global SfM algorithm effectively, rotational and mainly translational averaging are challenging tasks, as is its sensitivity to noisy data (Cui & Tan, 2015). New ideas are constantly being proposed to overcome these challenges. Cui et al. (2016) proposes a robust two-step translation averaging strategy. According to Martinec & Pajdla (2007), the rotation averaging problem can be viewed as a linear system, and rotation parameterizations can be relaxed. The Lie-algebra representation of Govindu (2004) is used to achieve better results by combining a robust L1 optimization (Chatterjee & Govindu, 2013). Cui & Tan (2015) constructed a sparse depth image for each camera to overcome both challenges.

A hybrid algorithm of global and incremental SfM was proposed and implemented by Zhu et al. (2017), combining and integrating both methods' advantages and providing highly accurate results. Furthermore, other hybrid formulations have also been presented (Bhowmick et al., 2015; Sweeney et al., 2016). On the other hand, Chen et al. (2017) presented a tree-structured SfM algorithm, which is more efficient and robust to noise than the traditional SfM algorithm. Furthermore, it was recently proposed by Li et al. (2022) to eliminate the background of the target object, thereby reducing its impact on the target object. Additionally, it overcomes the weakness of conventional SfM algorithms when no background feature points are present.

In recent years, deep learning algorithms and active learning algorithms have dominated the field of computer vision. Therefore, it is proposed to use machine learning or deep learning technology to solve the important problem of reconstructing a 3D object. Deep neural networks are often used to reconstruct 3D shapes from a single image (or several images) using their ability to reconstruct 3D shapes. Based on machine learning, Zhu & Zhou (2022) propose a method for virtual-real fusion 3D reconstruction technology with application to various fields, such as medicine, artificial intelligence, and education. A technique of active learning has also been suggested in the literature. For instance, Kowdle et al. (2011) used an energy minimization framework for piecewise planar reconstruction but provided support for uncertain regions through simple user interaction. Data-driven active touch is another interesting problem investigated (Smith et al., 2021).

2.4 3D reconstruction and xR technologies

During the past few years, there has been significant progress in the reconstruction of 3D scenes and the exploration of such data in VR/AR environments, with applications in many fields such as cultural heritage (Bruno et al., 2010), education and surgical simulation (Roh et al., 2021), medical training (Hsieh & Lee, 2018), content creation (Bhattacharjee & Chaudhuri, 2020), virtual fitting (Rhee & Lee, 2021), and hospitality & tourism (Nayyar et al., 2018). Furthermore, solutions to difficult 3D reconstruction problems are presented, particularly when VR/AR technology is considered. For example, 3D reconstruction of indoor environments (Navarro et al., 2017, Manni et al., 2021) or 3D reconstruction of the human body from egocentric viewpoints (Grover et al., 2021).

Producing reliable xR products requires a custom workflow and a lot of effort in terms of time and professionals involved to guarantee this fidelity (Ferdani et al., 2020). Thus, new workflows are constantly proposed to properly model a 3D reconstructed scene (Spallone et al., 2021). Rahaman and Champion (2019) describe a complete workflow from image capture to visualization of the model in VR/AR. Navarro et al., 2017, El Saer (2020[a]), El Saer (2020[b]), and Kalisperakis et al. (2022) present methodologies for accurate image-based 3D reconstruction of outdoor spaces for the needs of immersive VR applications.

3 MAIN STEPS OF 3D RECONSTRUCTION

This chapter describes the photogrammetric pipeline to obtain photorealistic 3D mesh models from sets of overlapping images that has been adopted for the image-based 3D reconstruction service of xR4DRAMA platform. As mentioned before, Structure from Motion refers to the first part of the workflow that provides the camera parameters (external and even internal orientation) and a sparse point cloud. This sparse cloud is then densified via so-called dense image matching algorithms such as MVS, thereby creating a dense point cloud (Iglhaut et al., 2019).

Feature Extraction

Finding the orientations of the images (exterior and interior) depends on tie points. Therefore, there must be a considerable number of reliable points. As a result, the first step is to extract (automatically) keypoints, i.e., those that can be identified from image to image with certainty. In addition to being distinct, such points should also be invariant to scale variations, rotations, perspective distortions, and illumination distortions (illumination). Consequently, keypoints in the images may be edges or corners, which are detectable in areas with high contrast or a color change. An effective and well-known algorithm for detecting feature points is SIFT (Scale-Invariant Feature Transform).

This algorithm not only finds some feature points in the images but also creates a unique description for each point, allowing for a comparison of these descriptions to be used for effective matching. Consequently, SIFT aims to extract discrete features invariant to changes in scale and rotation, as well as reliable for matching between images despite possible distortions in radiometry and perspective. Following is a brief description of the algorithm's steps:

- → Scale-space extrema detection
- → Keypoint localization
- → Orientation assignment
- → Keypoint descriptor

Initially, the algorithm searches for features in image pyramids (scale-space) for each image. In this way, the points are invariant to changes in scale. At first, internal copies of the image are created and divided into levels. The number of levels depends on the image's size. Each level's images are half the size of the earlier levels. In parallel, the Gaussian smoothing filter is applied gradually (with increasing standard deviations) within each level. As a result, the Gauss pyramid is obtained. Next, the differences between the images of each level (current and next) are calculated in twos to calculate the DoG (Difference of Gaussian) pyramid. Thus, each point is compared to eight neighbors within its level, nine neighbors above it, and nine neighbors below it (26 in total) (Figure 4). Keypoints are points (pixels) that are maximum, or minimum compared to their neighbors. Since the previous step produced lots of keypoints, the next step is to determine the exact location of the extrema using the Taylor series expansion of scale space. Similarly, points in low contrast areas (threshold) or along edges (angularity criterion) should be rejected.

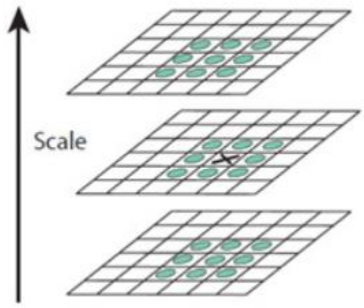

Figure 4: Detected Keypoints as Extremes of DoG [Lowe, 2004]

Next, the scale-invariant keypoints need to be assigned an orientation to become rotation-invariant. This is done by creating an orientation histogram with 36 bins (from $0°$ to $360°$), in which the amount added is proportional to the magnitude of the gradient at that point. The orientation of keypoints is determined by the highest peak of the histogram (which must exceed a threshold). When other peaks exceed 80% of the histogram's maximum, more than one keypoint is created with the exact location and scale but with different orientations.

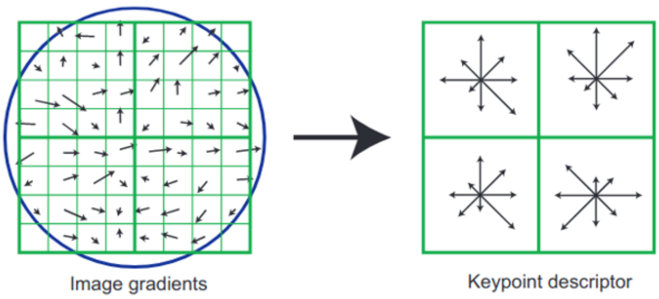

Figure 5: Keypoint descriptor [Lowe, 2004]

Then, its descriptor is created to make the feature point invariant to radiometric distortions such as illumination or changes in angle of view due to rotation between images. To generate the descriptor around each keypoint, a $16x16$ window is created at the scale at which the keypoint was found, and then from this window, a histogram of $4x4$ size sub-blocks in 8 bin orientation is created (Figure 5). This makes the descriptor vector consist of $4x4x8 = 128$ elements for each feature point. Finally, the keypoint descriptor is normalized to unit length to ensure the illumination independence. Then, the values are clipped to the chosen threshold and the resulting vector is again renormalized to unit length.

Keypoints Matching

It is then necessary to determine which images depict the same parts of the object or, in other words, which images overlap. The process should be automated without human intervention. The previous process generated keypoints and their descriptors for each image. By using the descriptors, keypoints can be compared between images. Hence, the assumption can be made that an image with a significant number of feature points that resemble (after comparison) those of another image sees a common part of that object (Bianco et al., 2018).

Although the keypoints are invariant, the above procedure may result in some incorrect matches. This is mainly due to the fact that this procedure ignores the most critical factor, which is the geometric consistency of these correspondences. In particular, if a minimum number of these correspondences are selected to describe the geometric relationships between the images, this should also apply to all other correspondences. Whether the object or scene photographed is planar (homography) or non-planar (epipolar geometry), the method used to describe the geometric relationship of the images will differ. It also varies depending on whether intrinsic camera parameters are known (essential matrix E) or not (fundamental matrix F) if it is not planar.

These methods are usually used in combination with robust estimation techniques such as RANSAC (RANdom SAmple Consensus) or its faster variations (e.g., PROSAC, ARRSAC, EVSAC, or LMed) to remove as many outliers as possible.

Structure from Motion

Finding the position of the cameras usually starts from a pair of images with the largest number of matches. This process of initialization is known as incremental reconstruction. First, the position of the two images is initialized, and their tie points form the first set of points of the sparse point cloud. Then, a new image is added each time, and accordingly, an additional set of tie points is repeatedly added to the sparse cloud, as described below.

In order to add the new tie points to the existing cloud, it is necessary to find the camera's pose (position and rotation) according to a world model. The existing 3D reconstruction points represent the common keypoints of the initial images in 3D. Therefore, the newly added image will share keypoints with one or both images. Through this correspondence, which is no longer only two-dimensional but also three-dimensional, the camera pose of the newly added image can be calculated. Location (X, Y, Z) and rotation (pitch, yaw, and roll) are each camera's six degrees of freedom. To calculate these parameters, the Perspective-n-Point problem must be solved. Several algorithms are proposed for this purpose. Moreover, RANSAC (or its variants) is used to remove outliers in order to ensure that the camera pose estimation is robust.

To add the points of a new recorded image, they must be visible in at least one of the already recorded images whose positions have been estimated. In addition, these points must be located at the intersection of the epipolar lines above the epipolar plane, considering the epipolar geometry and epipolar constraint. However, due to the reprojection error caused by possible inaccuracies of the previous steps, the point will likely not be at the intersection of the lines. Several algorithms consider this inaccuracy in this case, including sampling-based, linear (DLT), 2-view, midpoint, or N-view algorithms.

The Bundle Adjustment phase, performed for every image added to the reconstruction, attempts to minimize the accumulation of errors mentioned above and produce the best values for the 3D reconstructed points and camera calibration parameters. The algorithm used for BA is Levenberg-Marquardt, and the implementations that can be used are Ceres Solver, Multicore BA, or SBA.

Multi-View Stereo Matching

Bundle Adjustment produces extrinsic (and intrinsic) parameters of all cameras and a sparse cloud of 3D points. There are some points shown in the images that cannot be considered feature points, which explains the sparseness of the cloud. Nevertheless, to reconstruct the scene accurately, a dense cloud is necessary. Using Multi-View Stereo (MVS) algorithms, points missed in the previous process are detected. According to the classification given by Seitz et al. (2006), there are four categories of MVS algorithms:

- Voxel-based methods (Furukawa & Ponce, 2006; Seitz & Dyer, 1999; Sinha & Pollefeys, 2005; Treuille et al., 2004; Vogiatzis et al., 2005).
- Surface-evolution-based methods (Bhotika et al., 2002; Kutulakos, 2000; Kutulakos & Seitz et al., 2000; Yang et al., 2003; Zeng et al., 2005).
- Feature-point-growing-based methods (Faugeras et al., 1990; Manessis et al., 2000; Taylor, 2003).
- Depth-map-merging-based methods (Gargallo & Sturm, 2005; Narayanan et al., 1998; Szeliski, 1999).

As a result, voxel-based methods calculate a cost function and create a surface using a 3D volume within a bounding box. A cost function that yields a value below a threshold indicates that the given voxel is part of the object's surface. Likewise, the second category begins with a three-dimensional volume and continuously adds or subtracts voxels to minimize a cost function. Feature points are used in the third category, which combines them according to their visibility to form patches. As a result of the patches, the final surface is formed. Finally, the fourth category combines depth maps from multiple viewpoints into a point cloud.

Model Fitting

After the SfM-MVS workflow is complete, the triangulation of the model can begin. Firstly, the planar subdivision of the set of points (dense cloud) into triangles is done. Next, points are projected in a horizontal plane, and triangles are formed. Then these 2D triangles are mapped into 3D triangles. The points from which the triangles are created are not random but rather so that elongated triangles (small angles) are avoided, as they can interfere with distant points. This type of triangulation is known as Delaunay triangulation, aiming to maximize the triangle's smallest angle. According to Delaunay triangulation, the circle circumscribed by the triangle's points must not have any other points. This is because a line segment has an infinite number of circumscribed circles in the plane, but a triangle only has one. As a result, all points are connected to the two closest ones. At the same time, this criterion ensures that the creation of the triangular network is univocal.

In principle, a Delaunay triangle is one that is contained within a dense point cloud and a whose circumscribed circle is empty, i.e., it does not enclose another point within the cloud. Several methods of Delaunay triangulation exist, such as divide-and-conquer algorithms (Cignoni et al., 1998), randomized incremental algorithms (Devillers, 1998), or indirect methods based on Voronoi diagrams (Lee & Schachter, 1980). Also, constraints can be used for Delaunay triangulation. As in this case, breaklines are used as sides of Delaunay triangles. Using breaklines (discontinuity lines) is common in steep slope areas, such as roads, buildings, streams, etc. Additionally, an interpolated color can be added after triangulation by blending the colors of the three points.

Texture Mapping

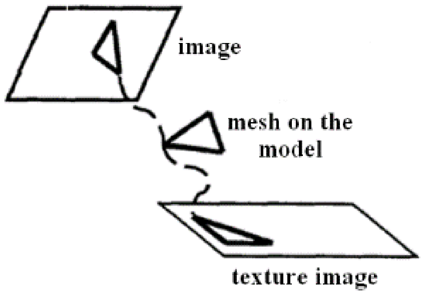

Figure 6: Principle of texture mapping [Li et al., 2010]

It is the last step of the 3D reconstruction that achieves the desired realism of the final model. Because the triangle's color is the result of blending the colors of the points that define it, it cannot accurately reflect its texture. This type of detail has the disadvantage of requiring a vast number of triangles. However, it is possible to reproduce an object's realistic texture directly from its images. It is important not to choose images that are radiometrically different from the others (e.g., due to different lighting) to maintain the color continuity of the model texture.

Creating realistic photorealistic 3D models relies heavily on texture maps. In a few words, each triangle is projected onto the image (Figure 6). In order to create the texture map, the pixels corresponding to that triangle are snapped off and placed in a new image. In other words, a texture map is a collection of smaller images (triangles or faces), usually packed together to reduce texture size.

4 XR4DRAMA 3D SERVICES

4.1 3D-Reconstruction Service

The 3D Reconstruction Service of xR4DRAMA platform is a web application that produces photorealistic 3D models from images and/or videos, completely automatically. In the current development cycle of the service, the primary purpose is to create 3D models to enhance actors' situation awareness on the ground and to facilitate the two use cases of the xR4DRAMA platform.

4.1.1 Implementation of 3D-Reconstruction Service

In this section, the xR4Drama image-based 3D-Reconstruction Service is described in detail, including how 3D models can be generated based on images and videos submitted by the user. For the algorithmic implementation, as described in Section 3 of this deliverable, *AliceVision* (Jancosek and Pajdla 2011) & *Meshroom* (Moulon et al., 2012) software libraries were used. The default pipeline consists of two main stages, SfM and MVS, each consisting of several nodes (Figure 7), including camera initialization, feature extraction, image matching, feature matching, SfM, depth mapping (including the steps of preparation, mapping, and filtering), meshing, and mesh filtering. A mesh decimation step was also added, following the mesh filtering step, during which the resulting model is simplified by 70% (default value).

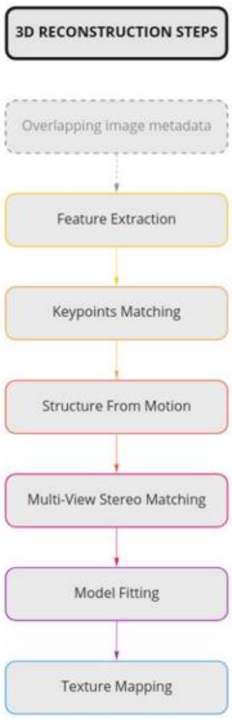

Figure 7: Basic 3D Reconstruction steps

Swagger is a framework that allows the visualization and use of API applications. In practice, Swagger displays interactive REST API documentation. All available nodes are described in detail, along with the information the API receives and returns. At the same time, it takes into account the available roles. For example, an administrator can see more types of queries than simple users.

Figure 8 shows the Swagger of the 3D-Reconstruction Service (https://baremetal.up2metric.com/swagger/). The clients (users) can create a 3D model by posting images and/or videos in ZIP archive format. The service can also download the client's images embedded in a JSON file containing their links. Furthermore, the service supplies the ability to download the model (ZIP file, which contains OBJ file, MTL file, and textures in JPG format) if its creation is complete. In order to use 3D Reconstruction Service, clients must be registered in the system, meaning they must get an API key for authentication and authorization.

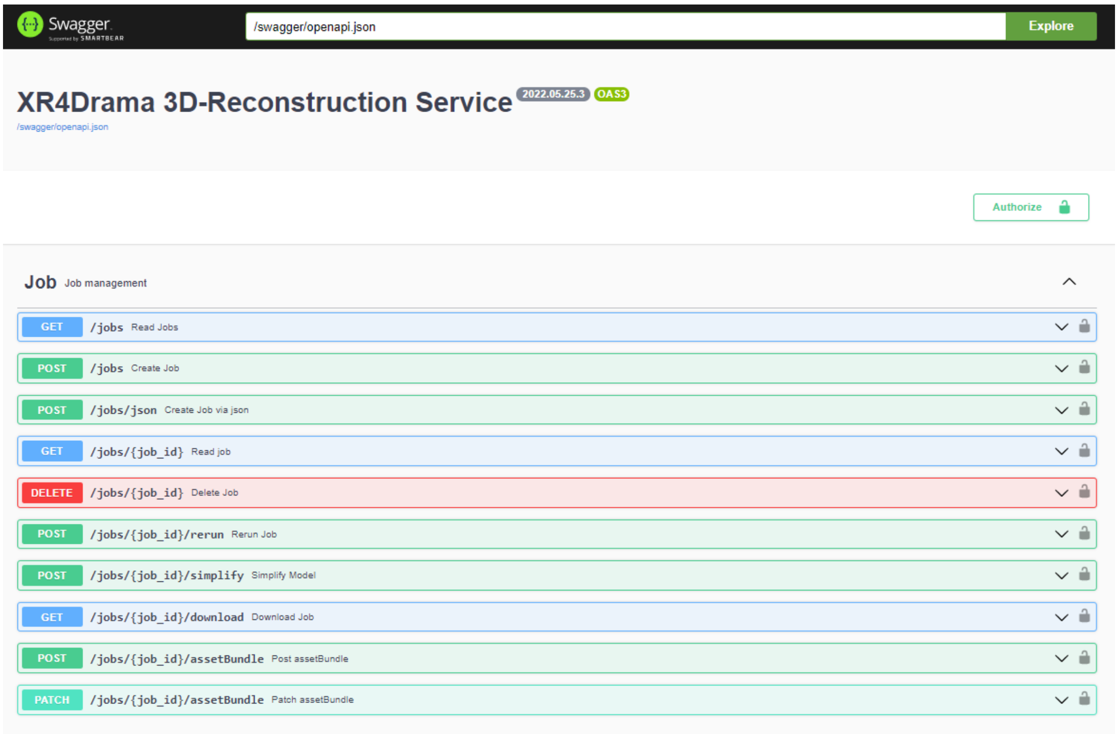

Figure 8: xR4DRAMA 3D-Reconstruction Service Swagger

The overview of the 3D reconstruction service is shown in Figure 9.

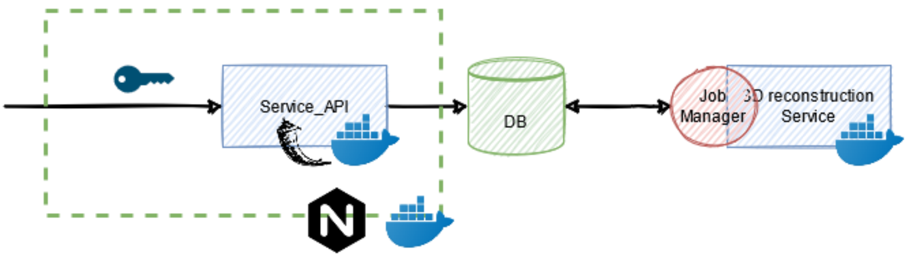

Figure 9: An overview of the components and the data flow in the 3D reconstruction service.

Briefly describing the above diagram, the incoming requests are authenticated by the API key, and the requested jobs are registered in the database as pending jobs. As a response, the Service API returns a job ID. This ID can be used to get the 3D model when the job is completed. The service is looking in the database for pending jobs using a Job Manager (Figure 9). The jobs that were created earlier have higher priority, so the oldest pending job in the database is picked and executed first. In general, the job's status can be described as:

- *pending:* the job waits to be executed
- *in-progress:* the job is currently in progress
- *completed:* the job is completed which means that the results can be downloaded by the user
- *failed:* the job failed for some reason

GET - https://baremetal.up2metric.com/jobs	Read jobs
POST - https://baremetal.up2metric.com/jobs	Create job via images/video
POST - https://baremetal.up2metric.com/jobs/json	Create job via JSON
GET - https://baremetal.up2metric.com/jobs/<*job_id*>	Read job
DELETE - https://baremetal.up2metric.com/jobs/<*job_id*>	Delete job
POST - https://baremetal.up2metric.com/jobs/<*job_id*>/rerun	Rerun job
POST - https://baremetal.up2metric.com/jobs/<*job_id*>/simplify	Simplify job
GET - https://baremetal.up2metric.com/jobs/<*job_id*>/download	Download job
POST - https://baremetal.up2metric.com/jobs/<*job_id*>/assetBundle	Post assetBundle file
PATCH - https://baremetal.up2metric.com/jobs/<*job_id*>/assetBundle	Post assetBundle JSON file

Table 7: REST routes in API

In addition, the user can simplify the model by a certain percentage after the basic 3D reconstruction process (Figure 10).

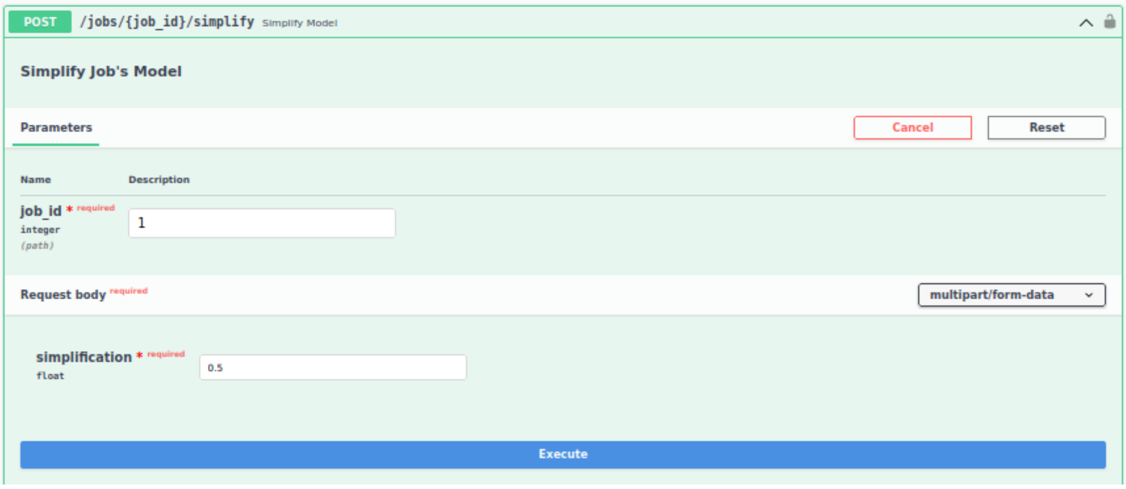

Figure 10: Simplify model

The model resulting from a typical Structure from Motion pipeline is positioned in an arbitrary system at an arbitrary scale. To georeference the resulting 3D model, i.e. to determine the scale, 3D rotation, and 3D position that it has in the real world, a 3D similarity transformation is estimated and applied to the final 3D mesh model. This is possible if the metadata of the images contains GPS data (latitude, longitude, and altitude). In particular, the seven parameters of this transformation are estimated which describe the transformation between the camera poses from the GPS data and the camera poses calculated by SfM. Thus, the ZIP file containing the final textured model in the case where the model is georeferenced also contains a text file with the origin in the EPSG:3395 system (Figure 11, Figure 12).

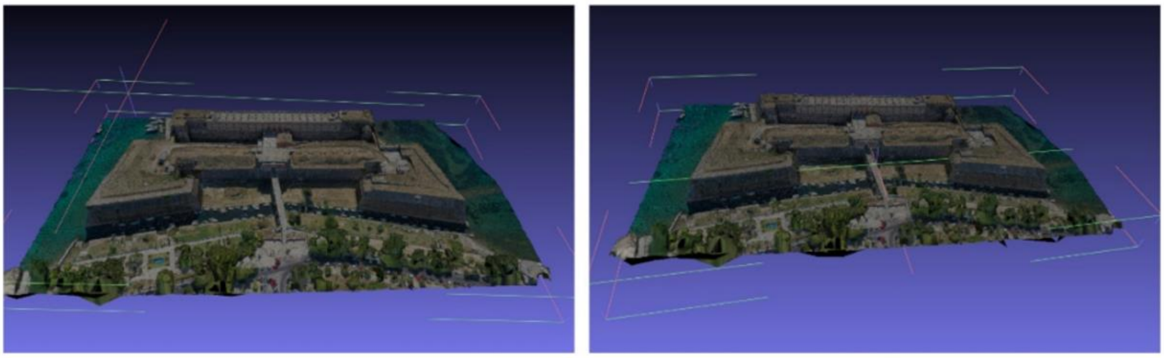

Figure 11: Arbitrary Coordinates System vs World Coordinates System

Figure 12: Comparison of the scale of the model before and after georeferencing

Figure 13 shows the position of a point in a 3D model and the corresponding position of the point in Google Maps.

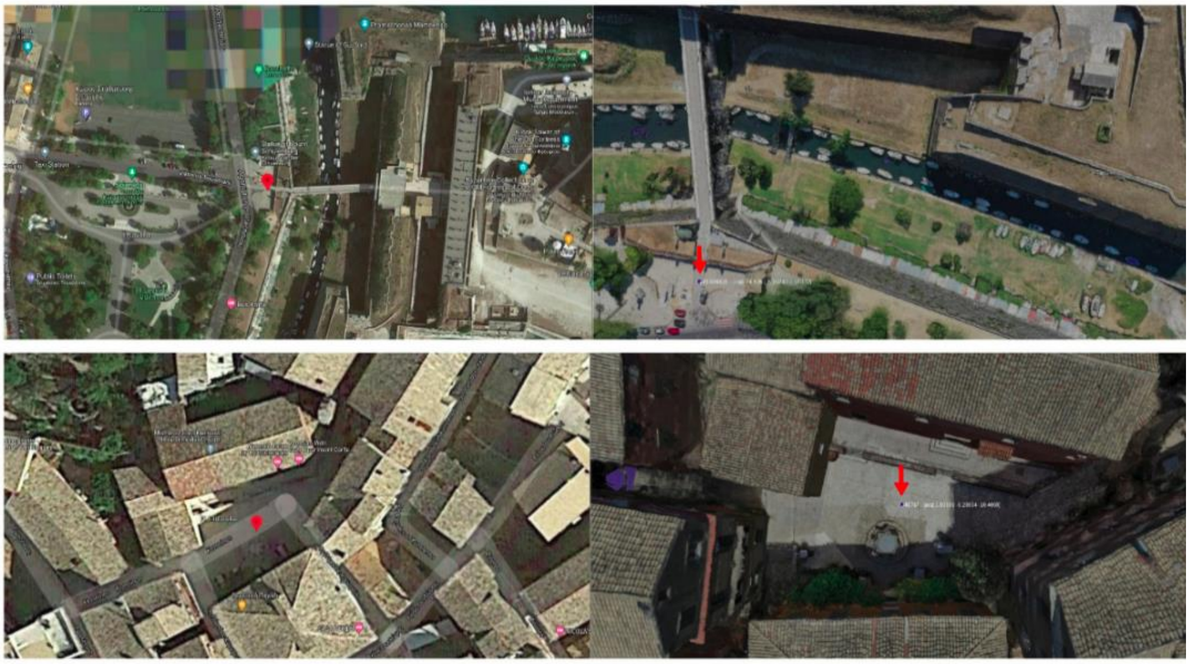

Figure 13: Position in Google Earth and in Meshlab

4.1.2 3D reconstruction examples

The 3D reconstruction service was tested in multiple occasions by the users of xR4DRAMA platform and it was employed to obtain photorealistic 3D mesh models in order to support the two pilot use cases of xR4DRAMA research project.

For the disaster management use case two areas were selected in the city of Vicenza. The first one covers a substantial part of the city centre and the second a suburban neighbourhood. Drone missions were scheduled and performed by xr4DRAMA's partner up2metric between 20 and 22 April 2022 (Figure 14). The drone campaigns were also assisted by members of AAWA and by Luca Fabris from the Civil Protection of Vicenza municipality. The drone imagery was fed in the 3D reconstruction service and 3D models were generated. Examples are showcased in Figure 15 and Figure 16.

Figure 14: Instances of the drone survey missions.

Figure 15: Vicenza city center in Italy. Sample vertical images of city center area captured by UAV (left and middle). Resulting 3D mesh model (right)

Figure 16: Resulting 3D mesh model of Vicenza city center in Italy

For the needs of PUC2 five sites in the island of Corfu were selected as possible locations for the media production. The data collection campaign was performed in Corfu by xR4DRAMA's partner up2metric between 11 and 13 July 2021. The acquired data included vertical and oblique drone images, drone videos, images with a handheld camera, 360 panoramic images and GPS measurements to evaluate the geolocalization of the final 3D models. The images below display the results of the automatic reconstruction process for the Spianada area (Figure 17, Figure 18) and the Venetian fortress in the city of Corfu (Figure 19).

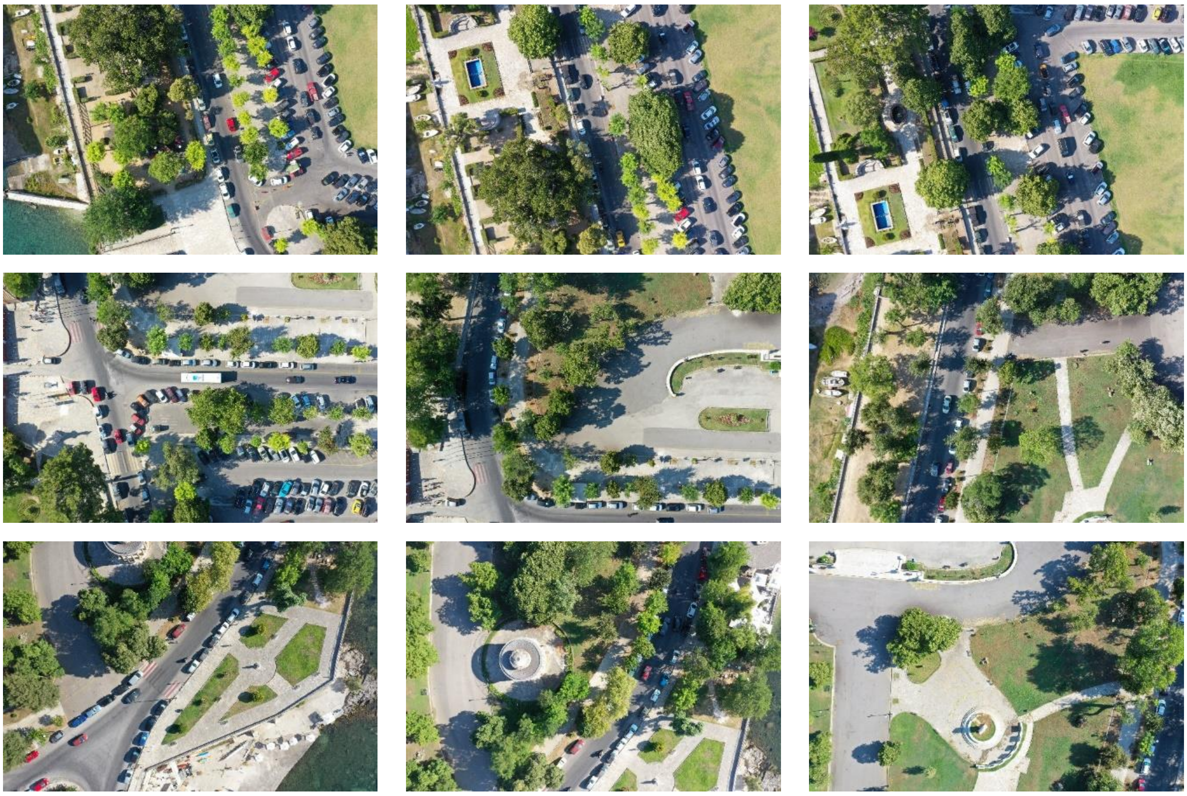

Figure 17: Sample vertical images of Spianada area.

Figure 18: 3D mesh model of the Spianada area

Figure 19: Venetian fortress in the city of Corfu

Another set of images, captured by a mobile phone camera and the generated 3D mesh model are also presented below (Figure 20).

Figure 20: War memorial Stralau in Berlin-Friedrichshain. Sample images (left and middle). Resulting 3D mesh model (right)

4.2 Satellite Service

Besides image-based 3D reconstruction, the space modelling component of the xR4DRAMA platform provides the possibility to generate 3D photorealistic mesh models of large areas from Satellite data. This functionality is implemented as a module of the Remote Sensing service.

Figure 21 shows the position of the 3D reconstruction service from Satellite data inside the GeoService architecture which also includes the GIS Service and the Satellite Service. The Satellite Service is a standalone application that downloads and provides satellite data (satellite images and digital elevation models - DEMs) from the Sentinel Hub.

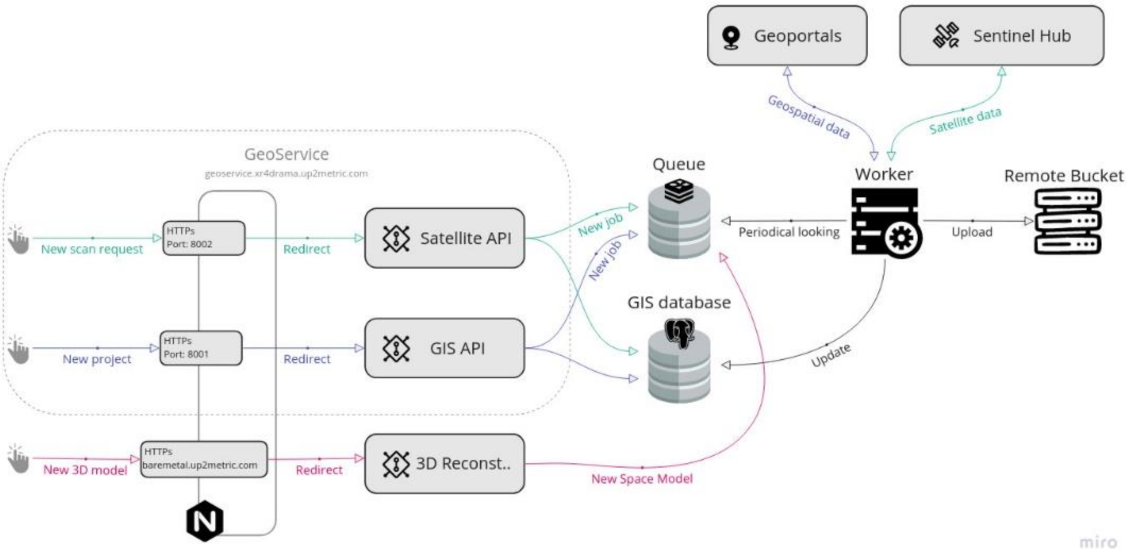

Figure 21: xR4Drama Services

It is documented in the following Swagger link (Figure 22): https://geoservice.xr4drama.up2metric.com:8002/swagger#/.

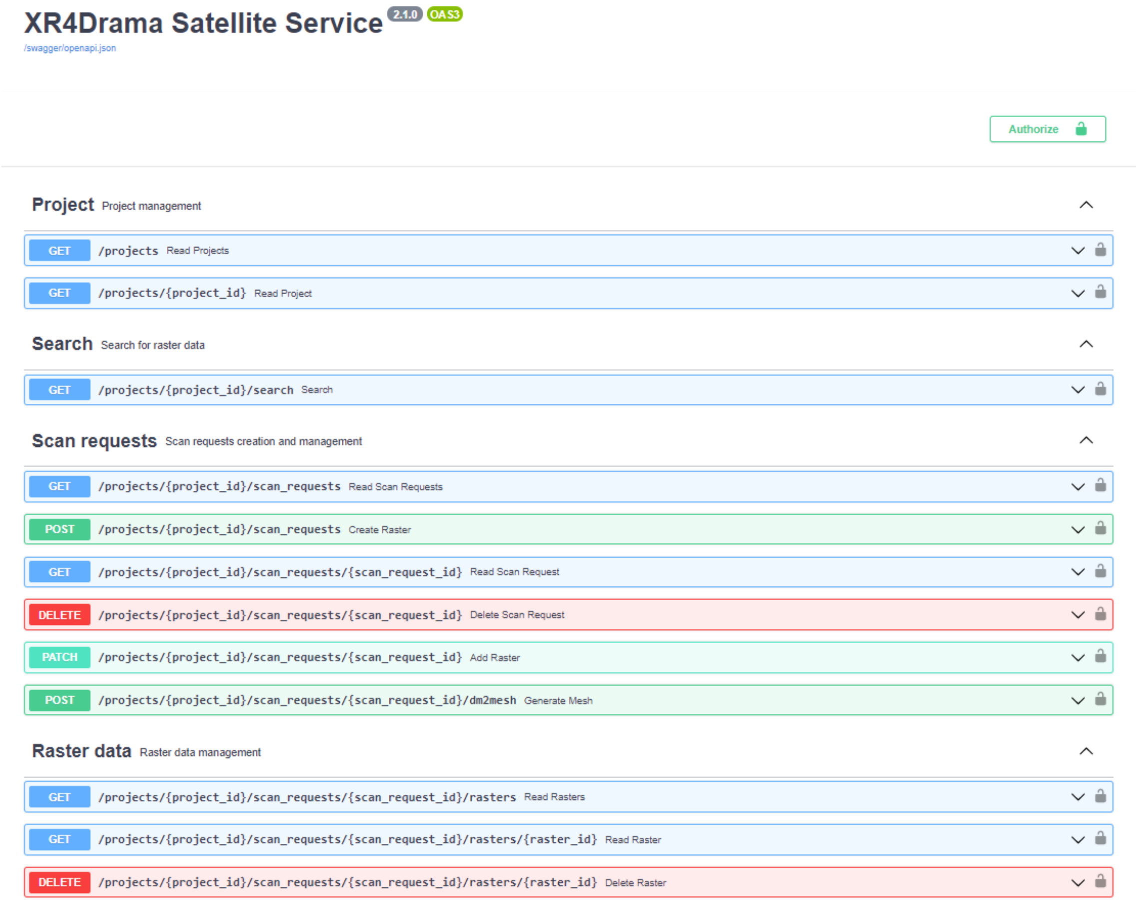

Figure 22: xR4DRAMA Satellite Service Swagger

The Satellite service provides data that can be valuable in disaster management scenarios. The authoring tool provides the ability to visualize these data. Specifically, when a user wishes to view a satellite image, the authoring tool communicates with the backend API, which obtains the data from the satellite service (Figure 23).

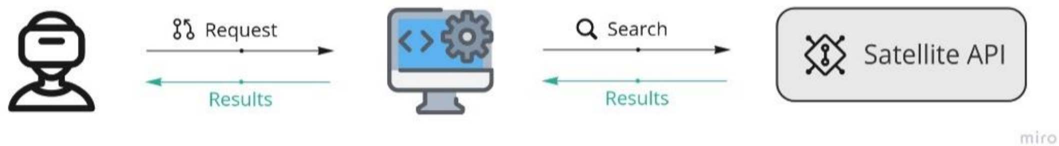

Figure 23: Satellite True-Color and Multispectral images

4.2.1 Implementation of Satellite Service

The xR4DRAMA Satellite service is implemented as an API. It provides Earth Observation imagery and 3D information data via the Sentinel Hub web service. The data supported by the current implementation are (Figure 24):

→ Multispectral or true-color (visible spectral zone) satellite images (10m resolution)

→ Digital Elevation Model (15m resolution)

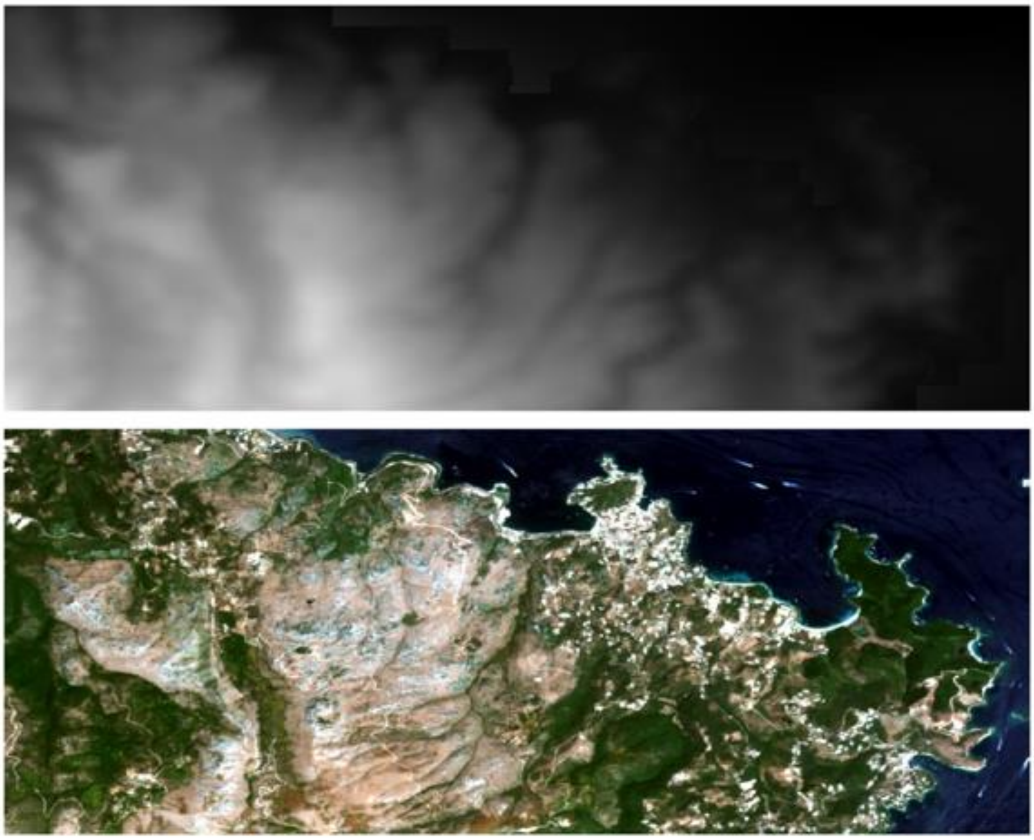

Figure 24: DEM & true-color image (Corfu)

Satellite data is organized into datasets called scan requests. This dataset can contain multiple rasters for different time reversals containing the same regions. Grouping the data into data sets makes their search faster and greatly facilitates the selection of the image and the corresponding DEM during the 3D reconstruction. The user of the Satellite service can send a scan request using an HTTP POST request. A request consists of the bounding box for the area of interest, the type of data (DEM, True-Color, or Multispectral), the satellite type (e.g., Sentinel), the time interval for the search, and timestamps (optional). The service works asynchronously to download the data in response to a scan request. Therefore, the client should check periodically to see if the job status has been completed. The following are the essential functions that a client of the service can perform:

→ Download satellite data

→ Search for existing satellite data

→ Create a 3D model

The Satellite service is responsible for downloading information satellite data within the project's boundaries. The information acquisition process aims to facilitate civil protection in hazardous weather conditions (Disaster Management use case of AAWA, PUC1) of the xR4DRAMA platform. Specifically, the user can download satellite data by specifying a bounding box for the area of interest, a time interval (start and end), and a list of raster types (True-Color or/and Multispectral image and Digital Elevation Model). Moreover, satellite data can be downloaded for multiple timestamps by defining the parameter. Essentially, satellite data are collected by defining the following parameters:

→ Bounding box
→ Timestamp
→ Time interval (start and end)
→ Raster types

By defining the following parameters, the user can search for available data:

→ The bounding box of the area of interest
→ Raster type (optional)
→ Time interval (optional)

After filtering all available satellite data, the satellite service will then return a list of results satisfying each request's requirements. Lastly, the satellite data is returned as a public link, which links to the storage files with an expiration date (Figure 25).

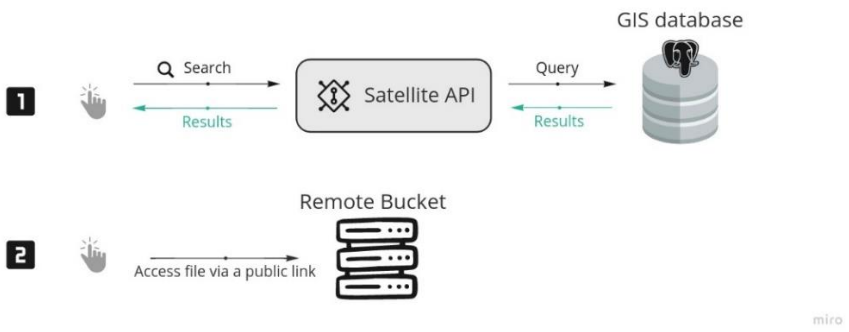

Figure 25: Searching for available satellite data

The Satellite service also offers the possibility of automatically reconstructing a 3D model based on satellite data. To generate their 3D model, users can select a satellite image (True-Color or Multispectral) and DEM from available datasets. Following is a brief explanation of how the 3D model is created.

Multispectral to RGB image conversion

In the first instance, the image is converted to RGB (e.g., True-Color) if it is Multispectral (Figure 26). After that, to improve the image's contrast, each band is equalized using the histogram equalization technique.

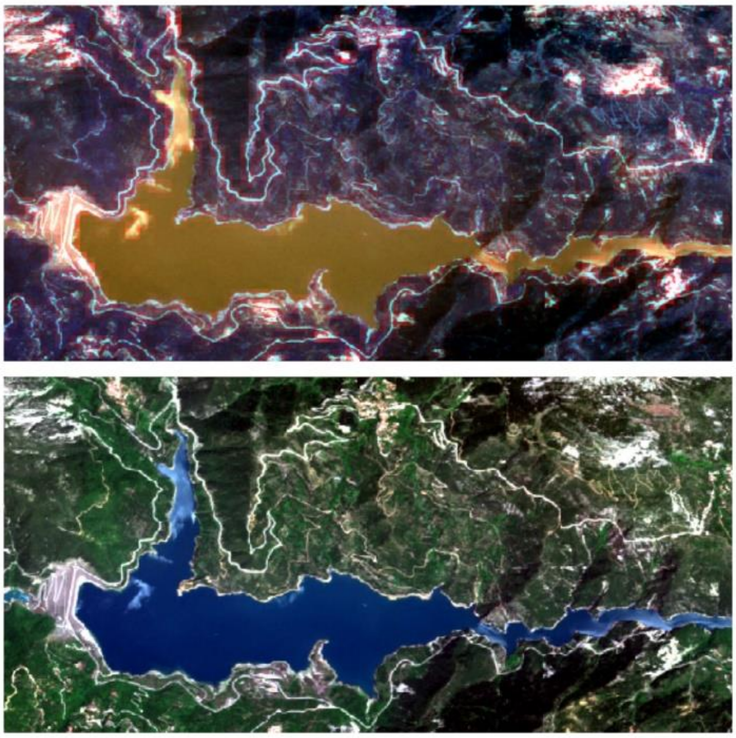

Figure 26: Satellite Multispectral and True-Color images

DEM resizing

The next step is to reduce the resolution of the DEM, which will result in fewer vertices and, thus, fewer faces (Figure 27). Alternatively, the RGB image resolution may remain unchanged, giving the model a realistic appearance.

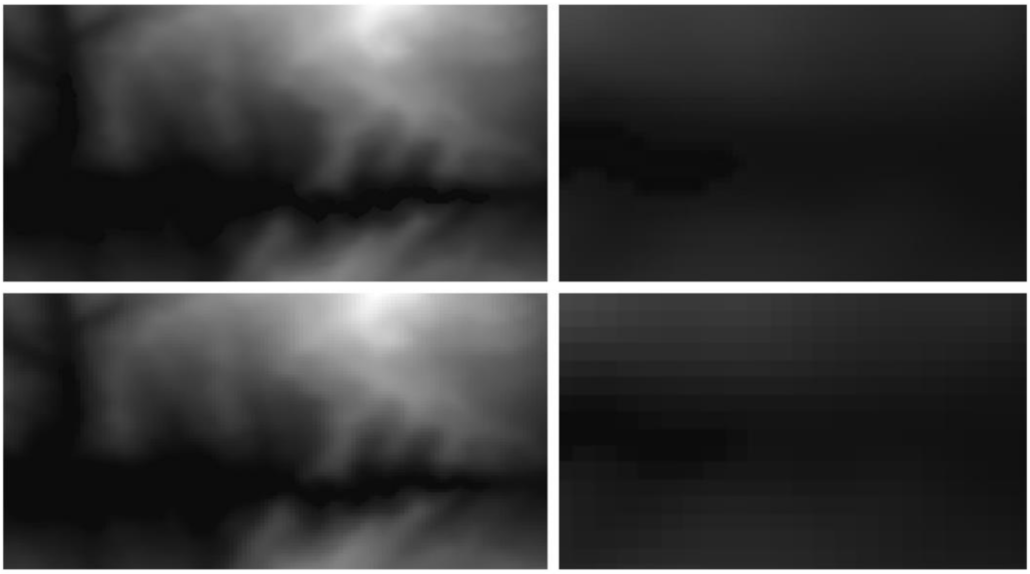

Figure 27: DEM resizing

Altitude correction

There is a possibility that the DEM contains outliers, i.e., values that are negative, or erroneously differ significantly from their neighboring values. In this case, the altitude value is replaced with the average of the neighboring altitudes.

Coordinate preparation

It is necessary to use a 3D cartesian coordinate system to display the generated model in any 3D model editing software. Therefore, grid coordinates are converted from WGS'84 or EPSG: 4326 (latitude, longitude, altitude) to WGS'84/World Mercator or EPSG: 3395 (easting, northing, altitude).

3D Model Creation

Delaunay triangulation is performed in the following steps, and the triangulated model is then textured using the RGB image. Lastly, the model is simplified to reduce the large number of redundant faces (Figure 28).

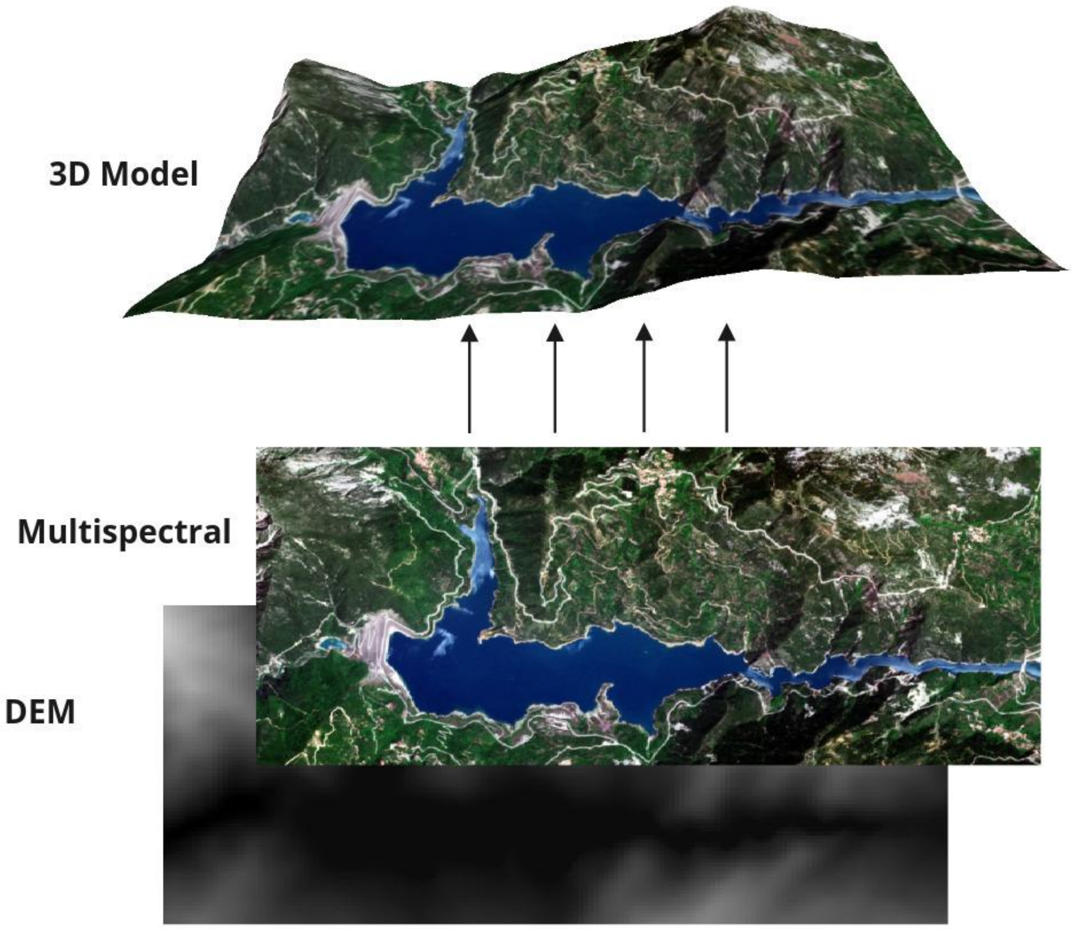

Figure 28: Satellite True-Color and Multispectral images

4.3 Internal components

The 3D-Reconstruction and the Satellite service contains the following internal components:

- The 3D-Reconstruction Service API, which was created by Flask and handles the 3D models created from the inputs provided by the user.
    - Flask “*is a widely used micro web framework for creating APIs in Python. It is a simple yet powerful web framework which is designed to get started quick and easy, with the ability to scale up to complex applications”*[11].
    - SQLite “*is a C-language library that implements a small, fast, self-contained, high-reliability, full-featured, SQL database engine. SQLite is the most used database engine in the world. SQLite is built into all mobile phones and most computers and comes bundled inside countless other applications that people use every day”*[12].
- The Satellite Service API, which is a FastAPI application that handles the scan requests and is wrapped by a uvicorn webserver in order to enable SSL.
    - FastAPI “*is a modern, fast (high-performance), web framework for building APIs with Python 3.6+ based on standard Python type hints*”[13].
    - Uvicorn *“is a lightning-fast ASGI server implementation, using uvloop and httptools”* [14].
    - PostGIS database for storing the stateful scan requests and the created data. PostGIS[15] is a spatial database extender for PostgreSQL object-relational database to perform CRUD (create, read, update, and delete) operations on GIS data. It adds support for geographic objects storing them as vectors, e.g., for geometry information, and as raster, and allowing location queries (spatial functions) to be run in SQL8. The spatial data types are points, lines (2 points), multi-lines (more than 2 points), and polygons (where the start and the end points are the same).
- NGINX “*is open-source software for web serving, reverse proxying, caching, load balancing, media streaming, and more”*[16].
- Redis as a message broker and for queuing.
    - Redis *“is an open source (BSD licensed), in-memory data structure store, used as a database, cache, and message broker”*[17].

All those components are “dockerised” and the structure of the service can be easily deployed using a docker-compose file.

[11] https://flask.palletsprojects.com/
[12] https://www.sqlite.org/
[13] https://fastapi.tiangolo.com/
[14] https://www.uvicorn.org/
[15] https://postgis.net/
[16] https://www.nginx.com/
[17] https://redis.io/

5 CONCLUSION

In this document the space modelling service for outdoors environments of the xR4DRAMA for public spaces was presented which includes the work performed in task T4.4. (3D reconstruction of the area). The relations with user requirements were also presented and were considered for the definition of the appropriate methodologies, and techniques for generating 3D models that will serve the foreseen objectives.

Details were given regarding the implementation and the use of the two main components. The first was about the development of an online 3D reconstruction service from drone images or images suitably taken by a handheld camera, or even videos, based on state-of-the-art 3D vision and photogrammetric techniques. This service can be used by any user who has an API key. If the metadata of images contains GPS data, the final model has the scale, rotation, and position it has in the real world. The second was about the development of the remote sensing 3D reconstruction service and in particular of the satellite 3D component, which provides 3D models of large areas, from available satellite images and DEMs within the project's boundaries. Again, these actions require an API key.

Feedback from the exploitation of the developed services from the relevant tasks of the xR4DRAMA project will be taken into account and necessary corrections and updates are going to be performed if needed.

REFERENCES

Agarwal, S., & Mierle, K. 2012. "*Ceres solver: Tutorial & reference.*" Google Inc, 2(72), pp. 8.

Agarwal, S., Furukawa, Y., Snavely, N., Simon, I., Curless, B., Seitz, S. M., & Szeliski, R. 2011. *"Building Rome in a day"*, Communications of the ACM, 54(10), pp. 105–112. https://doi.org/10.1145/2001269.2001293

Alcantarilla, P. F., Bartoli, A., & Davison, A. J. 2012. *"KAZE Features"*, Computer Vision – ECCV 2012, pp. 214–227. ZX https://doi.org/10.1007/978-3-642-33783-3_16

Alcantarilla, P., Nuevo, J., & Bartoli, A. 2013. *"Fast Explicit Diffusion for Accelerated Features in Nonlinear Scale Spaces"*, Proceedings of the British Machine Vision Conference 2013. https://doi.org/10.5244/c.27.13

Ann, N. Q., Achmad, M. H., Bayuaji, L., Daud, M. R., & Pebrianti, D. 2016. *"Study on 3D scene reconstruction in robot navigation using stereo vision"*, 2016 IEEE International Conference on Automatic Control and Intelligent Systems (I2CACIS). https://doi.org/10.1109/i2cacis.2016.7885292

Barath, D., Mishkin, D., Eichhardt, I., Shipachev, I., & Matas, J. 2021. "*Efficient Initial Pose-graph Generation for Global SfM*", 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). https://doi.org/10.1109/cvpr46437.2021.01431

Bay, H., Ess, A., Tuytelaars, T., & van Gool, L. 2008. "*Speeded-Up Robust Features (SURF)*", Computer Vision and Image Understanding, 110(3), pp. 346–359. https://doi.org/10.1016/j.cviu.2007.09.014

Becker, R. E., Galayda, L. J., & MacLaughlin, M. M. 2018. *"Digital photogrammetry software comparison for rock mass characterization"*, In 52nd US Rock Mechanics/Geomechanics Symposium. OnePetro.

Bhattacharjee, S., & Chaudhuri, P. 2020. *"A Survey on Sketch Based Content Creation: from the Desktop to Virtual and Augmented Reality"*, Computer Graphics Forum, 39(2), pp. 757–780. https://doi.org/10.1111/cgf.14024

Bhotika, R., Fleet, D. J., & Kutulakos, K. N. 2002. *"A Probabilistic Theory of Occupancy and Emptiness"*, Computer Vision — ECCV 2002, pp. 112–130. https://doi.org/10.1007/3-540-47977-5_8

Bhowmick, B., Patra, S., Chatterjee, A., Govindu, V. M., & Banerjee, S. 2015. *"Divide and Conquer: Efficient Large-Scale Structure from Motion Using Graph Partitioning"*, Computer Vision -- ACCV 2014, pp. 273–287. https://doi.org/10.1007/978-3-319-16808-1_19

Bianco, S., Ciocca, G., & Marelli, D. 2018. *"Evaluating the Performance of Structure from Motion Pipelines"*, Journal of Imaging, 4(8), pp. 98. https://doi.org/10.3390/jimaging4080098

Bruno, F., Bruno, S., de Sensi, G., Luchi, M. L., Mancuso, S., & Muzzupappa, M. 2010. *"From 3D reconstruction to virtual reality: A complete method for digital archaeological exhibition"*, Journal of Cultural Heritage, 11(1), pp. 42–49. https://doi.org/10.1016/j.culher.2009.02.006

Bujnak, M., Kukelova, Z., & Pajdla, T. 2008. *"A general solution to the P4P problem for camera with unknown focal length"*, 2008 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1–8. https://doi.org/10.1109/cvpr.2008.4587793

Burns, J. H. R., & Delparte, D. 2017. *"Comparison Of Commercial Structure-from-motion Photogrammety Software Used For Underwater Three-dimensional Modeling Of Coral Reef Environments"*, The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XLII–2/W3, pp. 127–131. https://doi.org/10.5194/isprs-archives-xlii-2-w3-127-2017

Chen, Y., Chan, A. B., Lin, Z., Suzuki, K., & Wang, G. 2017. *"Efficient tree-structured SfM by RANSAC generalized Procrustes analysis"*, Computer Vision and Image Understanding, 157, pp. 179–189. https://doi.org/10.1016/j.cviu.2017.02.005

Cheng, J., Leng, C., Wu, J., Cui, H., & Lu, H. 2014. *"Fast and Accurate Image Matching with Cascade Hashing for 3D Reconstruction"*, 2014 IEEE Conference on Computer Vision and Pattern Recognition, pp. 1-8. https://doi.org/10.1109/cvpr.2014.8

Chowdhury, S. A. H., Nguyen, C., Li, H., & Hartley, R. 2021. *"Fixed-Lens camera setup and calibrated image registration for multifocus multiview 3D reconstruction"*, Neural Computing and Applications, 33(13), pp. 7421–7440. https://doi.org/10.1007/s00521-021-05926-7

Chum, O., & Matas, J. 2005. *"Matching with PROSAC — Progressive Sample Consensus",* 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), 1, pp. 220–226. https://doi.org/10.1109/cvpr.2005.221

Chum, O., Matas, J., & Kittler, J. 2003. *"Locally Optimized RANSAC",* Lecture Notes in Computer Science, pp. 236–243. https://doi.org/10.1007/978-3-540-45243-0_31

Cignoni, P., Montani, C., & Scopigno, R. 1998. *"DeWall: A fast divide and conquer Delaunay triangulation algorithm"*, in Ed. Computer-Aided Design, 30(5), pp. 333–341. https://doi.org/10.1016/s0010-4485(97)00082-1

Collins, T., Pizarro, D., Gasparini, S., Bourdel, N., Chauvet, P., Canis, M., Calvet, L., & Bartoli, A. 2021. *"Augmented Reality Guided Laparoscopic Surgery of the Uterus",* IEEE Transactions on Medical Imaging, 40(1), pp. 371–380. https://doi.org/10.1109/tmi.2020.3027442

Cui, H., Shen, S., & Hu, Z. 2016. *"Robust global translation averaging with feature tracks",* 2016 23rd International Conference on Pattern Recognition (ICPR). https://doi.org/10.1109/icpr.2016.7900214

Cui, Z., & Tan, P. 2015. *"Global structure-from-motion by similarity averaging"*, 2015 IEEE International Conference on Computer Vision (ICCV), pp. 864–872. https://doi.org/10.1109/iccv.2015.105

Dehais, J., Anthimopoulos, M., Shevchik, S., & Mougiakakou, S. 2017. *"Two-View 3D Reconstruction for Food Volume Estimation"*, IEEE Transactions on Multimedia, 19(5), pp. 1090–1099. https://doi.org/10.1109/tmm.2016.2642792

Devillers, O. 1998. *"Improved incremental randomized Delaunay triangulation*", Proceedings of the Fourteenth Annual Symposium on Computational Geometry - SCG '98. https://doi.org/10.1145/276884.276896

Ðuric, I., Vasiljevic, I., Obradovic, M., Stojakovic, V., Kicanovic, J., & Obradovic, R. 2021. *"Comparative Analysis of Open-Source and Commercial Photogrammetry Software for Cultural Heritage"*, Proceedings of the 39th International Conference on Education and Research in Computer Aided Architectural Design in Europe (ECAADe). https://doi.org/10.52842/conf.ecaade.2021.2.243

e Sá, A. M., Vila, A. B. I., Rodriguez-Echavarria, K., Marroquim, R., & Fonseca, V. L. 2019. *"Accessible Digitisation and Visualisation of Open Cultural Heritage Assets",* In GCH, pp. 59-68. https://doi.org/10.2312/gch.20191349

El Saer, A., Stentoumis, C., Kalisperakis, I., & Nomikou, P. 2020. *"Developing a strategy for precise 3d modelling of large-scale scenes for VR"*, The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XLIII-B4-2-B4, pp. 567–574. https://doi.org/10.5194/isprs-archives-xliii-b4-2020-567-2020

El Saer, A., Stentoumis, C., Kalisperakis, I., Grammatikopoulos, L., Nomikou, P., & Vlasopoulos, O. 2020. *"3D RECONSTRUCTION AND MESH OPTIMIZATION OF UNDERWATER SPACES FOR VIRTUAL REALITY"*, The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XLIII-B2-2-B2, pp. 949–956. https://doi.org/10.5194/isprs-archives-xliii-b2-2020-949-2020

Faugeras, O., Bras-Mehlman, E., & Boissonnat, J. 1990. *"Representing stereo data with the Delaunay triangulation"*, Artificial Intelligence, 44(1–2), pp. 41–87. https://doi.org/10.1016/0004-3702(90)90098-k

Ferdani, D., Fanini, B., Piccioli, M. C., Carboni, F., & Vigliarolo, P. 2020. *"3D reconstruction and validation of historical background for immersive VR applications and games: The case study of the Forum of Augustus in Rome"*, Journal of Cultural Heritage, 43, pp. 129–143. https://doi.org/10.1016/j.culher.2019.12.004

Fischler, M. A., & Bolles, R. C. 1981. *"Random sample consensus",* Communications of the ACM, 24(6), pp. 381–395. https://doi.org/10.1145/358669.358692

Fragoso, V., Sen, P., Rodriguez, S., & Turk, M. 2013. *"EVSAC: Accelerating Hypotheses Generation by Modeling Matching Scores with Extreme Value Theory",* Proceedings of the IEEE International Conference on Computer Vision (ICCV), pp. 2472–2479.

Furukawa Y., Ponce J. 2006. "*High-fidelity image-based modelling*", Technical Report CVR 2006-02, University of Illinois at Urbana-Champaign.

Gabara, G., & Sawicki, P. 2018. *"A New Approach for Inspection of Selected Geometric Parameters of a Railway Track Using Image-Based Point Clouds",* Sensors, 18(3), pp. 791. https://doi.org/10.3390/s18030791

Gargallo, P., & Sturm, P. 2005. *"Bayesian 3D Modeling from Images Using Multiple Depth Maps"*, 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), 2, pp. 885–891. https://doi.org/10.1109/cvpr.2005.84

Georgopoulos, A., Oikonomou, C., Adamopoulos, E., & Stathopoulou, E. K. 2016. *"Evaluating unmanned aerial platforms for cultural heritage large scale mapping*", ISPRS - International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XLI–B5, pp. 355–362. https://doi.org/10.5194/isprsarchives-xli-b5-355-2016

Griwodz, C., Gasparini, S., Calvet, L., Gurdjos, P., Castan, F., Maujean, B., ... & Lanthony, Y. (2021). AliceVision Meshroom: An open-source 3D reconstruction pipeline. *In Proceedings of the 12th ACM Multimedia Systems Conference*, pp. 241-247

Grover, S., Sidana, K., & Jain, V. 2021. *"Pipeline for 3D reconstruction of the human body from AR/VR headset mounted egocentric cameras"*, arXiv preprint arXiv:2111.05409. https://doi.org/10.48550/arXiv.2111.05409

Ham, H., Wesley, J., & Hendra, H. 2019. *"Computer Vision Based 3D Reconstruction: A Review"*, International Journal of Electrical and Computer Engineering (IJECE), 9(4), pp. 2394. https://doi.org/10.11591/ijece.v9i4.pp2394-2402

Hartley, R., & Zisserman, A. 2004. "*Multiple View Geometry in Computer Vision*", Cambridge University Press.

Hesch, J. A., & Roumeliotis, S. I. 2011. *"A Direct Least-Squares (DLS) method for PnP",* 2011 International Conference on Computer Vision, pp. 383–390. https://doi.org/10.1109/iccv.2011.6126266

Hsieh, M. C., & Lee, J. J. 2018. *"Preliminary Study of VR and AR Applications in Medical and Healthcare Education",* Journal of Nursing and Health Studies, 03(1). https://doi.org/10.21767/2574-2825.100030

Iglhaut, J., Cabo, C., Puliti, S., Piermattei, L., O'Connor, J., & Rosette, J. 2019. *"Structure from Motion Photogrammetry in Forestry: a Review"*, Current Forestry Reports, 5(3), pp. 155–168. https://doi.org/10.1007/s40725-019-00094-3

Inzerillo, L., di Mino, G., & Roberts, R. 2018. *"Image-based 3D reconstruction using traditional and UAV datasets for analysis of road pavement distress",* Automation in Construction, 96, pp. 457–469. https://doi.org/10.1016/j.autcon.2018.10.010

James, M.R., Chandler, J.H., Eltner, A., Fraser, C., Miller, P.E., Mills, J.P., Noble, T., Robson, S. and Lane, S.N., 2019. *"Guidelines on the use of structure-from-motion photogrammetry in geomorphic research",* Earth Surface Processes and Landforms, 44(10), pp.2081–2084. https://doi.org/10.1002/esp.4637

Jancosek, Michal, and Tomas Pajdla 2011. *"Multi-view reconstruction preserving weakly-supported surfaces"*, Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 3121-3128.

Jiang, S., Jiang, W., Guo, B., Li, L., & Wang, L. 2021. *"Learned Local Features for Structure from Motion of UAV Images: A Comparative Evaluation"*, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing, 14, pp. 10583–10597. https://doi.org/10.1109/jstars.2021.3119990

Kalisperakis, I., Kopsiaftis, G., Rallis, I., Stentoumis, C., Kouka, A., Riga, V., & Koutsomitsos, D. 2022. *"Use of Photogrammetry in a Business Simulation Game",* The15th International Conference on PErvasive Technologies Related to Assistive Environments, pp. 636–640. https://doi.org/10.1145/3529190.3534767

Kingsland, K. 2020. *"Comparative analysis of digital photogrammetry software for cultural heritage"*, Digital Applications in Archaeology and Cultural Heritage, 18, e00157. https://doi.org/10.1016/j.daach.2020.e00157

Kowdle, A., Chang, Y., Gallagher, A., & Chen, T. 2011. *"Active learning for piecewise planar multiview stereo"*, In CVPR. https://doi.org/10.1109/CVPR.2011.5995638

Kukelova, Z., Bujnak, M., & Pajdla, T. 2013. *"Real-Time Solution to the Absolute Pose Problem with Unknown Radial Distortion and Focal Length",* 2013 IEEE International Conference on Computer Vision. https://doi.org/10.1109/iccv.2013.350

Kutulakos, K. N. 2000. *"Approximate N-View Stereo",* Computer Vision - ECCV 2000, pp. 67–83. https://doi.org/10.1007/3-540-45054-8_5

Kutulakos, K.N., Seitz, S.M. 2000. *"A Theory of Shape by Space Carving",* International Journal of Computer Vision 38, pp. 199–218. https://doi.org/10.1023/A:1008191222954

Lallensack, J., Buchwitz, M., & Romilio, A. 2020. *"Photogrammetry in ichnology: 3D model generation, visualisation, and data extraction"*, *Journal of Paleontological Techniques*. https://doi.org/10.31223/x5j30d

Lee, D. T., & Schachter, B. J. 1980. *"Two algorithms for constructing a Delaunay triangulation"*, International Journal of Computer & Information Sciences, 9(3), pp. 219–242. https://doi.org/10.1007/bf00977785

Lepetit, V., Moreno-Noguer, F., & Fua, P. 2008. *"EPnP: An Accurate O(n) Solution to the PnP Problem"*, International Journal of Computer Vision, 81(2), pp. 155–166. https://doi.org/10.1007/s11263-008-0152-6

Leutenegger, S., Chli, M., & Siegwart, R. Y. 2011. *"BRISK: Binary Robust invariant scalable keypoints"*, 2011 International Conference on Computer Vision, pp. 2548–2555. https://doi.org/10.1109/iccv.2011.6126542

Li, J., Miao, Z., Liu, X., & Wan, Y. 2010. *"3D reconstruction based on stereovision and texture mapping",* In Proceedings of the ISPRS Technical Commission III Symposium, Photogrammetric Computer Vision and Image Analysis, pp. 1-3.

Li, X., Xiong, B., Yuan, Z., He, K., Liu, X., Liu, Z., & Shen, Z. 2021. *"Evaluating the Potentiality of Using Control-free Images from a Mini Unmanned Aerial Vehicle (UAV) and Structure-from-Motion (SfM) Photogrammetry to Measure Paleoseismic Offsets",* International Journal of Remote Sensing, 42(7), pp. 2417–2439. https://doi.org/10.1080/01431161.2020.1862434

Li, Z., Zhang, Z., Luo, S., Cai, Y., & Guo, S. 2022. *"An Improved Matting-SfM Algorithm for 3D Reconstruction of Self-Rotating Objects"*, Mathematics, 10(16), pp. 2892. https://doi.org/10.3390/math10162892

Lowe, D. G. 2004. *"Distinctive Image Features from Scale-Invariant Keypoints",* International Journal of Computer Vision, 60(2), pp. 91–110. https://doi.org/10.1023/b:visi.0000029664.99615.94

Luhmann, T., Chizhova, M., Gorkovchuk, D., Hastedt, H., Chachava, N., & Lekveishvili, N. 2019. *"Combination of terrestrial laserscanning, UAV and close-range photogrammetry for 3D reconstruction of complex churches in Georgia",* The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XLII–2/W11, pp. 753–761. https://doi.org/10.5194/isprs-archives-xlii-2-w11-753-2019

Ma, X., Wang, Z., Li, H., Zhang, P., Ouyang, W., & Fan, X. 2019. *"Accurate Monocular 3D Object Detection via Color-Embedded 3D Reconstruction for Autonomous Driving"*, 2019 IEEE/CVF International Conference on Computer Vision (ICCV), pp. 6851–6860. https://doi.org/10.1109/iccv.2019.00695

Manessis, A., Hilton, A., Palmer, P., McLauchlan, P., & Shen, X. 2000. *"Reconstruction of scene models from sparse 3D structure"*, Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No.PR00662), pp. 666–673. https://doi.org/10.1109/iccv.2003.1238338

Manni, A., Oriti, D., Sanna, A., de Pace, F., & Manuri, F. 2021. *"Snap2cad: 3D indoor environment reconstruction for AR/VR applications using a smartphone device"*, Computers & Graphics, 100, pp. 116–124. https://doi.org/10.1016/j.cag.2021.07.014

Milàn, J., Falkingham, P. L., & Mueller-Töwe, I. J. 2020. *"Small ornithopod dinosaur tracks and crocodilian remains from the Middle Jurassic Bagå Formation, Bornholm, Denmark: Important additions to the rare Middle Jurassic vertebrate faunas of Northern Europe"*, Bulletin of the Geological Society of Denmark, 68, pp. 245–253. https://doi.org/10.37570/bgsd-2020-68-11

Moisan, L., Moulon, P., & Monasse, P. 2012. *"Automatic Homographic Registration of a Pair of Images, with A Contrario Elimination of Outliers",* Image Processing on Line, 2, pp. 56–73. https://doi.org/10.5201/ipol.2012.mmm-oh

Moon, D., Lee, K., Ko, H., Kwon, S., Lee, S., & Song, J. 2021. *"Analysis of accuracy factor and pre-processing method of image compensation for 3D reconstruction using 2D image obtained* *from unmanned aerial vehicle (UAV)”,* Journal of Asian Architecture and Building Engineering, pp. 1–14. https://doi.org/10.1080/13467581.2021.1971679

Moulon, Pierre, Pascal Monasse, and Renaud Marlet. “Adaptive Structure from Motion with a contrario model estimation 1 Introduction 2 Structure from Motion the classical pipeline.” Lect. Notes Comput. Sci. (including Subser. Lect. Notes Artif. Intell. Lect. Notes Bioinformatics) 7727 LNCS, no. PART 4 (2012): 1-14.

Muja, M., & Lowe, D. G. 2009. *“Fast approximate nearest neighbors with automatic algorithm configuration”*, VISAPP (1), 2, pp. 331–340.

Narayanan, P., Rander, P., & Kanade, T. 1998. *“Constructing virtual worlds using dense stereo”*, Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271), pp. 3–10. https://doi.org/10.1109/iccv.1998.710694

Navarro, F., Fdez, J., Garzón, M., Roldán, J. J., & Barrientos, A. 2017. *“Integrating 3D Reconstruction and Virtual Reality: A New Approach for Immersive Teleoperation”*, ROBOT 2017: Third Iberian Robotics Conference, pp. 606–616. https://doi.org/10.1007/978-3-319-70836-2_50

Nayyar, A., Mahapatra, B., Nhuong Le, D., & Suseendran, G. 2018. *“Virtual Reality (VR) & Augmented Reality (AR) technologies for tourism and hospitality industry”,* International Journal of Engineering & Technology, 7(2.21), pp. 156. https://doi.org/10.14419/ijet.v7i2.21.11858

Nister, D. 2004. “An efficient solution to the five-point relative pose problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 26*(6), pp. 756–770. https://doi.org/10.1109/TPAMI.2004.17

Pollefeys, M., van Gool, L., Vergauwen, M., Verbiest, F., Cornelis, K., Tops, J., & Koch, R. 2004. *“Visual Modeling with a Hand-Held Camera”,* International Journal of Computer Vision, 59(3), pp. 207–232. https://doi.org/10.1023/b:visi.0000025798.50602.3a

Raguram, R., Frahm, J. M., & Pollefeys, M. 2008. *“A Comparative Analysis of RANSAC Techniques Leading to Adaptive Real-Time Random Sample Consensus”*, Lecture Notes in Computer Science, pp. 500–513. https://doi.org/10.1007/978-3-540-88688-4_37

Rahaman, H., & Champion, E. (2019). To 3D or Not 3D: Choosing a Photogrammetry Workflow for Cultural Heritage Groups. *Heritage, 2*(3), pp. 1835–1851. https://doi.org/10.3390/heritage2030112

Remondino, F., Nocerino, E., Toschi, I., & Menna, F. 2017. *“A Critical Review of automated Photogrammetric Processing of large datasets”,* The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XLII-2/W5, pp. 591–599. https://doi.org/10.5194/isprs-archives-xlii-2-w5-591-2017

Rhee, H. L., & Lee, K. H. 2021. *"Enhancing the Sneakers Shopping Experience through Virtual Fitting Using Augmented Reality",* Sustainability, 13(11), pp. 6336. https://doi.org/10.3390/su13116336

Rickard, W. D. A., Ramos Coelho, J. F., Hollick, J., Soon, S., & Woods, A. 2020. *"Application of Photogrammetric 3D Reconstruction to Scanning Electron Microscopy: Considerations for Volume Analysis",* Journal of Imaging Science and Technology, 64(6), pp. 60404–1. https://doi.org/10.2352/j.imagingsci.technol.2020.64.6.060404

Roh, T. H., Oh, J. W., Jang, C. K., Choi, S., Kim, E. H., Hong, C. K., & Kim, S. H. 2021. *"Virtual dissection of the real brain: integration of photographic 3D models into virtual reality and its effect on neurosurgical resident education",* Neurosurgical Focus, 51(2), E16. https://doi.org/10.3171/2021.5.focus21193

Rublee, E., Rabaud, V., Konolige, K., & Bradski, G. 2011. *"ORB: An efficient alternative to SIFT or SURF"*, 2011 International Conference on Computer Vision, pp. 2564–2571. https://doi.org/10.1109/iccv.2011.6126544

Seitz, S. M., & Dyer, C. R. 1999. *"Photorealistic Scene Reconstruction by Voxel Coloring"*, International Journal of Computer Vision, 35(2), pp. 151–173. https://doi.org/10.1023/a:1008176507526

Seitz, S., Curless, B., Diebel, J., Scharstein, D., & Szeliski, R. 2006. *"A Comparison and Evaluation of Multi-View Stereo Reconstruction Algorithms"*, 2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition - Volume 1 (CVPR'06). https://doi.org/10.1109/cvpr.2006.19

Sinha, S., & Pollefeys, M. 2005. *"Multi-view reconstruction using photo-consistency and exact silhouette constraints: a maximum-flow formulation",* Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1, pp. 349–356. https://doi.org/10.1109/iccv.2005.159

Smith, E., Meger, D., Pineda, L., Calandra, R., Malik, J., Romero Soriano, A., & Drozdzal, M. 2021. *"Active 3D shape reconstruction from vision and touch",* Advances in Neural Information Processing Systems, 34, pp. 16064-16078. https://doi.org/10.48550/arXiv.2007.03778

Snavely, N., Seitz, S. M., & Szeliski, R. 2006. "Photo tourism", ACM SIGGRAPH 2006 Papers on - SIGGRAPH '06, 25(3), pp. 835–846. https://doi.org/10.1145/1179352.1141964

Spallone, R., Lamberti, F., Guglielminotti Trivel, M., Ronco, F., & Tamantini, S. 2021. *"3D reconstruction and presentation of cultural heritage: ar and vr experiences at the museo d'arte orientale di torino"*, The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, XLVI–M, pp. 697–704. https://doi.org/10.5194/isprs-archives-xlvi-m-1-2021-697-2021

Stathopoulou, E. K., Welponer, M., & Remondino, F. 2019. *"Open-source image-based 3d reconstruction pipelines: review, comparison and evaluation"*, The International Archives of Photogrammetry, Remote Sensing and Spatial Information Sciences, XLII-2/W17, pp. 331–338. https://doi.org/10.5194/isprs-archives-xlii-2-w17-331-2019

Stentoumis, C. 2018. *“Multiple View Stereovision”*, In A. Bentkowska-Kafel & L. MacDonald (Eds.), Digital Techniques for Documenting and Preserving Cultural Heritage, pp. 141–143. Amsterdam University Press. https://doi.org/10.1515/9781942401353

Stewénius, H., Engels, C., & Nistér, D. 2006. *“Recent developments on direct relative orientation”*, ISPRS Journal of Photogrammetry and Remote Sensing, 60(4), pp. 284–294. https://doi.org/10.1016/j.isprsjprs.2006.03.005

Sweeney, C., Fragoso, V., Hollerer, T., & Turk, M. 2016. *“Large Scale SfM with the Distributed Camera Model”*, 2016 Fourth International Conference on 3D Vision (3DV). https://doi.org/10.1109/3dv.2016.31

Szeliski, R. (1999). A multi-view approach to motion and stereo. Proceedings. *1999 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No PR00149)*, pp. 157–163. https://doi.org/10.1109/cvpr.1999.786933

Tareen, S. A. K., & Saleem, Z. 2018. *“A comparative analysis of SIFT, SURF, KAZE, AKAZE, ORB, and BRISK”*, 2018 International Conference on Computing, Mathematics and Engineering Technologies (ICoMET). https://doi.org/10.1109/icomet.2018.8346440

Taylor, C. J. 2003. *“Surface reconstruction from feature based stereo”*, Proceedings Ninth IEEE International Conference on Computer Vision, pp. 184–190. https://doi.org/10.1109/iccv.2003.1238338

Tefera, Y., Poiesi, F., Morabito, D., Remondino, F., Nocerino, E., & Chippendale, P. 2018. *“3DNOW: image-based 3d reconstruction and modeling via web”*, The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XLII–2, pp. 1097–1103. https://doi.org/10.5194/isprs-archives-xlii-2-1097-2018

Tome, D., Russell, C., & Agapito, L. 2017. *“Lifting from the Deep: Convolutional 3D Pose Estimation from a Single Image”*, 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 2500–2509. https://doi.org/10.1109/cvpr.2017.603

Treuille, A., Hertzmann, A., & Seitz, S. M. 2004. *“Example-Based Stereo with General BRDFs”*, Lecture Notes in Computer Science, pp. 457–469. https://doi.org/10.1007/978-3-540-24671-8_36

Vergauwen, M., & van Gool, L. 2006. *“Web-based 3D Reconstruction Service”*, Machine Vision and Applications, 17(6), pp. 411–426. https://doi.org/10.1007/s00138-006-0027-1

Vogiatzis, G., Torr, P., & Cipolla, R. 2005. *“Multi-View Stereo via Volumetric Graph-Cuts”,* 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR’05), pp. 391- 398. https://doi.org/10.1109/cvpr.2005.238

Wallner, M., Steininger, D., Widhalm, V., Schörghuber, M., & Beleznai, C. 2021. *“RGB-D railway platform monitoring and scene understanding for enhanced passenger safety”,* In International Conference on Pattern Recognition, pp. 656-671. Springer, Cham. https://doi.org/10.48550/arXiv.2102.11730

Wu, C., Agarwal, S., Curless, B., & Seitz, S. M. 2011. *"Multicore bundle adjustment"*, CVPR 2011. https://doi.org/10.1109/cvpr.2011.5995552

Xiao-Shan Gao, Xiao-Rong Hou, Jianliang Tang, & Hang-Fei Cheng. 2003. *"Complete solution classification for the perspective-three-point problem"*, IEEE Transactions on Pattern Analysis and Machine Intelligence, 25(8), pp. 930–943. https://doi.org/10.1109/tpami.2003.1217599

Yang, R., Pollefeys, M., & Welch, G. 2003. *"Dealing with textureless regions and specular highlights - a progressive space carving scheme using a novel photo-consistency measure"*, Proceedings Ninth IEEE International Conference on Computer Vision, pp. 576–584. https://doi.org/10.1109/iccv.2003.1238399

Zarnowski, A., Banaszek, A., & Banaszek, S. 2015. *"Application of Technical Measures and Software in Constructing Photorealistic 3D Models of Historical Building Using Ground-Based and Aerial (UAV) Digital Images"*, Reports on Geodesy and Geoinformatics, 99(1), pp. 54–63. https://doi.org/10.2478/rgg-2015-0012

Zeng G., Paris, S., Quan, L., & Sillion, F. 2005. *"Progressive surface reconstruction from images using a local prior"*, Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1, pp. 1230–1237. https://doi.org/10.1109/iccv.2005.196

Zeybek, M., & Biçici, S. 2021. *"3D Dense Reconstruction of Road Surface from UAV Images and Comparison of SfM Based Software Performance"*, Turkish Journal of Remote Sensing and GIS, pp. 96–105. https://doi.org/10.48123/rsgis.983251

Zhu, S., Shen, T., Zhou, L., Zhang, R., Wang, J., Fang, T., & Quan, L. 2017. *"Parallel structure from motion from local increment to global averaging"*, arXiv preprint arXiv:1702.08601. https://doi.org/10.48550/arXiv.1702.08601

Zhu, S., Zhang, R., Zhou, L., Shen, T., Fang, T., Tan, P., & Quan, L. 2018. *"Very Large-Scale Global SfM by Distributed Motion Averaging"*, 2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition. https://doi.org/10.1109/cvpr.2018.00480

Zhu, W., & Zhou, S. 2022. *"3D Reconstruction Method of Virtual and Real Fusion Based on Machine Learning"*, Mathematical Problems in Engineering, pp. 1–11. https://doi.org/10.1155/2022/7158504