



MATTERSITE

CIRCULARIZING DEMOLITION WASTE WITH
DIGITIZATION AND MACHINE LEARNING STRATEGIES

MATTHEW GORDON

Thesis presented to obtain the qualification of Master Degree from the
Institute for Advanced Architecture of Catalonia

MatterSite

Matthew Gordon

Thesis presented to obtain the qualification of Master Degree from the
Institute for Advanced Architecture of Catalonia

Supervised by:
Alexandre Dubor, Aldo Sollazzo

Industry Advisors:
Stuart Maggs (Scaled Robotics) and Shajay Bhooshan (ZHA Code)

Institute for Advanced Architecture of Catalonia

Master in Robotics and Advanced Construction

Barcelona, Spain

September 24, 2021

Abstract

The building industry is the number one consumer of global raw materials. The majority of these materials are retained in members of the building stock during their lifetime, yet after demolition most are lost as waste, with only one third of these materials being recovered for another use. The roots of this issue include cost of recovery, access to sufficient quantities of material on construction timelines, designer willingness and knowledge to use these materials, and the economic inertia to run markets of these materials. This research aims to address several of these issues, by developing an automated, scalable system for extracting, collating, and communicating quantitative and qualitative data on recoverable materials from demolition sites.

Initially, results from a variety of reality capture methods were compared by efficiency and accuracy. These ultimately create a single point cloud representation, from which methods were tested for transformation into a simple and measurable geometric representation of the space. Simultaneously, a computer vision system was developed to localize relevant materials for recovery within this space, mapping them into the 3d representation. Finally, database and UI systems were evaluated for communicating this information to a future designer or purchaser of these materials. Concurrently, the real-world viability of these techniques was judged by framing this process as a commercial service.

In reality capture, the major techniques balance cost and accuracy. While highly accurate, hardware purchase or commercial service rental costs for LIDAR systems push it beyond the reach of a low-margin demolition industry. Furthermore, the quality obtainable from photogrammetry solutions was largely sufficient for the level of detail required. At the next stage, contemporary point cloud geometric reconstruction algorithms are designed for low level geometry, simpler than volumetric reality, thus creating the need for consolidation and domain-specific reconstruction techniques. The localization systems developed were able to achieve good results within the test sites studied, but had issues with generalization and working with finishes and covering. Finally, the level of analysis and data synthesis was found to be set by the constraints of communication through a digital interface. With the consumer of the data needing to quickly search through and review large amounts of material items, the point clouds, meshes, and even image textures needed to be synthesized down to smaller dimensional and qualitative datapoints. Overall, a framework has been developed for identifying waste materials reclaimable during demolition. Moving forward, further tests will examine generalization when running the complete system with data from active demolition sites, while measuring the timing and efficiency of the processing and analysis. Additionally, companion research is being conducted on design systems to make use of these materials and data.

Keywords

Circular Economy | Demolition | Reality Capture | Computer Vision | Point Cloud



Index

| | | |
|----------|--|-----------|
| 1 | Preface | 4 |
| 1.1 | State of Waste in the Demolition Sector | 4 |
| 1.2 | The Circular Economy of Products and Resources | 6 |
| 1.3 | Current Actors in the Demolition Circular Economy | 8 |
| 1.4 | Placement of Research | 9 |
| 1.5 | Process Overview | 9 |
| 1.6 | Current State of Building Site Digitization and Analysis | 10 |
| 1.7 | Acknowledgements | 12 |
| 2 | Methods for Pre-Demolition Analysis | 13 |
| 2.1 | Overview of Experimental Sites | 13 |
| 2.2 | Site Digitization | 15 |
| 2.2.1 | Photogrammetry | 15 |
| 2.2.2 | Consumer Static LIDAR | 17 |
| 2.2.3 | Consumer Mobile LIDAR | 17 |
| 2.3 | Geometric Reconstruction | 18 |
| 2.3.1 | Form Finding with RANSAC | 18 |
| 2.3.2 | Cleaning Sub-Elements Via Splitting and Segmentation | 19 |
| 2.3.3 | BIM Element Reconstruction | 22 |
| 2.4 | Material Localization | 23 |
| 2.4.1 | Material Class Choice | 24 |
| 2.4.2 | Training Set Assembly | 24 |
| 2.4.3 | Material Classification Methods | 26 |
| 2.4.4 | Patch Combination for Localized Maps | 30 |
| 2.5 | Synthesis | 31 |
| 2.5.1 | Point Matching | 31 |
| 2.5.2 | Reverse Raycasting | 32 |
| 2.6 | Building Dataframe and User Interface | 33 |
| 3 | Methods for Post-Demolition Analysis | 37 |
| 3.1 | Element Scanning Methods | 37 |
| 3.2 | Geometric Reconstruction | 38 |
| 3.3 | Qualitative Element Analysis | 41 |

| | | |
|----------|---|-----------|
| 3.3.1 | Surface Curvature | 41 |
| 3.3.2 | Mesh Skeleton Curvature | 41 |
| 3.3.3 | Textural Defect Detection | 42 |
| 3.3.4 | Textural Evaluation | 43 |
| 3.3.5 | Mass Comparison | 44 |
| 3.4 | Element Tagging,Lifespan Tracking and Interface | 44 |
| 4 | Design and Fabrication Proof of Concept | 49 |
| 4.1 | Test Site Materials | 49 |
| 4.2 | Demolition and Processing Procedure | 50 |
| 4.3 | Design System Overview | 51 |
| 4.4 | Digital Fabrication Methods | 61 |
| 5 | Business Case Investigation | 67 |
| 6 | Results | 69 |
| 6.1 | Comparison of Reality Capture Methods | 69 |
| 6.2 | Localization Performance and Generalizability | 73 |
| 6.3 | Fabricated Demonstrator | 76 |
| 7 | Conclusions | 79 |
| 7.1 | Site Digitization | 79 |
| 7.2 | Material Identification and Reconstruction | 79 |
| 7.3 | Fabricated Demonstrator | 81 |
| | Appendices | 82 |
| A | Grasshopper for Large Scale Data Processing | 83 |
| 8 | References | 85 |

Chapter 1

Preface

1.1 State of Waste in the Demolition Sector

The construction industry is the number one consumer of raw materials globally (WEF 2016), while being one of the largest single producers of waste in the European Union (Herczeg et al. 2014). It accounts for approximately one third of all waste generated, and specifically up to half of all the solid waste produced in the United States. In some European countries, reuse and high-quality recycling (upcycling) of construction and demolition waste (CDW) remains below 3%, while accurate reporting remains different internationally, with many different standards. The sector is facing a crisis of non-sustainable resource consumption, with available technologies held back by a surfeit of unstructured data and poor digitization, culminating in a lack of knowledge and access to circular opportunities.

By 2030, need for housing stocks is expected to rise by 3 billion. Combined with the age of much of the current housing stock, this represents a massive future need vs surfeit of new vs demolished building materials. Approaches to this problem vary across the relevant nations, as seen in Figure 1.2, however the complexity of management means that national attention does not necessarily translate to better rates.

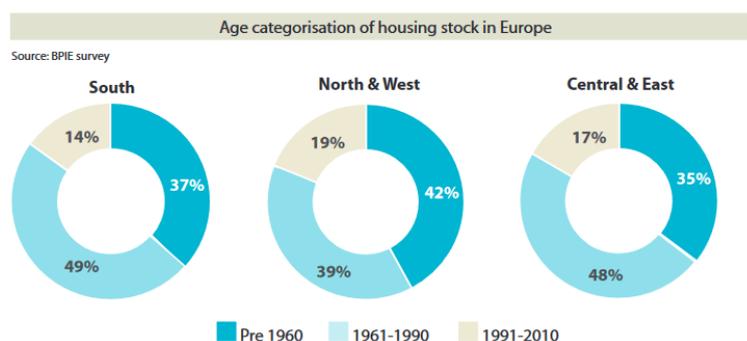


Figure 1.1: Age of current European housing stock as of 2011. Original by BPIE (Economidou 2011)

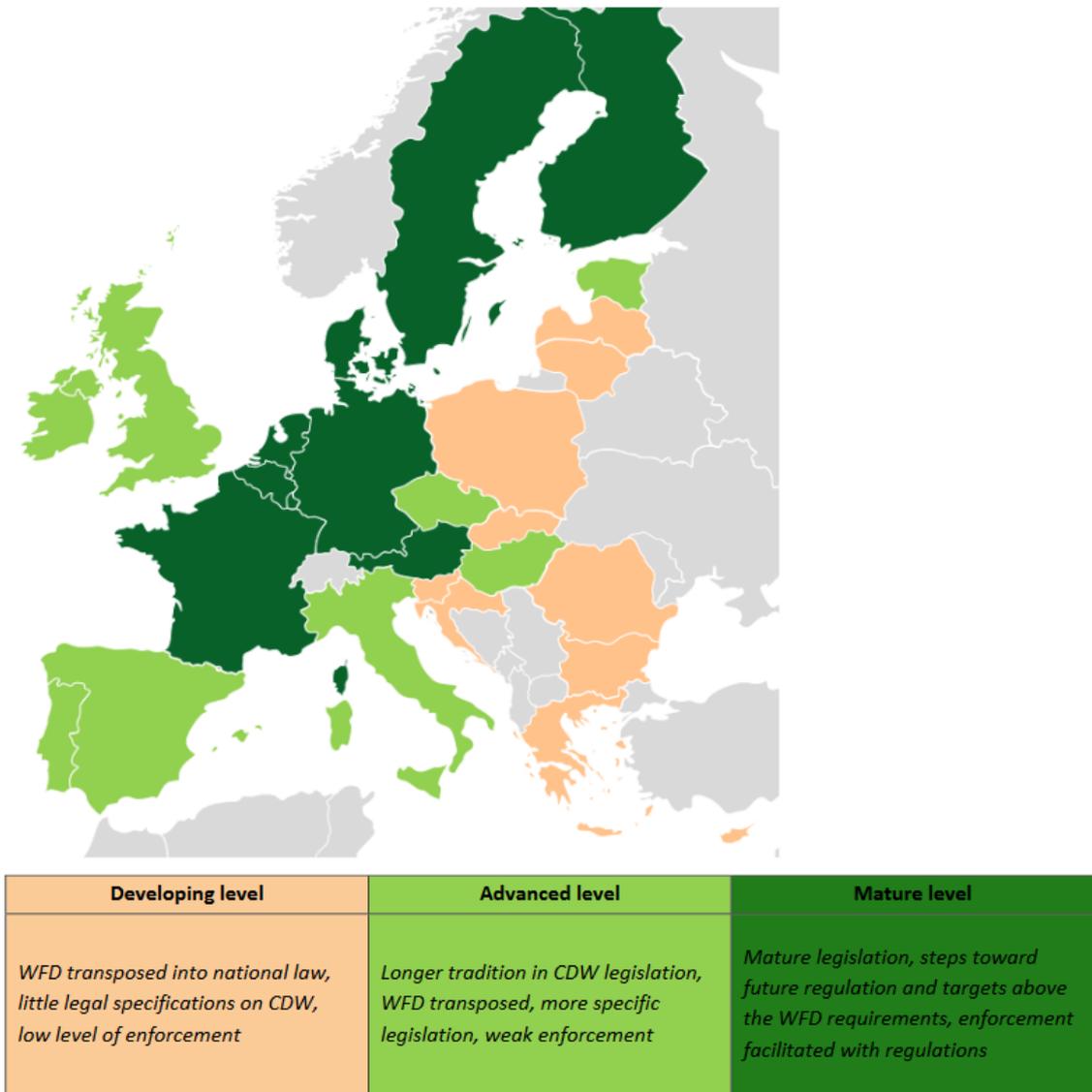


Figure 1.2: European legislative approach to CDW as of 2017. Original by Deloitte (Deloitte 2017)

1.2 The Circular Economy of Products and Resources

The circular economy describes an approach to products, industry and technology, based on distinguishing dead-ends and loops in the long-term movement of products and the resources they are made of. Primary to the concept is the need to clarify the various methods applied to an item's end of life, based both on the physical fate of the material, as well as the energy devoted to the process (Ellen MacArthur Foundation 2019). While applicable to many product spaces, these levels will be described here in terms of buildings. Here, the 'products' may be both discrete elements such as an operable window, or various scales of formal elements, such as concrete slabs.

Firstly are actions which prolong the building from reaching an end-of-life state, including both preventative and reactionary maintenance. Sometimes this is to deal with structural issues, inevitable over a building with a long lifespan, and sometimes for energy concerns as new regulations are passed. Next, assuming the building does reach end-of-life, are methods which re-use products in their near-original state, such as removing and installing an entire stair assembly in a new location. Further, some methods require a greater energy expenditure to refurbish the product into a usable state (such as encasing a recovered window in a new housing to meet modern requirements). Requiring the most energy expenditure, a product may be recycled by deconstructing to its material base and re-manufacturing into a new product. This process, however, also has multiple levels of severity depending on the resulting capabilities for further recycling.

True recycling can be repeated arbitrarily many times (such as correctly separated glass bottles), however 'downcycling' processes may produce diminishing returns over time, or possibly only once (such as grinding concrete for use as new aggregate, with poorer structural results).

From here, total waste and disposal represents the final, worse possibility. The viability and industrial reality of each of these stages is closely tied to the economic infrastructure in place to run or use it (that is, a technologically recoverable material with no buyers does not serve the overall goal) (UN Directorate-General for Environment 2008).

While refurbishment-in-place is considered the most positive outcome, many contemporary groups and researchers focus on the lowest divide - between waste and reuse/recycling, as therein lie the greatest untapped sources of value.

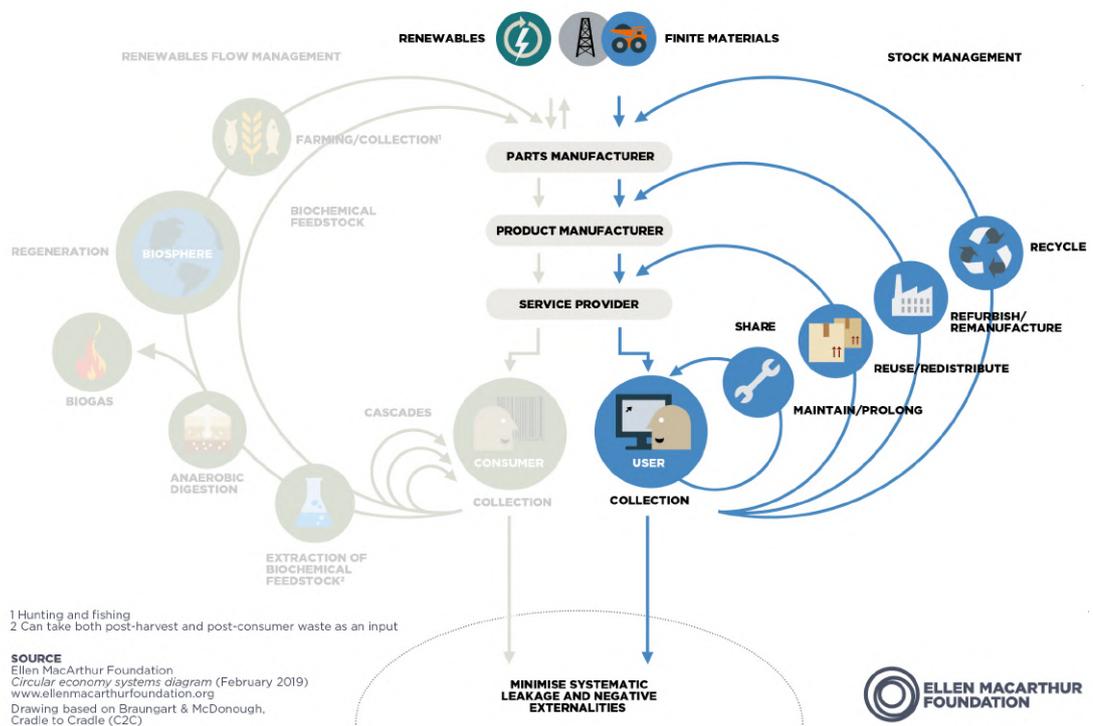


Figure 1.3: Circular Economy industry diagram showing different resource flow levels, highlighting the processes for products. Original by Ellen MacArthur Foundation(Ellen MacArthur Foundation 2021)

1.3 Current Actors in the Demolition Circular Economy

Contemporary circular-economy actors occupy several broad groups, generally following the different stages of a material's lifespan.

Deconstruction Companies

To obtain usable materials, deconstruction needs to be performed with re-use in mind, thus demolition companies must work with specific intention. This process involves pre-planning for scheduling the safe removal of materials at different stages of demolition, specific at-location practices to not damage materials upon removal, and on-site marking practices to track materials once removed. As the building marketplace is generally not aligned with this methodology, companies choosing to employ it often must work on lower margins, or subsidize the methods by also functioning as another role. Existing companies in this space include New Horizon in the Netherlands, and Rotor in Belgium, though both are to varying extents involved in other stages of the process as well.

Reclaimed Material Marketplaces

Marketplaces serve as platforms from which one or several deconstruction groups sell recovered materials. Materials may be stored with the marketplace while awaiting sale, or with individual deconstruction actors and owners. Due to the variability of elements after recoverability, materials may be listed as types rather than individual products, with representative images and ranges of dimensions. The listed origins of materials can also range from specific sites to general regions. European marketplaces include Excess Material Exchange in the Netherlands, which focuses on finding the maximum economic value for a material, and the marketplace arm of New Horizon, which puts emphasis on the value of quick connections with new buyers, to reduce storage and transport costs.

Circular-Minded Design

Re-use of materials at scale requires many additional considerations at the design development, structural certification, and construction management stages. Contemporary projects are generally one-off in methodology; either a client with a specific interest, or a demonstration piece to explore new methods. Thus generally, design with recovered materials is neither widespread or easy to integrate. Contemporary design groups working with these methods include WAO Paris and Bauburo in Situ (Switzerland).

Also important from a design side, the new-construction complementary approach covers the design of new buildings with new or recovered materials, but focused on the next stage of deconstruction. This approach seeks to bypass the many difficulties of material recovery present in today's buildings, with techniques covering Design for Deconstruction. However, while critical in the long term, the effects of this methodology won't start to pay off until these new buildings reach end-of-life, on a roughly 40-year timeline.

Advisors and Networks

Given the difficulties of working at scale, many circular economy operators are small-volume or operate in a limited area. For this reason, some groups exist primarily to connect disparate actors and ensure needs can be met locally. This allows for clients or designers to work in this manner to operate without as much domain expertise. These include Opalis in Belgium, which focuses on mapping a wide variety of operators and services, and Building Material Scout in Germany, which produces a broad documentation database for the techniques and details of working with a variety of recovered materials.

1.4 Placement of Research

In order to diminish the dependency on new natural resources and reduce the overall environmental costs of the industry (extraction of raw material, transportation, and the manufacturing of new construction elements), this project develops an automatic digitization method for the near-end-of-life stage of a building, and considers it as a source of high value assets. This analysis may be performed by building owners, or deconstruction groups as a new first step in their process. This is supplemented by the additional data taken from the elements at extraction time, again performed by the relevant deconstruction groups, or specialists working the marketplaces who will move the extracted products. The subsequent dataset is then transferred to designers and builders to promote the use of valuable secondary-source materials and better inform early design decisions when repurposing construction waste.

1.5 Process Overview

The first part of the research develops an innovative digital analysis and logistic support system for pre-demolition sites, integrating several tools and processes (3d reality capture, computer-vision based material localization, and geometric site reconstruction), that help to define the most sustainable and economical deconstruction and reuse strategy for a building. The results show that it is possible to quantify, qualify and map the flow of products and materials from buildings undergoing renovation, redevelopment, or deconstruction, and assess their potential for reuse and recycling. The second part of the research links the database with a computational design tool that can be integrated into construction software for architects and construction companies. The proposed system both matches the designed components with relevant stored materials by their design requirements, as well as providing suggestions for design changes. The proposed iterations aim to optimize repurposed material utilization, performance, and cost.

1.6 Current State of Building Site Digitization and Analysis

The AEC field has brought a particular set of new problems to the long-running fields of computer vision and geometric digitization. Many new advancements stem from the intersections of the specific types of data available on construction sites, and the more constrained data of many digitization problems.

Computer Vision

In 2014, Dimitrov and Golparvar Fard developed a SVM-based classification system specifically for building materials, to aid in automated digital reconstruction for progress monitoring (Dimitrov & Golparvar-Fard 2014). This system focused on overcoming changing environmental conditions and the textural particulars of the relevant materials involved, however was focused on individual classifications and did not yet tie into the 3d reconstructions.

Construction sites in particular present additional challenges to the normal methods of computer vision. For instance, as studied by Xiao et al. (2021) (Xiao et al. 2020), modern computer vision systems often put attention on background details than expected, and an active building or demolition site will contain many conditions not present in a carefully constructed training set.

Geometry and Point Clouds

When digitizing a building the first time, point clouds (collections of colored and oriented points in space) are the widely accepted datatype when considering efficiency and automation.

Though present in the field in general, digitizing buildings particularly highlights the problem of point cloud registration and alignment (connecting point clouds captured in different locations). In 2017, Wijmans and Furukawa developed a method for improving scan alignment using 2d floor plans (Wijmans & Furukawa 2016). While this data is not guaranteed for all buildings, the method was able to use relatively low-resolution scans, and effectively reduce the necessary number of scans taken.

Even after a reliable point cloud is extracted, there is the primary issue of accessing useful and semantic information. Recent reviews of this problem summarize the issues as stemming from a point cloud's inherent irregularity, unstructured form, and unordered nature. (Bello et al. 2020)

While there are some specific semantic analysis systems in common production use, such as the Canupo Classifier (Brodu & Lague 2012), the techniques and models are developed for a specific industry niche (e.g. geology).

However, some methods have been developed specifically for the building reconstruction domain. For instance, Macher et. al. developed a method of segmentation of basic element types, employing the selective projection of points into planes to also make use of computer vision techniques. (Macher et al. 2015). Approaching the 3d data more fully, Maalek et. al. developed a classification system based on individual point classification

and domain-specific clustering(Maalek et al. 2018). Looking to the additional data that could be captured from the techniques used, Yuan et. al. developed a classifier that also makes use of reflectance and surface roughness in its calculation.(Yuan et al. 2020)

In many cases, the research is primarily focused on accurate classification of different element types, over the final reconstruction into the geometric primitives forming these types. While a comparatively easy problem, it is nonetheless critical for the goals of complete digitization automation.

Element Level Analysis

Beyond general construction-domain questions, research has begun focusing on analysis of individual elements. For example, methods using precise data capture have been developed for reusing large-scale concrete into dry masonry with minimal shaping (Design et al. 2019).

Before the specifics of reuse, there is already substantial research into defect detection of various types, both for new use and conservation efforts. For instance, Ding et.al. developed a neural network system specifically for localizing knots and checking in wood, obtaining high accuracy with low resolution inputs(Ding et al. 2020). While we developed our own similar method focusing on a different range of defects(see 3.3.3), many of these diverse detectors will need to be brought together for a complete automatic quality analysis.

There have also been considerations for the infrastructure required to re-use data-critical elements (i.e. the calculated and tested capabilities of structural elements) at scale. In 2020, Bertin et. al. explored the needs for communication, storage, and tracking of BIM data for structural elements and how it can be communicated to new design(Bertin et al. 2020).

Taking a wider view of the question of data flow in the industry, the ReflowOS project is developing a distributed system of tracking resources in communities, elevating the materials into the broader economic network(Reflow 2021).

Several groups have also begun to apply element-scale reality-capture and analysis in a commercial manner. For example, Scaled Robotics and Avvir employ Lidar-based reconstruction for construction monitoring and verification.

1.7 Acknowledgements

This thesis was developed as a joint research with Roberto Vargas, who was an invaluable friend and research partner, and an ideal foil to my own approaches and focuses.

Thanks to our directors Alexandre Dubor and Aldo Sollazo for their guidance and support as this thesis developed from a single studio project, which was developed in collaboration with Anna Batalle, Irem Yagmur Cebeci, and Roberto Vargas.

Thanks to Stuart Maggs, Pol Cirujeda, and Shajay Bhooshan for continuing business and technical advice across a very broad project. Thanks to Peter Naaijken and Jose Artesero for valuable insight into the contemporary demolition field and connections to live demolition projects.

This thesis was developed to realistically frame the technology developed as ready for use by a new startup. Thanks to Driven by Volumes for their support and advice regarding the economics of this endeavor.



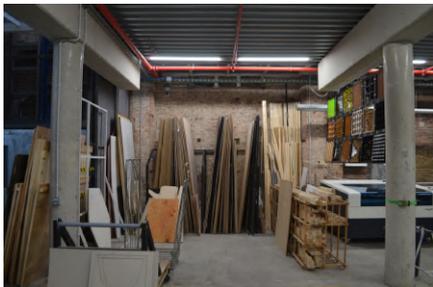
Chapter 2

Methods for Pre-Demolition Analysis

2.1 Overview of Experimental Sites

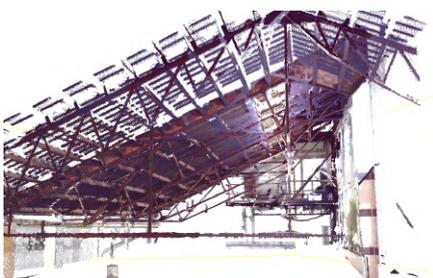
Testing was performed on a variety of datasets, taken from active demolition sites, as well as approximate data sources chosen for testing different parts of the system. See 4.1 for a description of the site and materials used for the postdemolition and design phases.

IaaC Atelier Photogrammetry



Initial photogrammetry tests were performed in the IaaC Atelier building. While not an active demolition site, the site is a converted warehouse space with exposed concrete, brick, metal decking, and wooden beams. Also on site was stored plywood and lumber, presenting relevant wooden examples in a different context. Site photography also contained common clutter elements including people, furniture, and tools. This will be referenced as site IAAC1.

IaaC Roof Scan



The roof structure of the IaaC Main hall had previously been scanned as a point cloud, and was used as an alternative test case. Instead of photogrammetry, this was compiled from several static Lidar positions on ground level. The point-cloud data was used for testing early element reconstruction systems, as well as the early iterations of the design system. This will be referenced as site IAAC2.

Scaled Robotics Dataset



Project advisor Scaled Robotics provided access to example LIDAR data from one of their analyzed sites, providing a real-world look at terrestrial LIDAR data. While this data was captured during construction not demolition, it contains a range of exposed materials with reasonable on-site clutter. This data was used for continued testing of the geometric reconstruction system. This will be referenced as site SCALED1.

Leica Example Data



The LIDAR hardware manufacturer Leica supplies example data for various of their models in various contexts. This allowed us to judge the quality of various input sources without the need to purchase various hardware models. This data was used for testing element reconstruction, as well as the tests for artificial image synthesis when only point clouds are available. ^a. This will be referred to as site LEICA1.

^aThis system was not finalized in the ultimate workflow, as sites with clouds but no photography were considered unlikely

Barcelona Demolition Site



The primary test site was an active demolition site in the Poble-Sec neighborhood of Barcelona, scanned in collaboration with a local demolition company. Built in 1900, the scanned area occupies 196 square meters, and due to undergoing demolition before renovation, contained exposed materials and structure with fitout removed. The site also featured some common deviations that the system will encounter in the field, included painted brick and asbestos roofing materials that closely resemble corrugated metal roofing. For the sake of testing, it was decided to consider metal to be the 'correct' classification for these elements. This will also be referenced as site BCN1.

2.2 Site Digitization

Site Digitization technology covers a broad range of techniques for capturing spatial data from the real world into computer systems. It generally creates collections of unorganized data (either points or pixels) which must be further analyzed to understand what they represent. In our workflow, methods tested for gathering site data had two primary goals. Firstly, to obtain three-dimensional geometric data to reconstruct the layout and dimensions of site elements; and secondly two-dimensional imagery to perform the material localization. Each method primarily focuses on one of these goals, and had accuracy and alignment considerations for how the type is extracted or generated.

2.2.1 Photogrammetry

Photogrammetry is a process by which spatial data is created from dense collections of site photography, usually in the form of point clouds. Output quality and processing time can vary wildly, and improved by tunable parameters in software as well as operator expertise when capturing the initial imagery. Many existing tools exist, both open source and primarily research based, as well as proprietary and commercially aimed.



Figure 2.1: An operator taking photo captures of a site, with mockup guide-software indicating photo coverage

Agisoft Metashape

Metashape is a proprietary photogrammetry software developed by Agisoft, with capabilities up to GIS scale. In addition to dense clouds, it also produces textured meshes of the reconstructed scenes. It was tested with version 1.7.3. In context, this will be referenced as MS.

Meshroom

Meshroom is an open source photogrammetry tool based on the AliceVision framework. It was tested with version 2021.1.0. In context, this will be referenced as MR.

COLMAP

COLMAP is an open source reconstruction tool based on SfM and MVS technologies. In context, this will be referenced as CM.

Photogrammetry Capture Methods

A standard practice was decided for taking photos for use with photogrammetry. In general, the outside edge of the room was traced by the operator, while facing towards the center of the room, or the longest visible depth, to capture maximum parallax. Photos were taken approximately 1 meter apart, with alterations made if nearby elements would block the field of view (for example a very close column). Photos were captured in two passes, first pointing directly at the 'horizon' of the room, and secondly pointed upwards at roughly 30 degrees to capture roof structure details. Photos for a site were taken in one session and by one operator, as it was found that differing times of day as well as resolution and lens settings could highly effect feature detection and thus alignment between photos.

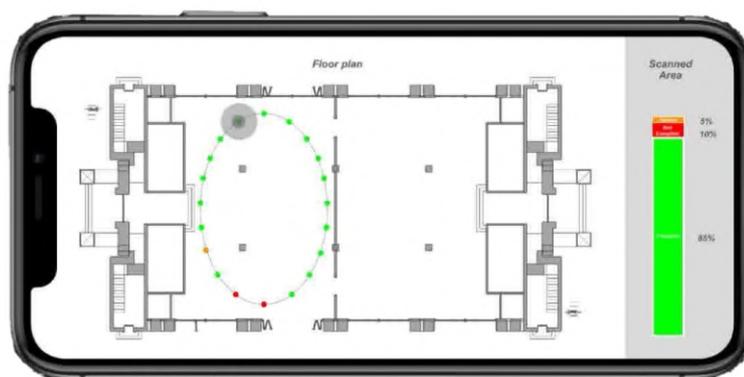


Figure 2.2: Mockup of guide software when showing photo placement in a room

2.2.2 Consumer Static LIDAR

This scanning was carried out using an Apple Lidar Scanner equipped device, employing a single-location reconstruction software. The user moved similar to how a terrestrial LIDAR system would; standing in one location and orienting the device in various directions. Registration between scan locations was performed automatically by the software product during processing.

2.2.3 Consumer Mobile LIDAR

This also made use of the Apple Lidar Scanner, however it employed a continuous-scanning software product. Here the user performed a mobile scan of the space, walking through the space as they oriented the scanner. This would likely be a more natural method for an inexperienced operator as a live sparse recording could indicate areas that were missed.

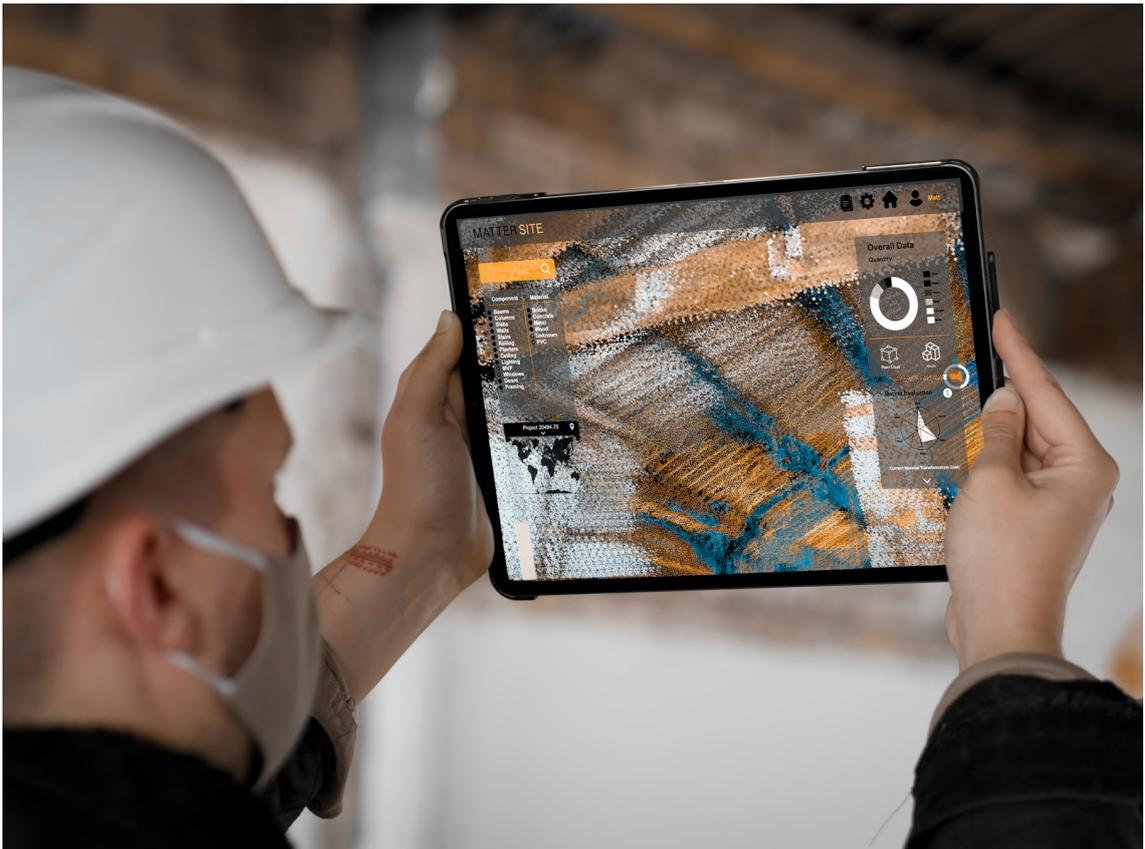


Figure 2.3: An operator scanning a site using an Apple Lidar system, with a mockup of a domain-oriented assistive software

2.3 Geometric Reconstruction

During the geometric reconstruction step, we transformed the flat and context-less point-cloud data into building element primitives, that represented the geometry of our final BIM objects. Our initial implementation specifically reconstructed 'slabs' (generically, floors and ceilings of solid or component construction), walls, columns (considering cylindrical and rectangular), and beams. While commercial services exist for BIM digitization, here we attempted to make the process as automated as possible, by essentially encoding the recognizable features of various element types in a relatively simple decision system.

2.3.1 Form Finding with RANSAC

First, simple geometric primitives were extracted from the point cloud. This was performed using the implementation of the RANSAC-based shape detection algorithm (Schnabel et al. 2007) built into CloudCompare. This specifically looked for planes and cylinders, which were enough to cover the desired BIM types. As the process expands, these primitives will stay relevant for most types, and only require new rules to describe how they fit together.

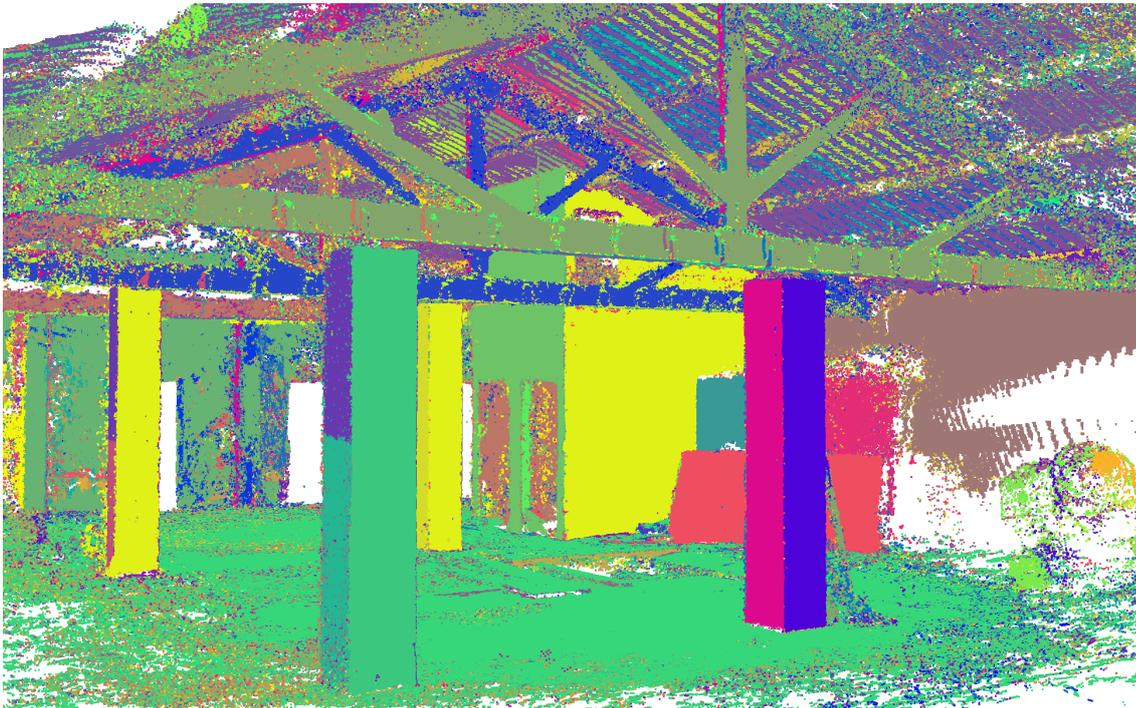


Figure 2.4: Raw clustering output from RANSAC on site BCN1. Errors visible can be addressed by further RANSAC tuning, as well as by methods outlined in 2.3.2

2.3.2 Cleaning Sub-Elements Via Splitting and Segmentation

Some subclouds obtained from RANSAC did not correctly map to a component of a single BIM object. The first common instance was for cylindrical shapes; this ideally should only catch cylindrical columns. However, often RANSAC would misinterpret noisy points or some slight actual curvature in what should be a planar element, and interpret it as a chunk of a very large cylinder. This was detected heuristically; as in this domain cylinders will always be vertical. When checking a possible cylinder the points were projected to the world plane, then their distance from the centroid was calculated. If a cloud had many distances with a high deviation from the mean, it was likely not a true cylinder, and the subcloud was refiled with the planes for further analysis.

The larger issue stems from the intersections and holes in planes. The AEC domain contains many situations where different elements will have adjoining and parallel faces. In our tests, this commonly occurred where columns met beams or along one side of a truss. Processing these subcloud without alteration would lead to incorrect volume estimates and being unable to reconstruct correct BIM element geometry.

Thus, each subcloud that was deemed necessary for further splitting was oriented onto the two-dimensional world plane, and treated as a low-dimensional clustering problem. Several methods were tested for this.



Figure 2.5: In site IAAC1, RANSAC has correctly extracted one face of the concrete beam, but has assumed part of the adjoining column, as well as some clutter elements, to be part of the plane

Segmentation with Manual Locality Analysis

This method focused foremost on finding a directionality for each area of a cloud (i.e. the long edge of each truss component). Each point took a subcloud of its local neighbors within a certain radius, and calculated the eigenvector thereof. Points with similar directional vectors were then grouped. The base implementation had poor results, with high variability based on the local radius. If too small, a neighborhood would not catch enough points to accurately find the directional vector, whereas too large and it would start to pick up points outside the final object, leading to inaccurate results. While this variable could likely be heuristically tuned across different areas of the cloud, it was decided at this point to test adapting existing algorithms. The results can be seen in the second cloud in Figure 2.6.

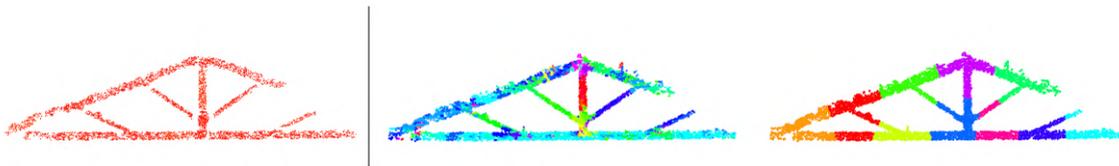


Figure 2.6: From left to right : points of the raw 'plane' as extracted by RANSAC, clustering using manual locality, clustering using KMeans

Segmentation with KMeans

K-Means was chosen as a baseline for non-domain specific clustering methods. While producing cleaner individual clusters, K-Means is focused entirely on distance, and thus makes no distinctions at the borders between intersecting components. Additionally, the number of clusters must be known beforehand, with automatic cluster count estimation also being a nontrivial problem. The results can be seen in the last cloud in Figure 2.6.

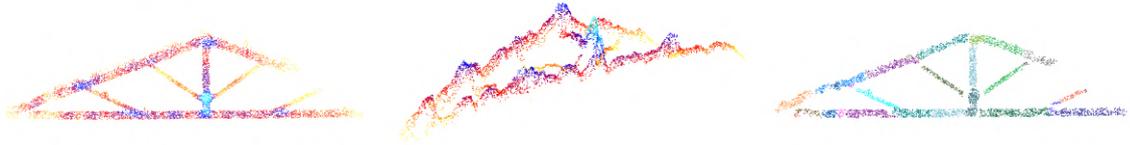


Figure 2.7: Each point calculates its local neighborhood density, and is translated on the Z axis accordingly

Segmentation with DBScan

The DBSCAN algorithm was specifically designed with noisy spatial data in mind, and takes into account both variable local density and local edges (Ester et al. 1996). Additionally, cluster count is not needed to be decided before hand. However, additional tuning was still employed to accentuate the density factor. Here, each point was moved first moved in the Z axis according to the density of its neighborhood (see Figure 2.7. This led to clusters with greater respect for component joints, although they often still didn't capture the whole element. These clusters were then merged by looking for adjoining clouds of a similar eigenvector. The final result is shown in Figure 2.8.

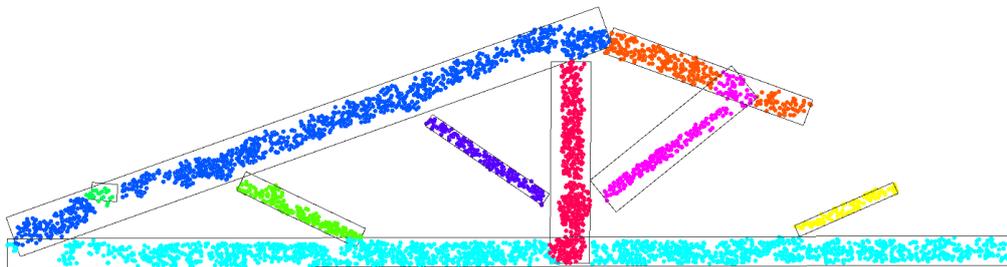


Figure 2.8: Final segmentation of the truss. Some elements still extend too far into their intersections

2.3.3 BIM Element Reconstruction

Once each subcloud was fully split or merged until representing a part of only a single BIM object, they were joined and meshed into BIM representations of these elements.

First each subcloud was heuristically sorted to determine which element type it was part of. Cylindrical columns were simplest; all cylinder objects that were not sorted out during the earlier step were automatically vertical cylindrical columns, whose dimensions could be read simply from the bounding box. From here, the normal, eigenvector, and in-plane dimensions of the subcloud were considered. See figure 2.9.

Different methods were used for each type to merge subclouds into the final element. Within each element type, the largest subclouds iteratively looked through a sorted list of smaller subclouds until no further merges occurred. For walls and slabs, each cloud looked for subclouds of similar normal orientation, with a centerpoint within a certain distance of the existing plane. Beams looked for subclouds of a similar eigenvector orientation, with a tuned maximum distance between closest points in the clouds (to reduce over-joining). As columns had a simpler domain, they were merged simply looking only at proximity in the XY plane.

A reference frame was then found for each final merged cloud. The Z-basis remained the normal of the parent cloud plane, flipping as necessary to point roughly in positive-Z. The X-basis was the eigenvector of the final cloud, thus following e.g. a beams long edge. This frame was then used to align a bounding box which formed the elements final primitive geometry.

These geometries were stored as 9 main values. Firstly, the type of element, then the position and dimensions of the box in the XYZ axes. Lastly, the orientation of the box was considered as spherical coordinates, and the azimuth and inclination were stored.

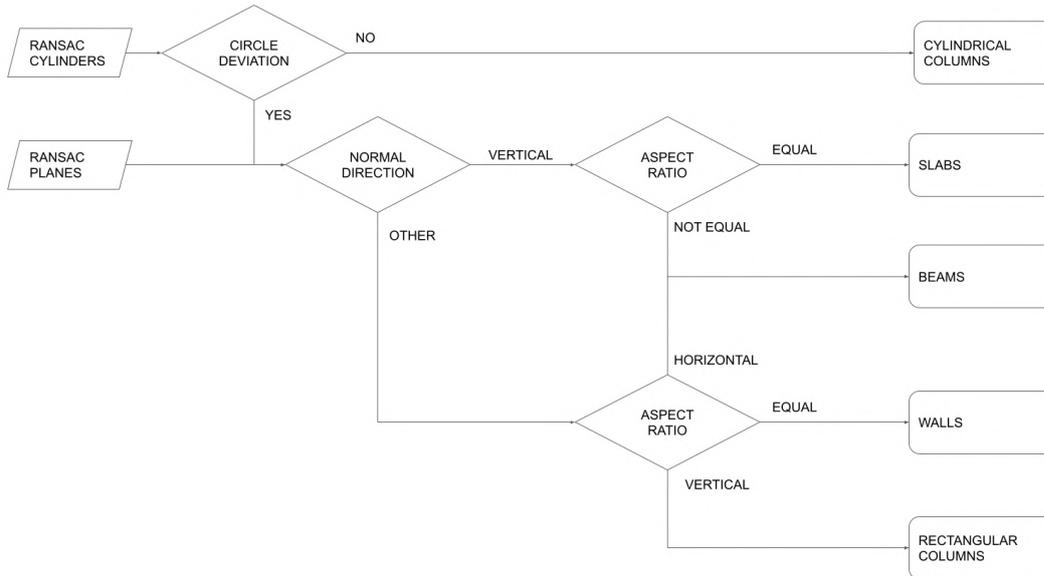


Figure 2.9: Decisions process for assigning a subcloud to a BIM type

2.4 Material Localization

Once the shape and form of the on-site elements are determined, it is necessary to understand the specific materials that make them up, and thus how useful the element is for recovery. Given the current state of the art, it was decided to perform this classification using 2d computer-vision techniques, rather than point-cloud analysis or other 3d methods.

The structure of the final workflow took an input image captured at the site, (ideally used in the earlier photogrammetry steps), and created a class map, of various possible resolutions. This used several classification methods in a custom patch combination system. This was chosen over full-image methods such as Mask R-CNN (He et al. 2018), given the slow ramp-up time to manually annotate ground-truth images, and the relative difficulty of debugging new classes in a CNN system vs more inspectable SVM based systems.

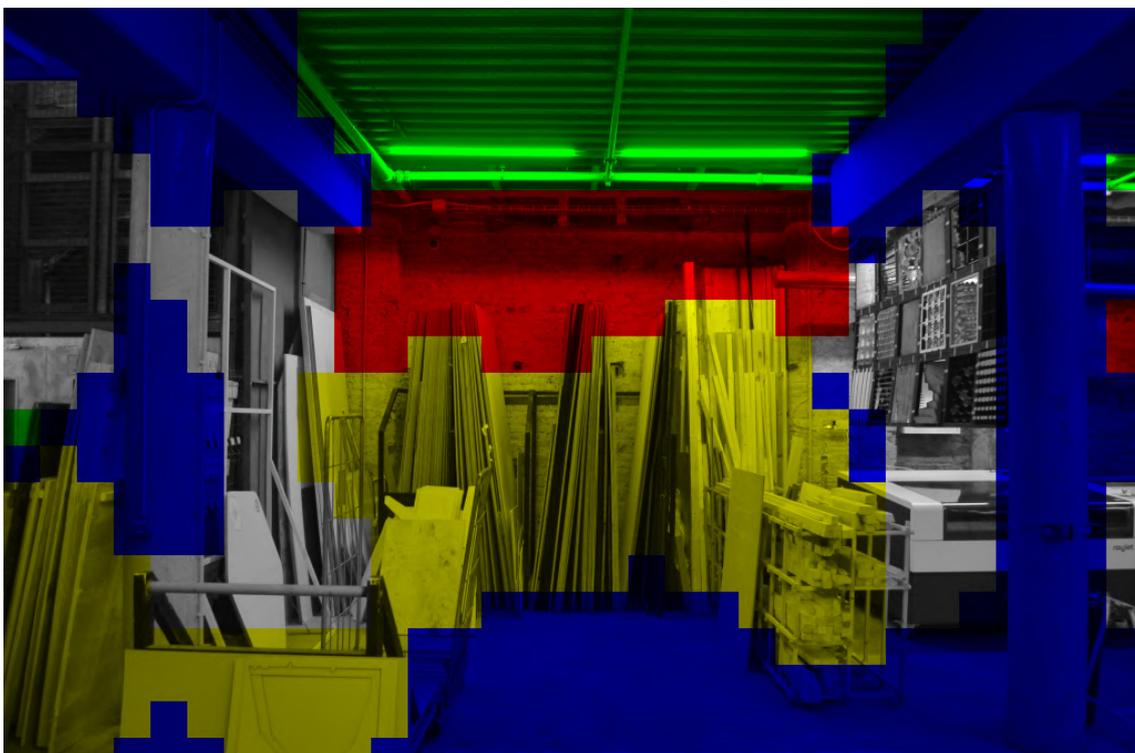


Figure 2.10: Example of a output classification map from site IAAC1

2.4.1 Material Class Choice

The central classification task takes place on individual patches of the image; generally 100x100 pixels. Each patch is considered to have only a single class; that is the only (or predominate) material visible in the patch. The considered classes are as follows.

- Brick : Structural or facade brick masonry ¹
- Concrete : Generally structural poured or precast concrete elements; including slabs, columns, beams etc
- Metal : Predominantly taking the form of beams and trusses, wall studs, or corrugated slab components. Makes no distinction between steel, aluminum, etc.
- Wood : Generally wall studs or beams and trusses, also commonly present in facades
- None : A patch not showing any chosen recoverable material
- Unconfident : A patch with low confidence ratings.²

2.4.2 Training Set Assembly

The final training set contained 3654 images, across the four material classes and the None class, as shown in table 2.1. These images were assembled from manual photography of clear examples, product images compiled using the internet, and masks of the hand-labelled materials in ground-truth images (e.g. 2.11). All of these sources contained particular drawbacks. While manual photography was guaranteed to be clear and highly-detailed, it limited by the author’s local environment during the course of the investigation. While our major test sites were also located in Barcelona, expansion of the range of application will expose the system to a new range of styles and building practices. When using manufacturer imagery, product photos are often contextless, or from a limited range of angles limiting the full set of features that should be associated with these items. Finally, the primary drawback to hand-labelled imagery is time. Many large scale machine-learning database-gathering efforts use third party workers for manual labelling. The specifics of this problem have also been approached in the architectural space, for instance in Han and Galpalvar-Fard’s system for crowdsourcing classification of construction-site material surfaces(Han & Golparvar-Fard 2017).

| Brick | Concrete | Metal | Wood | None | Total |
|-------|----------|-------|------|------|--------------|
| 914 | 494 | 647 | 790 | 809 | 3654 |

Table 2.1: Training set sizes between classes

¹see 7.2 regarding the limits of this delineation

²This is not a class arrived at by the classifier itself, but instead is reassigned to all patches with low ultimate confidence values

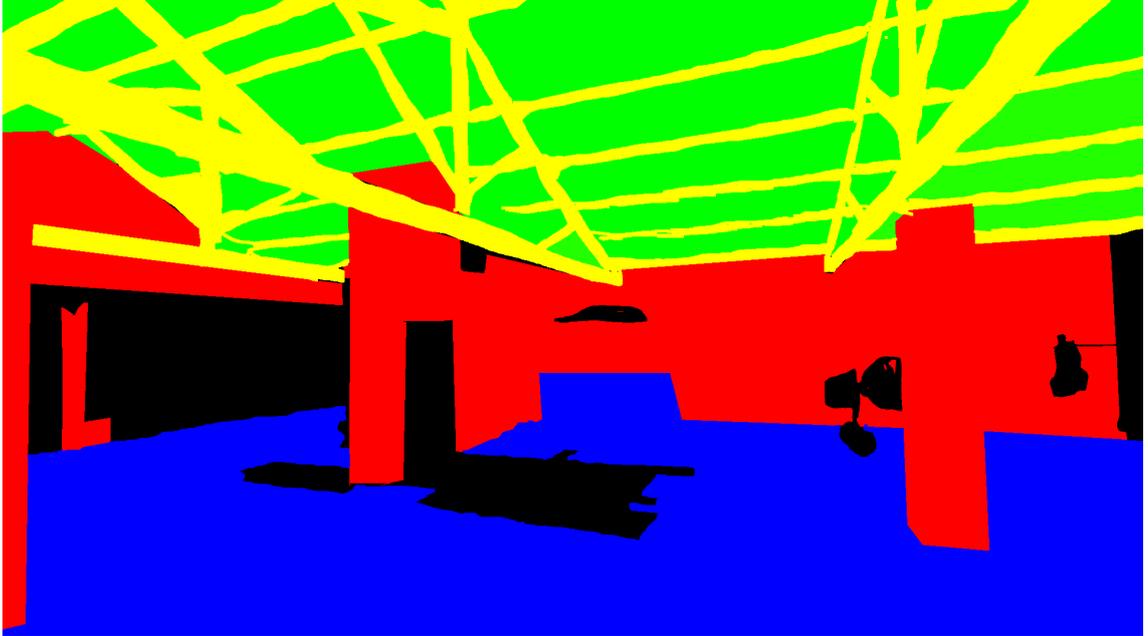


Figure 2.11: Example hand-annotated ground truth full-resolution class map for site BCN1

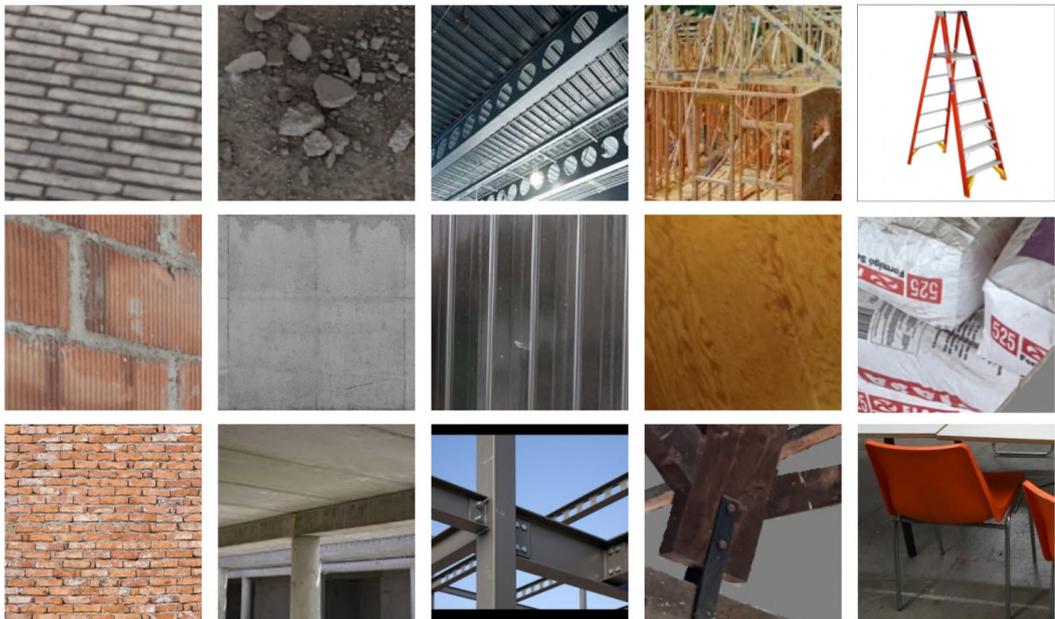


Figure 2.12: Sample of training images from various sources, across the relevant classes

2.4.3 Material Classification Methods

Support Vector Machines (SVM) were chosen as the primary model for classification. While originally designed as non-probabilistic binary classifiers, extensions to the system allow for multiclass problems as well as probabilistic results. In general, these algorithms take an arbitrarily shaped vector of numerical data, and return a flat vector of the probability for each class. The important distinctions SVM-based image classifiers lie in how these input vectors are generated; ranging from directly summing an images pixels to more complex feature analysis and counting. Additionally, in our system, additional transformation was applied to the output vectors before the most likely class was determined.

The main classification algorithm is based on the stack-classifier model. This involved taking the 5-value class probability vectors from several other classifiers and concatenating them into a final 20-value vector, on which a final SVM is trained. From here, these probabilities are stored with the patches they cover, although further processing was applied as detailed in 2.4.4.

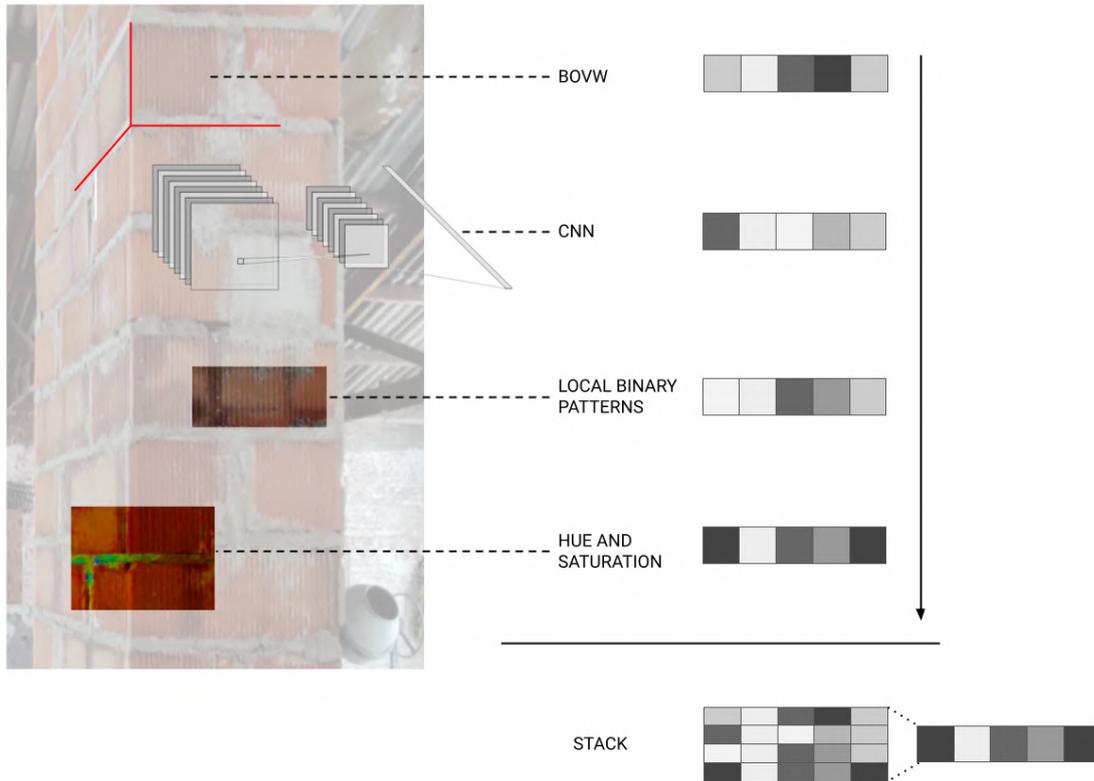


Figure 2.13: Intermediate classifiers combined within the final stack classifier

Bag of Visual Words Classifier

The central classifier is based on the Bag of Visual Words technique. This follows the common computer-vision methodology of first finding the features (small easily recognizable patterns) in an image patch, then analyzing which, and how many, features are present. In this case, the number of possible features is very large, so a vector containing the feature counts would be extremely sparse and hard to analyze, so additional processing is performed. During training, the entire corpus of features is extracted from the supplied dataset, and stored as a series of descriptions. In this case, the GFTT algorithm was used as the feature detector, and the BRISK algorithm for the Descriptor. For the purposes of time and available memory, subsampling of this descriptor corpus was used going forward; we chose a 25% sampling. Stored as 64-value vectors, this descriptor corpus was then clustered using the K-Means algorithm into a limited number of 'visual words' that are present across the training set; we clustered to a size of 1000 words. The final vector trained against for each training image was thus a histogram of the concentration of these 'words' within each image. This algorithm contained many tunable parameters, and while well-performing settings were found via exploration, the space was not examined exhaustively here, or correlated with e.g. how the algorithm responds to different data sources in a training set.

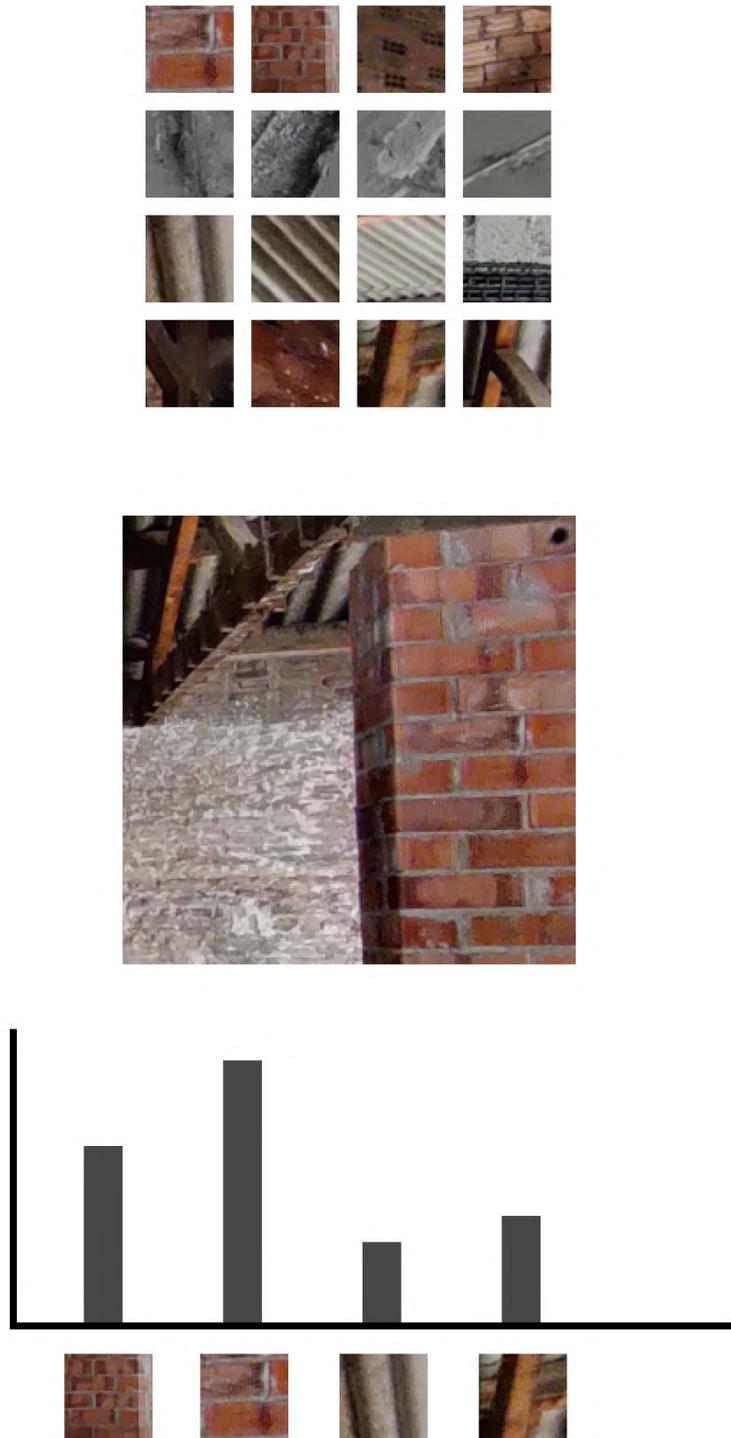


Figure 2.14: A rhetorical example of a set of Visual Words, and the concentrations existing in a sample patch, indicating the sample contains mostly features associated with the Brick class. Note that in the true model these features may be much smaller and less distinct.

Local Binary Patterns Classifier

Local Binary Patterns is a simpler descriptor system focused on identifying textural patterns in images. This system involves determining unique numerical representations of brightness patterns at various scales. For each path, the calculated 'pattern ids' are stored in a histogram vector of all possible combinations (i.e. size 256). This histogram is associated with a class and used to train an SVM.

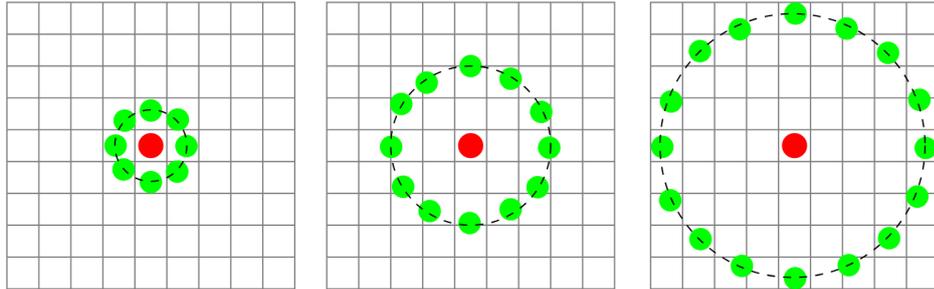


Figure 2.15: Local patterns at various scales. The center pixel is compared to each outer pixel to form a descriptor vector.

Hue and Saturation Density Classifier

The BOVW and LBP classifiers ignore hue and saturation data and only analyze brightness. As this ignored data may still contain useful information for this domain, it was used in the third classifier in a simple manner. For each image, the H and S space was divided into size-16 normalized histograms, which were concatenated into a size-32 vector for training an SVM.

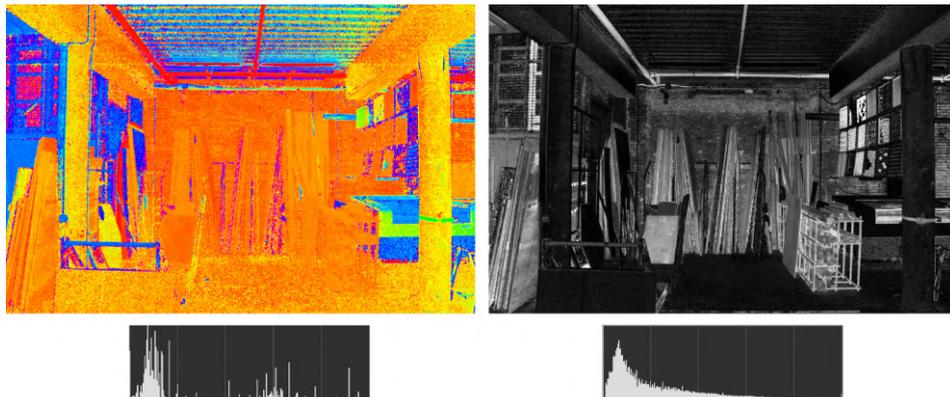


Figure 2.16: Example of hue and saturation channels taken from an image of site IAAC1, with expanded histograms

Convolutional Neural Network Classifier

Lastly, a Convolutional Neural Network (CNN) Classifier was used. This model was built using the Tensorflow and Keras frameworks, to make use of CUDA based graphics-card acceleration (see 6.2 for a measurement of classifier speed). CNN's are generally divided into two stages; a series of convolutional layers based on analyzing progressively larger patterns in small kernels of the image, then a series of dense (fully connected layers), to arrive at a final size-5 vector of class probabilities.

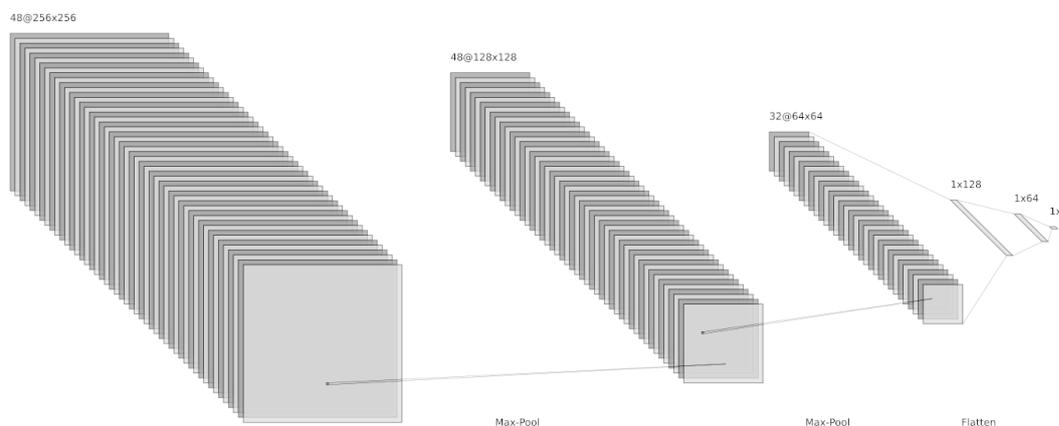


Figure 2.17: Architecture of the CNN used for classification

2.4.4 Patch Combination for Localized Maps

With this system for classifying individual patches in place, the patches were combined into an (often lower-resolution) classification map of the entire image. This was achieved by considering the output map to be made of subpatches, which fit evenly into the classified window size. The final method used 100px classification patches with 33px subpatches. The classifier thus follows a moving window across the image, to consider each subpatch in several local contexts, and each classified patch thus adds its probabilities to a list stored at each subpatch location.

Several methods were tested for combining the local results at each subpatch. The simplest method involved taking the average (arithmetic mean) of the probabilities. Additionally, the harmonic mean (used often for compiling machine learning accuracy scores), and geometric mean³ (which decreases with greater spreads) were tested. Several additional factors were tested with these methods, including applying a falloff (adding a

³As classes have the possibility of receiving a probability of zero, the pure geometric mean is unusable, so the common workaround described in de la Cruz et.al. (de la Cruz & Kreft 2019) was applied, wherein a value of 1 was added to each datapoint before calculation, and 1 subtracted from the end result. While, as mentioned, this method is not necessarily recoverable to the 'real' geometric mean, it remains monotonic, and is thus usable for comparison here.

stronger probability at subpatches in the middle of a patch, with less at the edges). To combat a loss of detail when using averaging, a local difference was also applied, where the probability of a category at a particular patch was boosted the greater it was than its immediate neighbors.

These methods were tested in isolation, assuming a perfect classifier as their input. At times in development, a salt vector (constant positive or negative modifier) was added to the final probability vectors, to account to perceived overconfidence in certain classes. However, no good method for automatically tuning these values was found, so was not used in the final workflow.

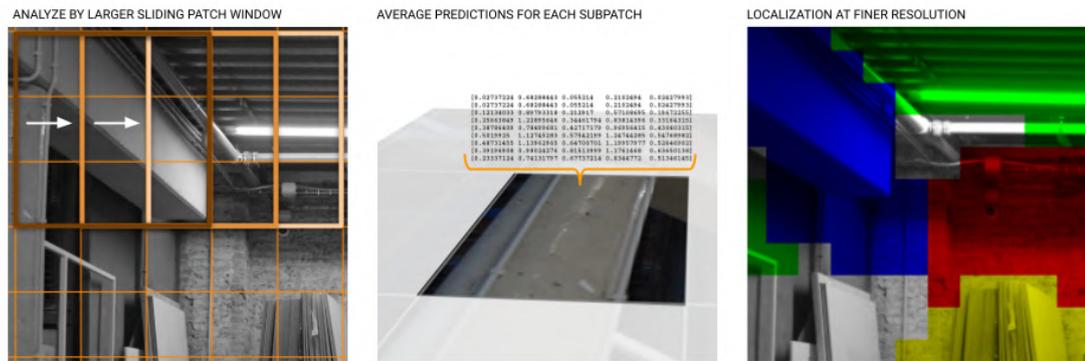


Figure 2.18: Arrangement of classification patches and final classification map

2.5 Synthesis

When the resulting geometric and localization data were obtained, they were combined to locate and collate the real locations and volumes of materials. This was accomplished by pushing the 2d localization imagery back into three dimensional space, either by photogrammetry with prepared inputs, or reverse raycasting.

2.5.1 Point Matching

Point Matching relied on creating new colorized meshes from the localization maps. After classification, each image was colorized with its output map, and this imagery was then run through the same photogrammetry process. This created a colorized point cloud occupying the same space as the previously constructed geometry.

The geometry of each element was then subdivided to create vertices at roughly 10cm intervals. Each vertex would then find the nearest point in the cloud, and record its material id color. An element's final classification would then be determined by the class with the most associated vertices.

This method suffered issues in cases where an element was classified differently from different angles, reducing the quality of the photogrammetric outcome. Additionally, this

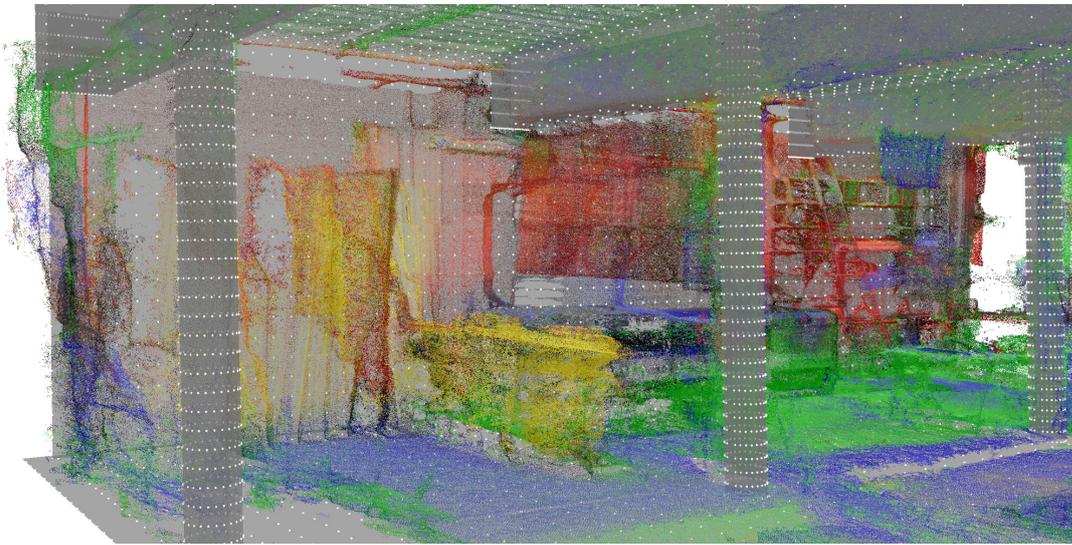


Figure 2.19: Subdivided reconstructed geometry vertices (white) overlaid with point cloud created from classification maps

disagreement would affect the final outcome somewhat arbitrarily, in terms of which point cloud points happened to be close to the subdivided points, rather than a true comparison of classifications from different angles. Finally, it lost the direct correlation between the images and the (ideally) correct BIM representations.

2.5.2 Reverse Raycasting

As the point-matching system suffered in its method to bring the data into 3d space, we instead integrated the already-known camera geometry and image capture locations. The position and orientation of the camera was known for all localized imagery, in our case as an output of the photogrammetry process, although camera localization techniques exist if captured separately, or stored inherently when exporting synthetic renders of a cloud. Additionally, a real or synthetic camera will also export information about its focal length and 'film' dimensions. These two sets of data are encoded in an extrinsic matrix (position and orientation) and intrinsic matrix (perspective information). By using the intrinsic matrix to define a grid of points in space, and transforming them by the inverse of the extrinsic matrix, each pixel in a mapped image was represented by a ray in space moving out from the camera. Collision for this ray was calculated for each piece of geometry in the scene - the closest collision is assumed to be the real point in space that pixel was looking at. This collision detection also returned the UV coordinates of the collision on the geometry, thus each ray met a specific location on the objects UV map, which would be colored accordingly in the stored texture. Element material assumptions were calculated by totalling the points of each mapped-color associated with each piece of geometry. The quality of this techniques is inherently tied to the accuracy of the final geometric representation, as a missing or incorrectly placed element may block or let

through a particular ray from reaching its correct location.

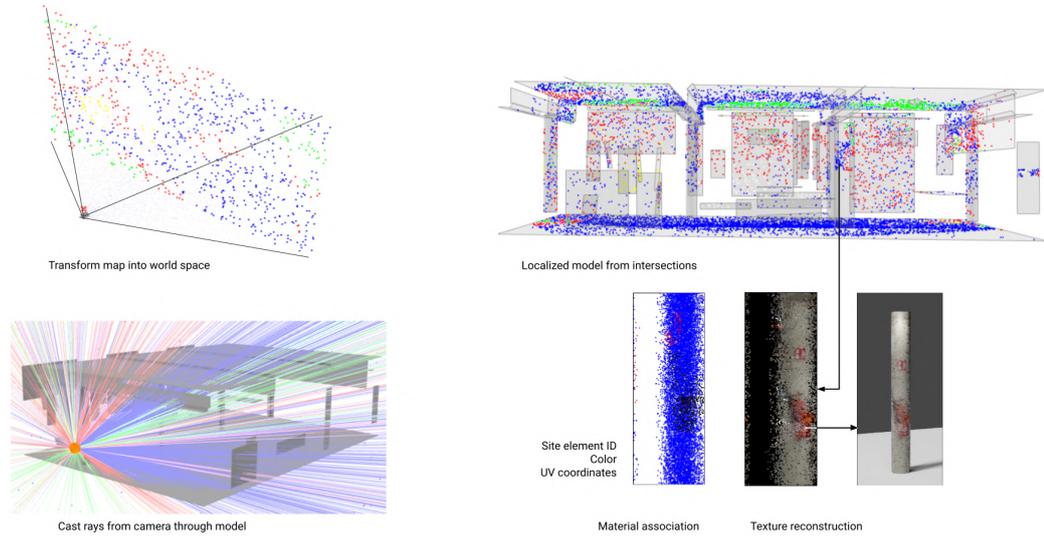


Figure 2.20: Overview of the process to combine classification and reconstruction via raycasting

2.6 Building Dataframe and User Interface

This data is stored in a MySQL database running through AWS. Each detected element is stored via two ideas, being the UUID of its associated site, and its own UUID within the site. From here, in addition to its BIM type, each element stores its position and dimensions, which can then be directly mapped to a transformation matrix once read into a viewing software. All of the existing BIM types scanned for can be represented in this way, as they can all be represented as rotations and scalings on planes, boxes, and cylinders. In the long run, more complex mesh or BREP types may need to be included.

To test the storage and retrieval, a simple web-viewer was built to view scanned sites. It was built on the p5js framework. The viewer shows a 3d representation of the detected BIM objects, and the user can hide certain types or materials. Selecting each element shows information about its material prediction and sie estimations.

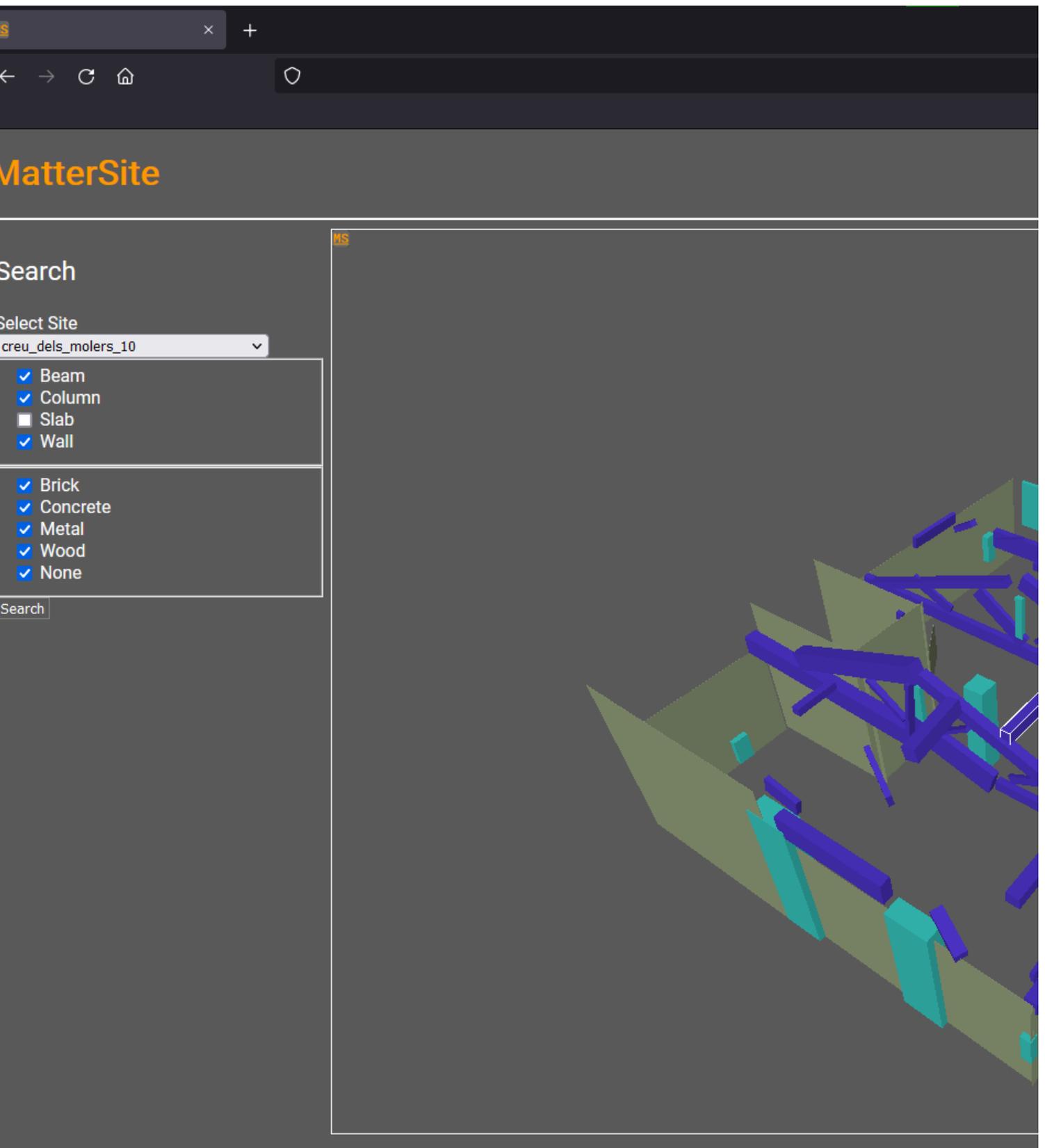
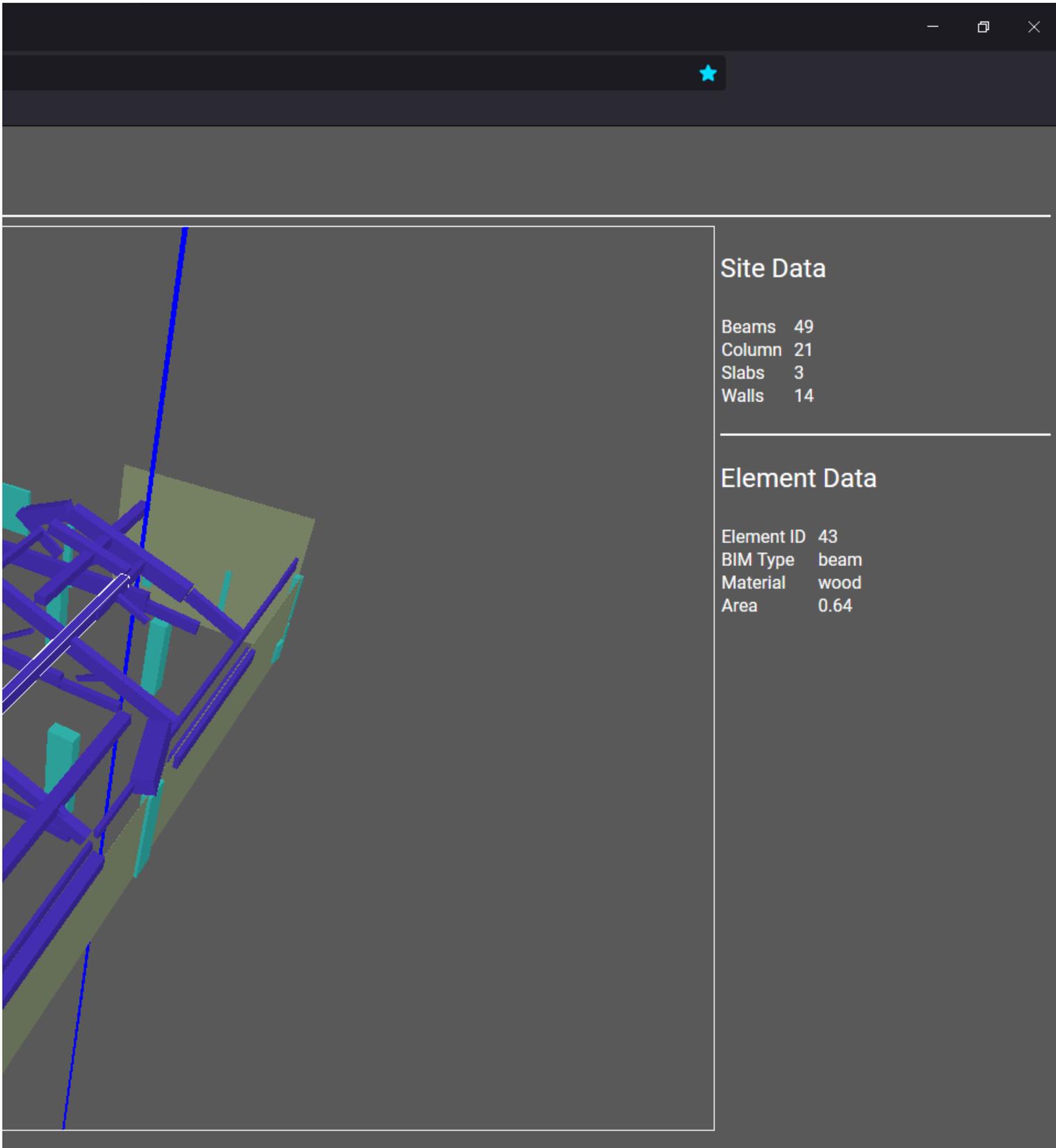


Figure 2.21: Screenshot of the building-site webviewer running on firefox



Site Data

Beams 49
Column 21
Slabs 3
Walls 14

Element Data

Element ID 43
BIM Type beam
Material wood
Area 0.64

Chapter 3

Methods for Post-Demolition Analysis

While working with elements after recovery may pose many of the same general questions as analysis before demolition (understanding what we have, its form, and its quality), the precise priorities and techniques varied greatly upon development. Monolithic and inter-connecting reconstructions are replaced with reconstructing many small elements with a high throughput efficiency. Likewise, accuracy falling within the margins of error for general procurement is replaced for instance by needing to understand the exact displacement along three meters of a wooden beam. To understand the specifics of structural details, material identification must also become much more precise than general 'wood' or 'concrete' descriptions.

3.1 Element Scanning Methods

As will be outlined in Design and Fabrication(Section 4.1), post-demolition workflows were developed primarily for the test case of wooden elements from a deconstructed pavilion. Thus, the methods developed were tied specifically to the shape and scale of the elements recovered. Different methods will be more or less adaptable across different material types, for instance tagging can likely apply equally, whereas scanning methods may vary greatly by scale.

Automated RGBD Camera

A system was developed to automatically scan building-framing-scale timber of up to 1.5m. This employed a Universal UR10 robot and an Intel Realsense RGBD camera. In this setup, timber elements were placed in custom holders, and static captures were taken with the RGBD camera along the length of the element. As these were captured at known (robotically aligned) positions, the resulting clouds could be immediately aligned without the need for other registration techniques. Although this technique was proven for the data as captured, the resolution of the RealSense hardware was insufficient for

the methods of analysis developed further in the process.

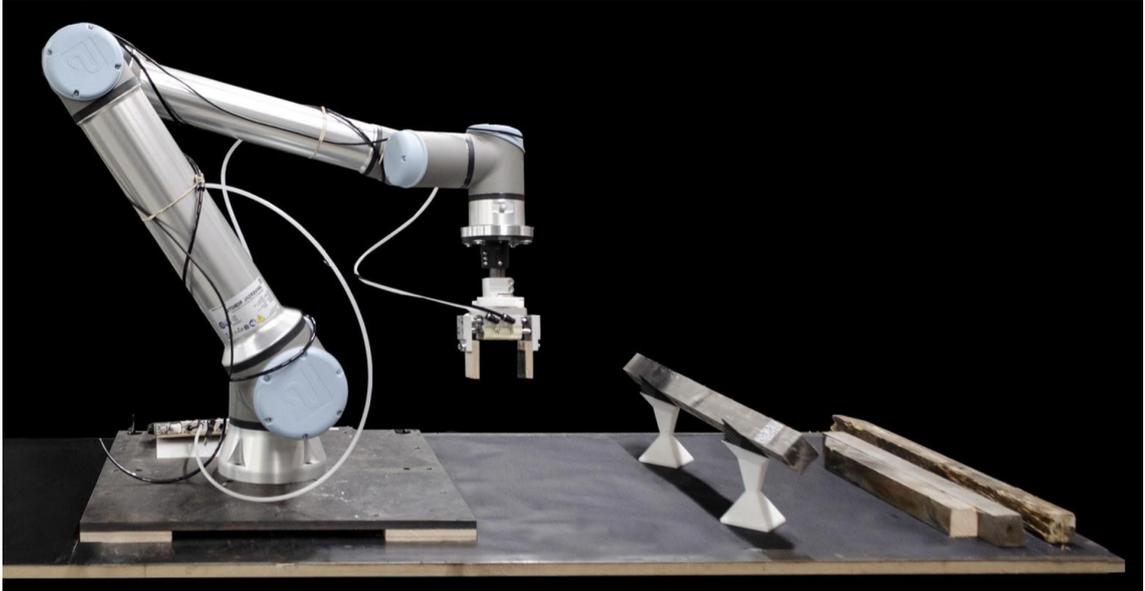


Figure 3.1: The UR10 with combination gripper-scanner tool, with the scanning holder

Photogrammetry

To gain additional resolution, we also used a photogrammetry based process. Each element was mounted upright and captured with roughly 30 photos. This produced point clouds of roughly one million points.

3.2 Geometric Reconstruction

The chosen photogrammetry solution (Metashape), contained features to export meshes as well as point clouds. These meshes were generally topologically unstructured, and aimed foremost to retain local detail. In order to obtain more efficient meshes for storage and representation, the meshes were run through a quad-remesh tool using Grasshopper Cockroach, retaining just 3% of the original mesh face count. This process also recalculated the correct UV and textural setup. These more rectilinear meshes were also used to extract final face textures, by re-rendering them while aligned to an orthographic camera.

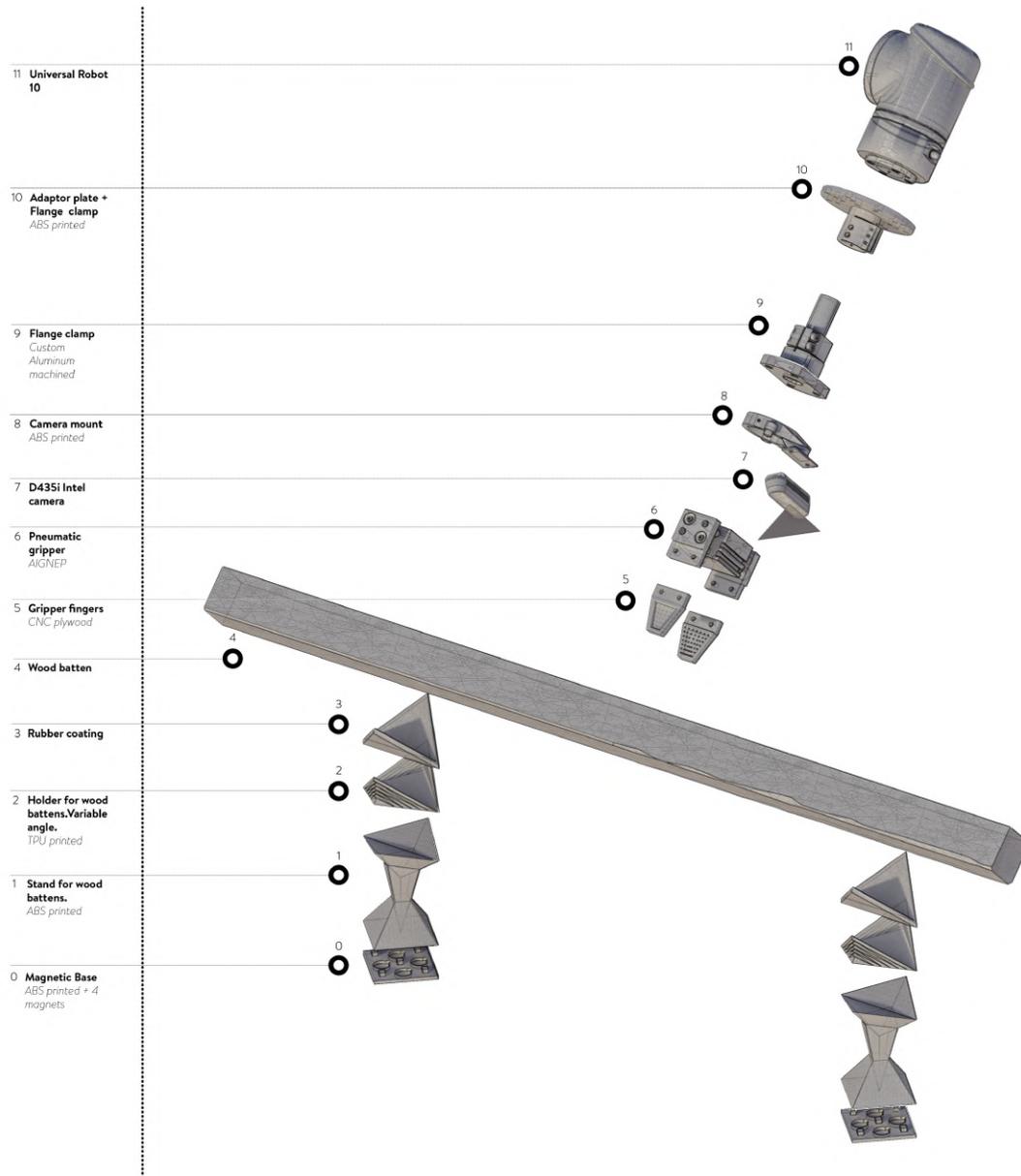


Figure 3.2: The complete set of parts for the gripper/camera assembly, as well as the holders designed to work with the scale of recovered element

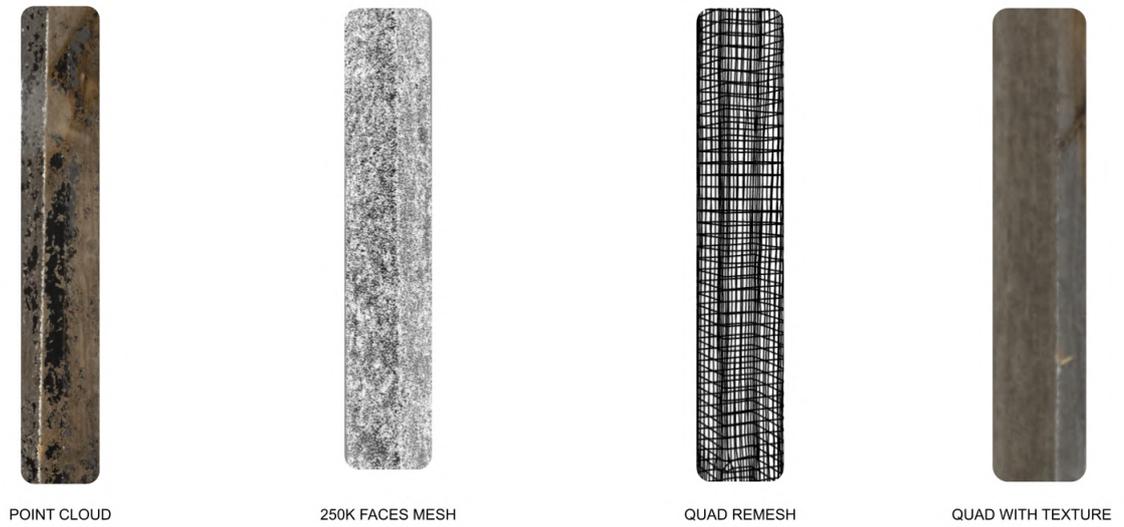


Figure 3.3: Process for a usable mesh

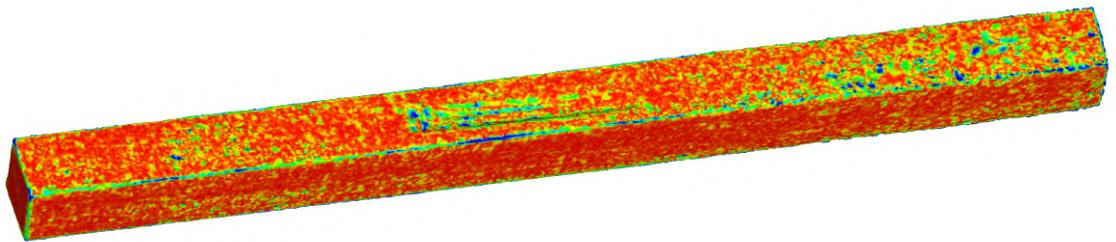


Figure 3.4: Visualization of the difference between the original mesh and quad remesh using Meshlabs Hausdorff Distance filter (Cignoni et al. 1998), on a standard element with mild surface damage. Red areas are close to the original shape, and blue areas further away; the greatest error is seen at the edges and at fine details in the surface damage.

3.3 Qualitative Element Analysis

3.3.1 Surface Curvature

The first qualitative measurement taken focused on local surface curvature, performed using cloud compare. Computed from local neighborhoods of points, this process primarily looks for decayed areas, while passing over deviations across the entire element. This curvature result was recorded as a Scalar Field attached to each point, and recorded as histograms. For an ideal piece of wood, the curvature histogram is a function of the dimensions and scan density of points, given that the corners will register as an expected degree of curvature. Thus comparing the resulting histogram values can inform the prevalence and depth of surface imperfections or decay. At this stage, these areas were not further localized, however this could be approached by finding the closest UV positions of original points on the quad mesh.

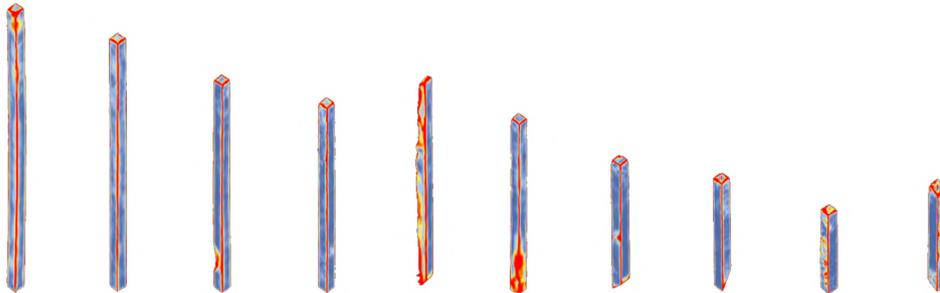


Figure 3.5: Examples of Curvature Analysis for Various Test Pieces

3.3.2 Mesh Skeleton Curvature

Larger-scale variations in the elements were analyzed using the Mesh Skeleton, from Grasshopper's NGon plugin. This reduces a mesh to a series of curves with associated normals, responding to the average positions as well as changing surface normals. This allowed for discovery of various types of warping across the entire piece, due to natural grain patterns and various types of exposure.

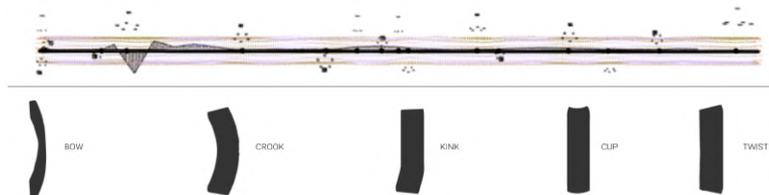


Figure 3.6: Types of Element Problems Identified by Skeleton Curvature

3.3.3 Textural Defect Detection

As not all issues are visible at the geometric scale, analysis was also performed on the image textures extracted in previous steps. The primary defects considered were knot holes and nail/metal connector holes, appearing visually similar and representing similar possible structural issues. Localization was performed using the Mask R-CNN algorithm (He et al. 2018). As this detection is performed on face-separated images, and Mask R-CNN returns the local centroid of each detected region, each detected defect was ultimately stored as a vector of its face id and local UV position.



Figure 3.7: Examples of the range of contexts and features present in our test set.



Figure 3.8: Example output when localizing texture defects using Mask R-CNN

3.3.4 Textural Evaluation

Once extracted the surface textures were useful beyond structural analysis, such as for design and aesthetic concerns. Each element face may undergo a broad variation in weathering due to original built use, orientation, human use, loading conditions, climate, local location, and post-processing marks. Even within a single project and material this can lead to a wide array of resulting visual textures, which a designer may wish to incorporate or downplay. As one means of encoding these factors in a searchable way, a simple scoring system was developed to get an overview of the textural variation present in each piece. For each face, the average saturation and luminance value was calculated. From here, the percentage of pixels beyond a certain threshold from this value was extracted. This gave each piece a comparable value for both its rough visual appearance and the degree of visual surface variation.

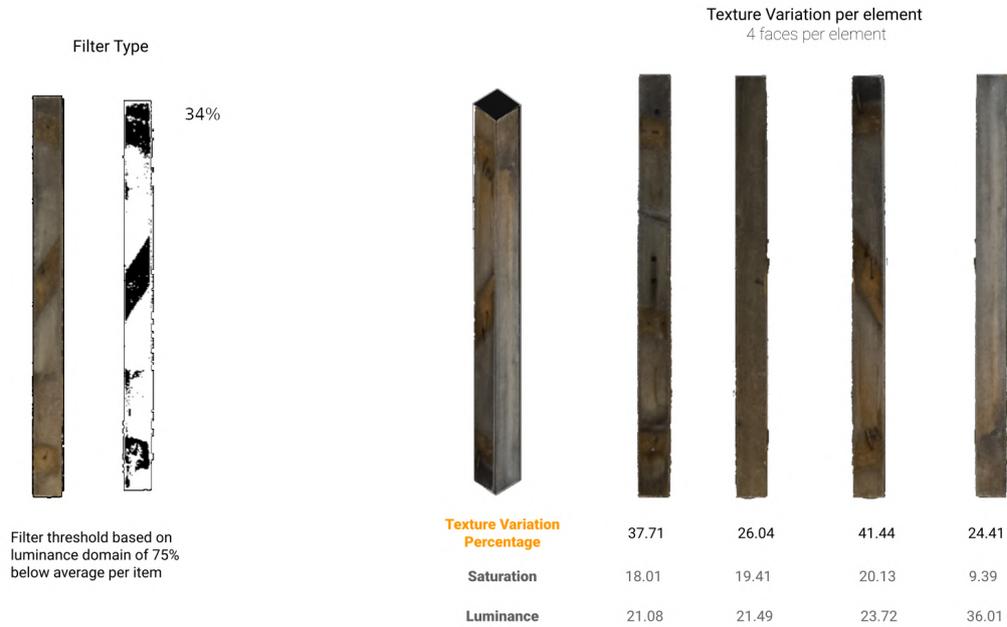


Figure 3.9: Textural variation percentages for an example wood piece



Figure 3.10: A selection of the surface textures present in our test set

3.3.5 Mass Comparison

All described methods have thus far depended entirely on surface details. While in industry there are various methods for non- or minimally-destructive sub-surface analysis, these were found to be beyond the scope of this investigation. However, mass analysis allows for extracting some information about the element interior, with minimal additional tooling.

First the species of wood is determined, either from records or predictions based on the building age and location. Given the dimensions of the element as measured in previous steps and the average density for the species, an expected mass is calculated for the element. Simultaneously, the actual mass is measured, either manually or via torque-sensors in automation equipment (e.g. the Universal UR10 line). From here a direct ratio was calculated, indicating how much decay may have occurred over time. As an example, the worst decayed test pieces ended with only 55% the expected mass.

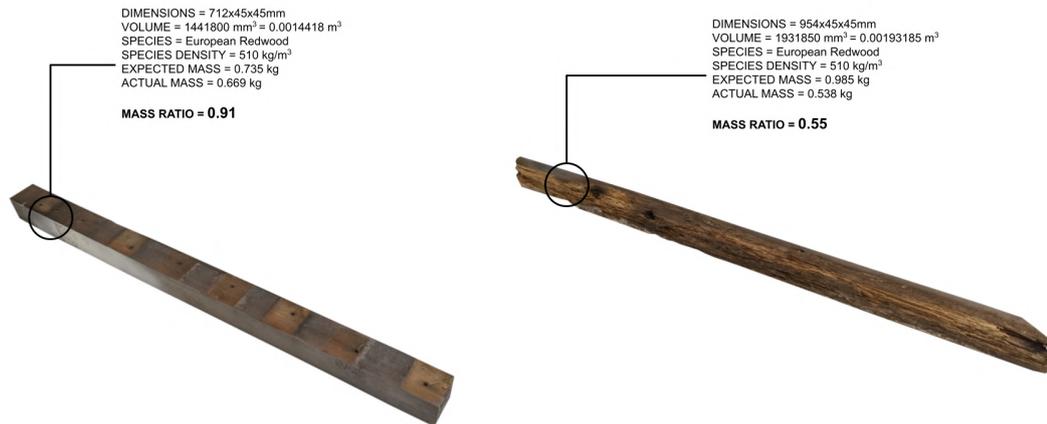


Figure 3.11: Example mass ratios for structurally-usable vs a highly decayed wooden element

3.4 Element Tagging, Lifespan Tracking and Interface

Through these methods, each recovered piece received a dataframe of extracted geometric, structural, and textural information useful for new design and beyond. As with the pre-demolition building elements, this information must be stored efficiently and effectively associated with the specific materials over their time of storage. Given the high degree of individual variation, it was decided to store this information in an external database rather than attempt to encode it within the element itself. A MySQL database was chosen, running on Amazon RDS. The database structure was based on a simple primary table describing the elements rough dimensions and origin site, with additional associated tables storing information from each analysis method. These tables were all associated by

the element's primary UUID (universally unique identifier), which was also used as the primary reference in other software. The UUID is a 128-bit number, generally represented as 32 hexadecimal digits, thus only these 32 digits needed to be stored on the element itself after processing. For this stage, QR codes were chosen for storing the id, given their built-in redundancy features, negligible cost of application and ubiquity of available software libraries for reading them. Each tag also contained a human readable version of the 32-character UUID. However, the method of application contained possible issues.

Several methods were tested for applying QR codes to recovered elements:

- Ink Transfer : Tags printed via laser printer were transferred to the element using acetone
- Laser Etching : The etching mode of a commercial laser printer created the black and white image as a darkened layer
- Paper Tags : Normal paper tags were adhered to the element and covered with a thin layer of glue for protection



Figure 3.12: Test pieces using ink transfer, laser etching, and paper tags

To test ways this tracking system would be accessed on site, a mobile app was developed for on-site data-lookup. The app scanned the QR codes of chosen elements and retrieved a portion of the information from the MySQL database.

Figure 3.13: Screenshot of the QR reader running on Android, showing the information returned from the database



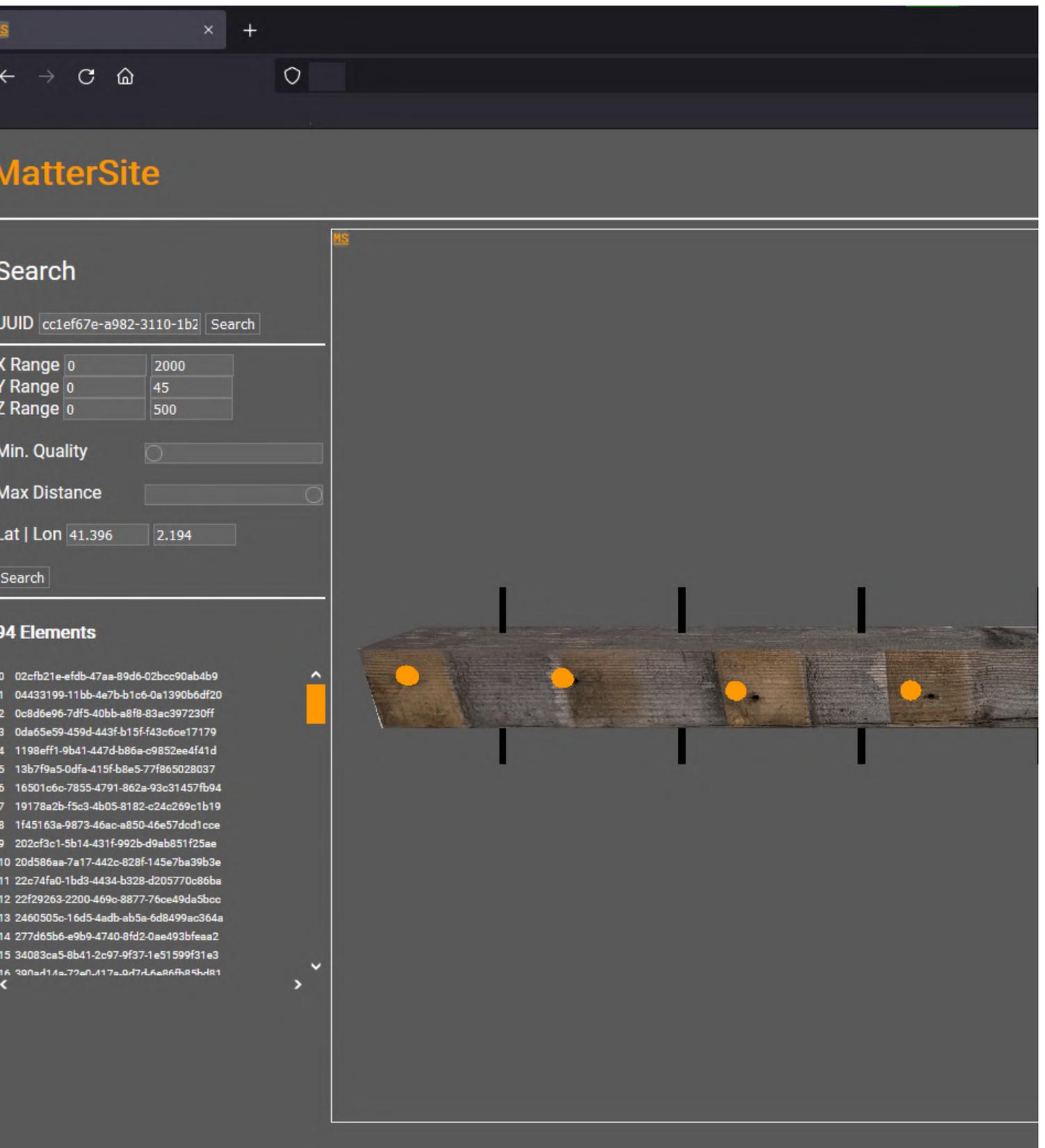
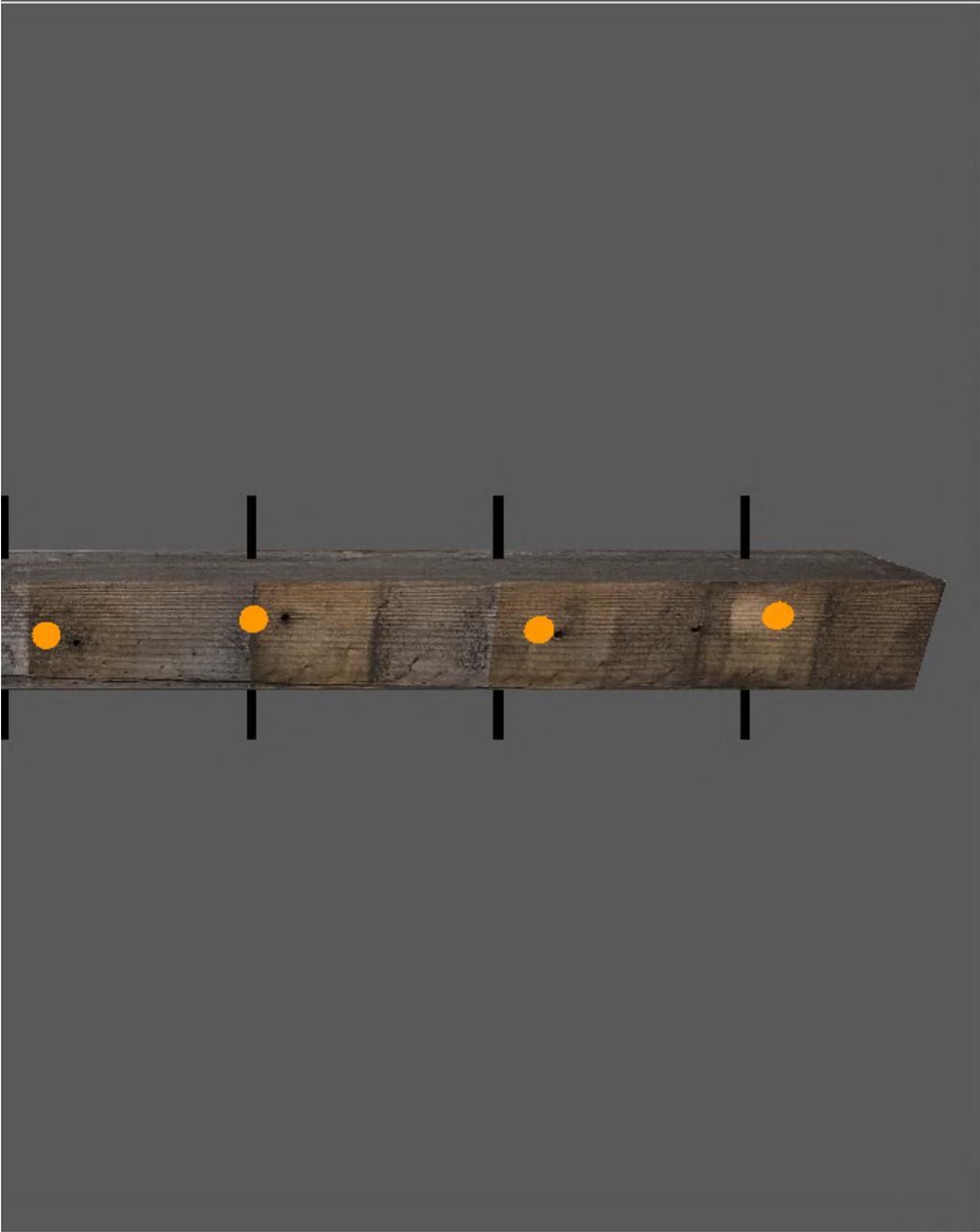


Figure 3.14: The prototype viewing interface for recovered elements running on Firefox



Data

| | |
|--------------|--------------------------------------|
| uuid | bba3a71b-9ede-ec24-15bb-50d4929ede90 |
| x | 712 |
| y | 45 |
| z | 45 |
| quality | 73.24 |
| species | pine |
| lat | 41.6 |
| lon | 2.28 |
| mass | 669 |
| mass ratio | 0.91 |
| defects | 16 |
| show defects | <input checked="" type="checkbox"/> |

Chapter 4

Design and Fabrication Proof of Concept

While the many methods of analysis can improve logistics and throughput, the ultimate lynchpin of the process depends on how these materials can be incorporated at scale into new construction work. While this work has been arranged following the timeline of a single piece of material, development was largely continuous across all parts. The needs and specifications of the design and fabrication stage informed many of the datapoints that were ultimately measured. Developing this section required a set of materials ready to be run through the entire existing-building to new-construction process, at a scale and quantity workable by a small team.

4.1 Test Site Materials

Digital Urban Orchard Pavilion

The Digital Urban Orchard was built by the IaaC Open Thesis Fabrication class of 2016. Built of European Redwood, the consisted largely of small-cross-section (40x40mm) wooden elements of variable length, between 30cm and 2m. Scanned using Leica hardware shortly after construction, it was demolished in 2021. Siripurapu et al. (2016) The scan data was used to test reconstruction techniques and judge the scale of elements to be worked with. Additionally, the recovered materials were used to test the element analysis workflows outlined in chapter 3. The pavilion was outdoors for its life span, in Barcelona's low-rain Mediterranean climate. Thus while there was minimal moisture damage, the components sustained noticeable solar bleaching and thermal cycling. While the pavilion also had a flooring structure made of plywood; this proved to be too decayed to be properly digitized or re-used using our system.

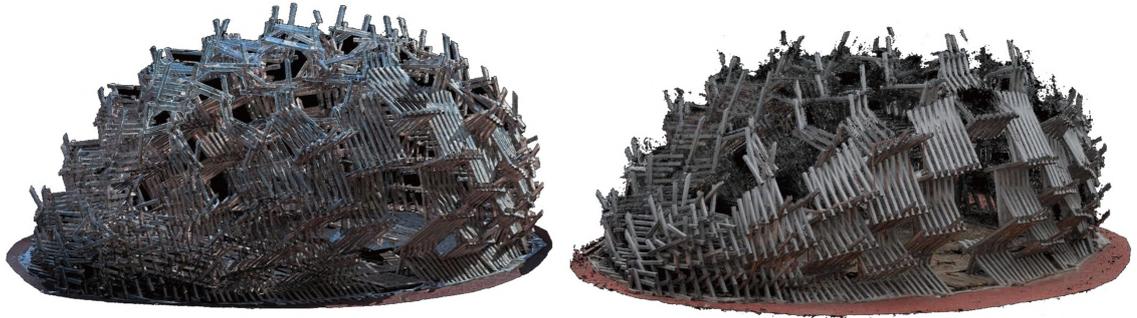


Figure 4.1: The original scan of the pavilion after construction, and a photostan taken before demolition, showing the settling that occurred over time

4.2 Demolition and Processing Procedure

The degree of integration of reclamation principles during the deconstruction process itself is critical to the material state and degree of post processing needed for the materials. This test site followed one likely scenario for this relationship - the deconstruction was carried out by a third party, leaving the materials on site and without undue damage to materials, however without specific considerations for reuse. In an ideal scenario, operators with re-use knowledge would be either on-site as advisors, or carry out the deconstruction themselves. Thus, our initial processing covered three main steps. Firstly, it was found that the structure was largely held together with adhesive-coated spiral-shank nails. During deconstruction, these were generally left in the nail-side piece and hammered flat for storage. These had to be individually pulled or drilled out. While on average an element could be processed in about one minute, any connectors that initially broke during extraction slowed the process down considerably.

Additionally, some connections were made with wood screws. While these had suffered less corrosion and were ostensibly easier to remove, these also had some bending which often prevented easy removal. Finally for those connections which proved impossible to remove, the piece had to be manually cut down to a size to exclude this connector. Across the 200 pieces processed, only six connectors were unable to be removed, and only eight elements were judged too decayed to be used at all (although the matching system would later make its own quality judgements). In a fully automated system, given the nail-finder algorithm described in 3.3.3, and new possible decay-finding systems, these could be left whole during processing, and the 'bad' length properly avoided during fabrication.



Figure 4.2: An unprocessed recovered element showing remaining decayed plywood base and wood screw connector

4.3 Design System Overview

For the design of the demonstrator, we chose to highlight the goals of design and material adaptability that digitization brings to the process, in the form of a small-scale shelter pavilion. While more practical buildings will contain a slightly different set of concerns, this work demonstrates the flexibility of geometry that can now be incorporated in those designs due to our system. The design was based on approximating arbitrary input surfaces using two layers of opposingly oriented pieces. Development of this system took initial inspiration from the overlapping geometry of reciprocal structures, although this orientation does not utilize the same structural ideas.

Given a starting surface, it was first converted into a mesh using a stretched hexagon tiling method. This was then simplified into a series of diamond strips. These edges were then turned into initial solid 'reciprocal'-layout forms using the Grasshopper NGon plugin. These pieces were divided into two layers based on their orientation in the original surface's UV space. Each piece is then given a minimum scaling to overlap with its neighbors, and offset from the surface by a calibrated amount. Thus the two layers are then connected by a series of specifically cut connector pieces.

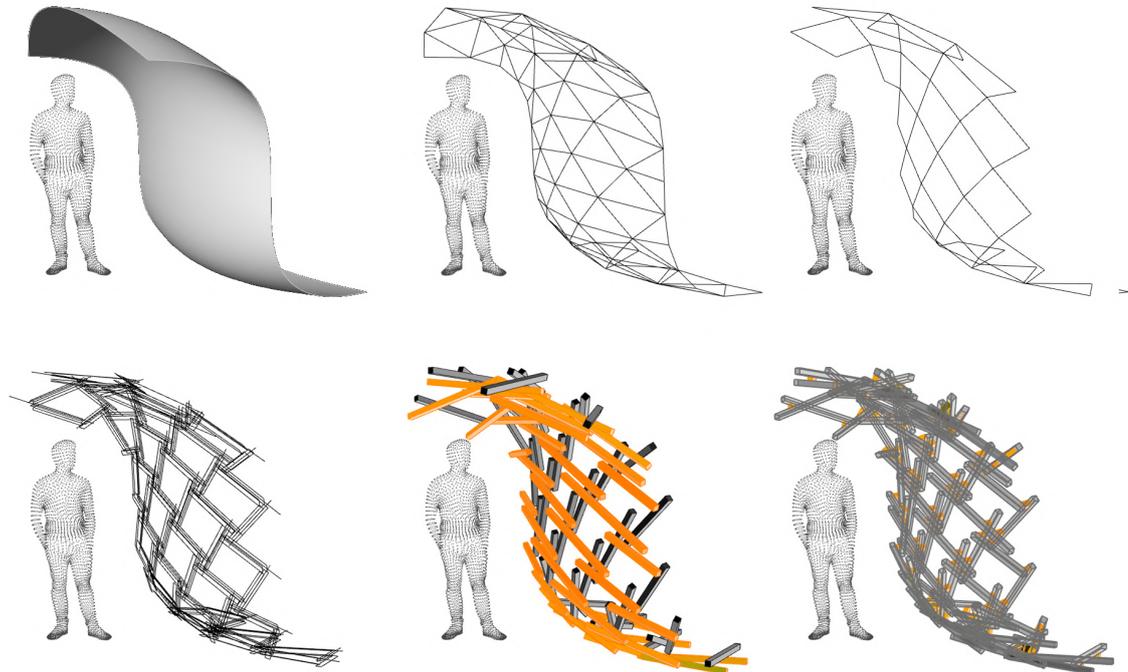
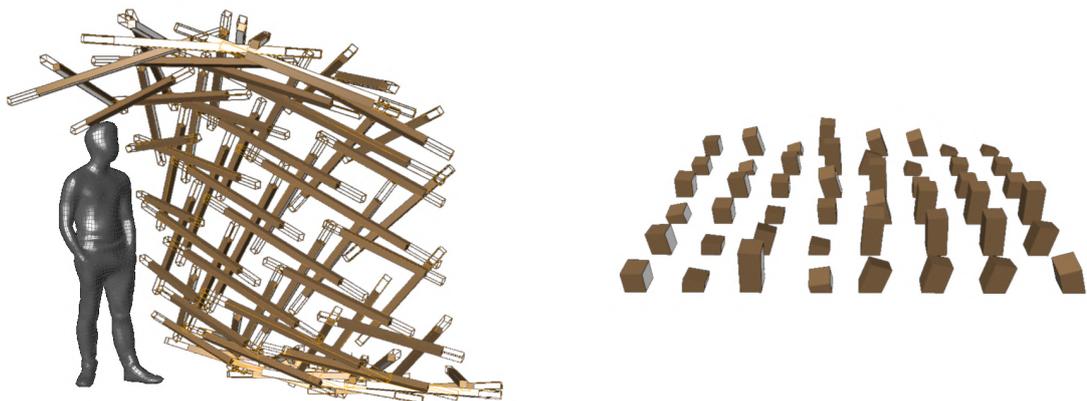


Figure 4.3: Stages from an input surface to a two-layer timber structure



Element dimensions have high flexibility once bridging connection points

Design information and specificity is embedded in the shape of the connectors between layers

Figure 4.4: Material Usage

Matching Algorithm

With this strategy for building arbitrary surfaces from straight wooden elements, it then needed to be realized with the actual materials available. With material-buyer matching already a common process for reclaimed material marketplaces, a system was developed for matching each piece to be assembled with one from the database.

This was based on a top-down greedy matching system, with the largest elements in the design passing through the database first. Each element had a degree of tolerance where it could possibly match with elements larger than designed, but never with elements shorter. As the database is arbitrarily large, the list of elements to be matched against comes from an initial reduced database query, immediately filtering to relevant items based on dimensions, quality, or distance from storage. Particularly here, we rejected all elements below a certain quality threshold as calculated in Section 3.3.

Each design element thus had three possible outcomes; either it was well matched with a real piece within tolerance, it was matched with a significantly larger piece that would need to be cut down, or no match was found and would require fulfillment with new stock.

Each matching pass could then be assigned a score, with the amount of new stock as well as new labour required negatively impacting viability of a particular design option given the available materials. In a larger scenario, the transport distance of each item would also serve as a negative variable.

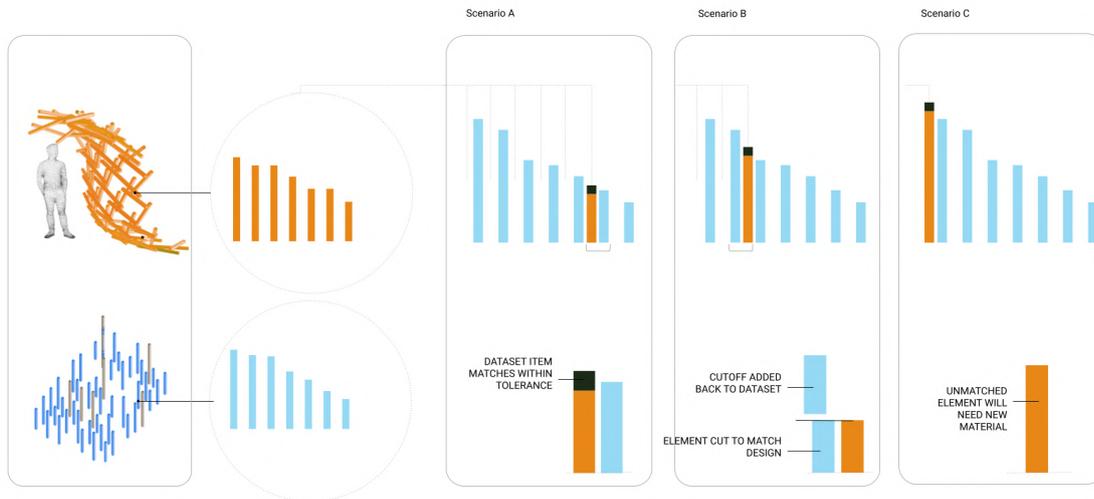


Figure 4.5: Outcomes from a design piece matching itself against the database

Design Optimization

This method of scoring a design was then used to optimize the design through, using the multi-goal evolutionary solver system Octopus in Grasshopper. Although a unique problem for every design, adding continuous variables for solving to this design system was straightforward, as the lengths, S-curvature, and overall bend of the surface could all be controlled directly. Conversely, the size and orientation of the diagrid applied to the surface was also controlled, though within tighter bounds.

Controlling these variables, the system arrived at a design that would use the most stored materials with the least amount of additional manual processing.

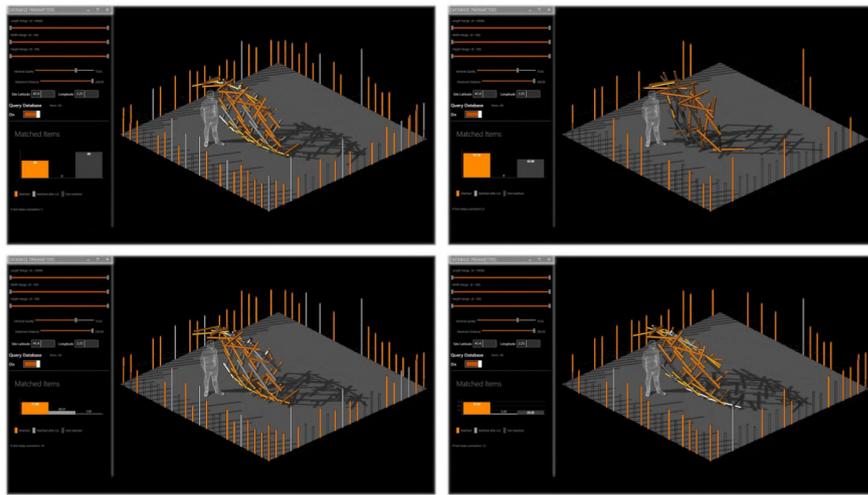


Figure 4.6: Design iterations considered by the solver, showing the design interface and a selection of the database elements well matched

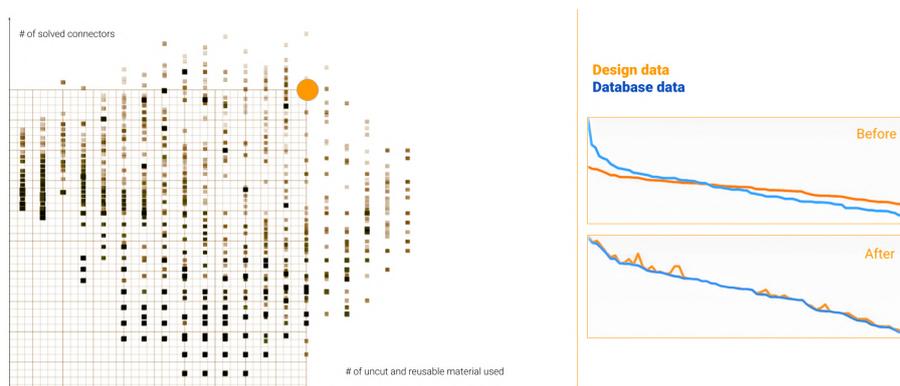


Figure 4.7: Diagram of iterations with results, as well as histograms of dimensions in the currently considered design and database lists

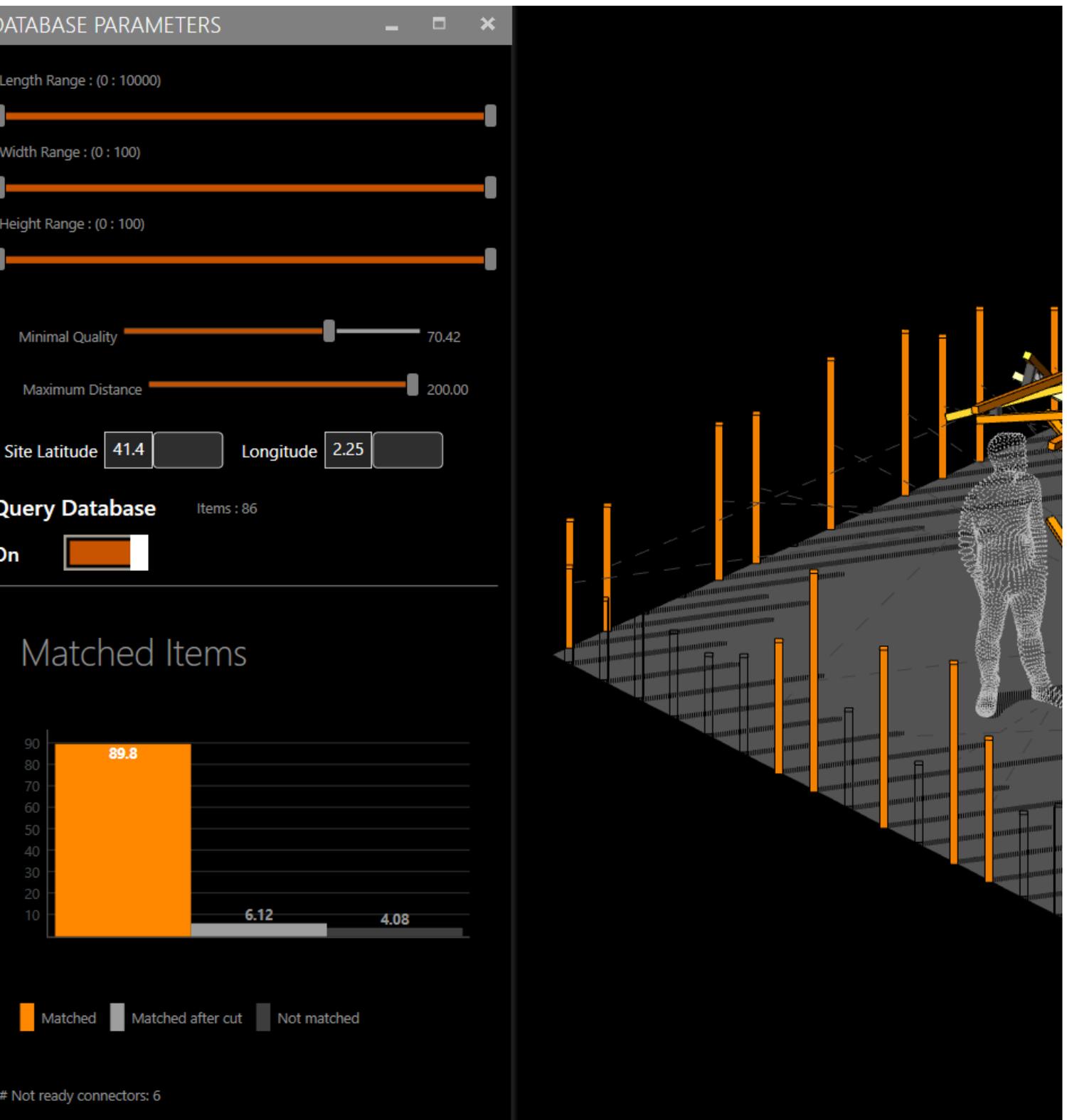
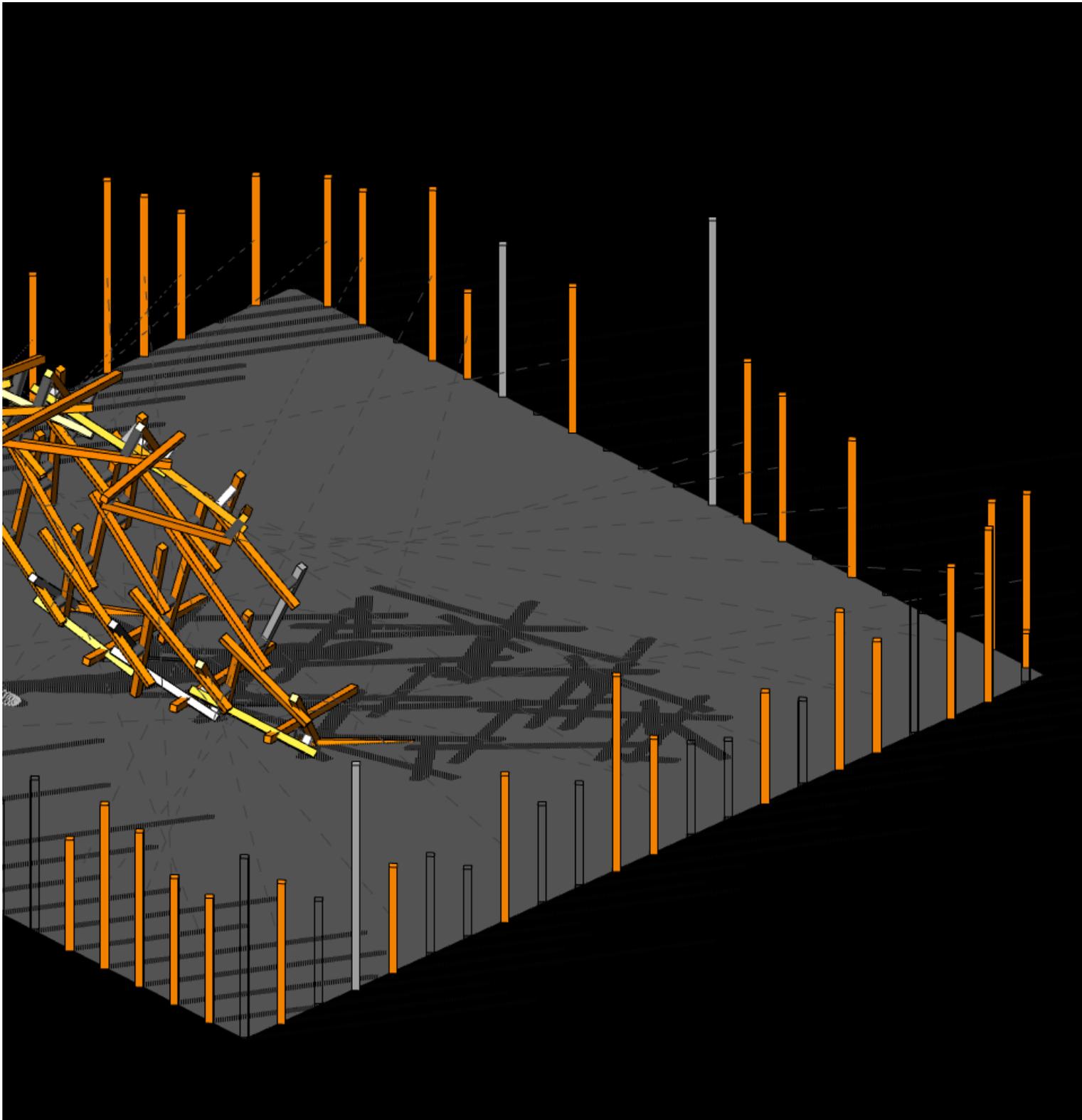


Figure 4.8: Design Interface



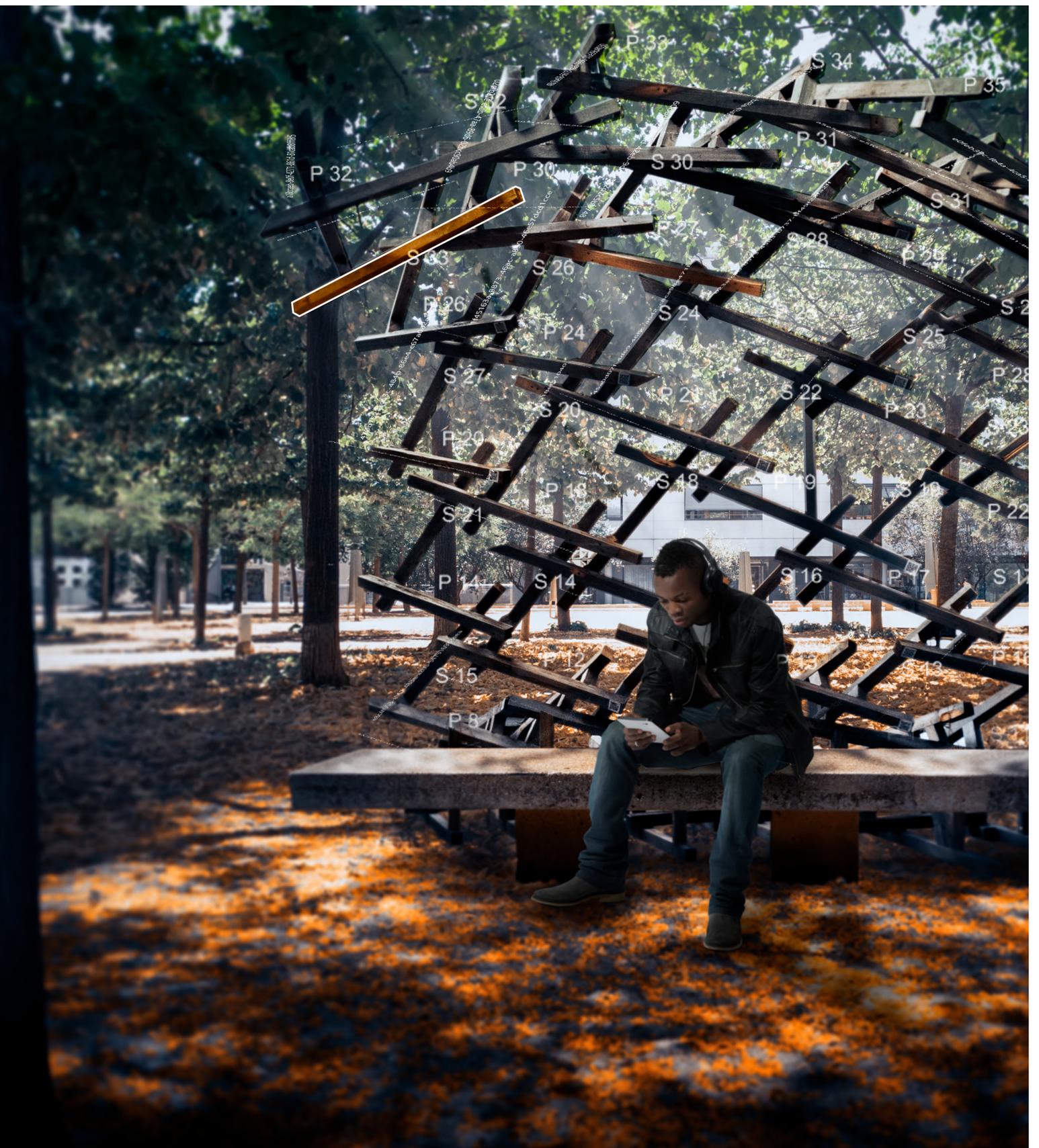


Figure 4.9: Design Render



ELEMENT DATA

Wood Battens

Mass Ratio: 0.73

Usage: 7 years

Warping: 0.96 %

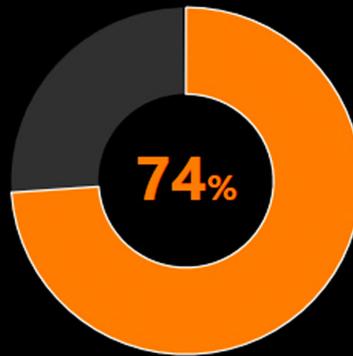
Nail Density: 4

Surface quality: 84.88 %

Profile:



Material Health



Health



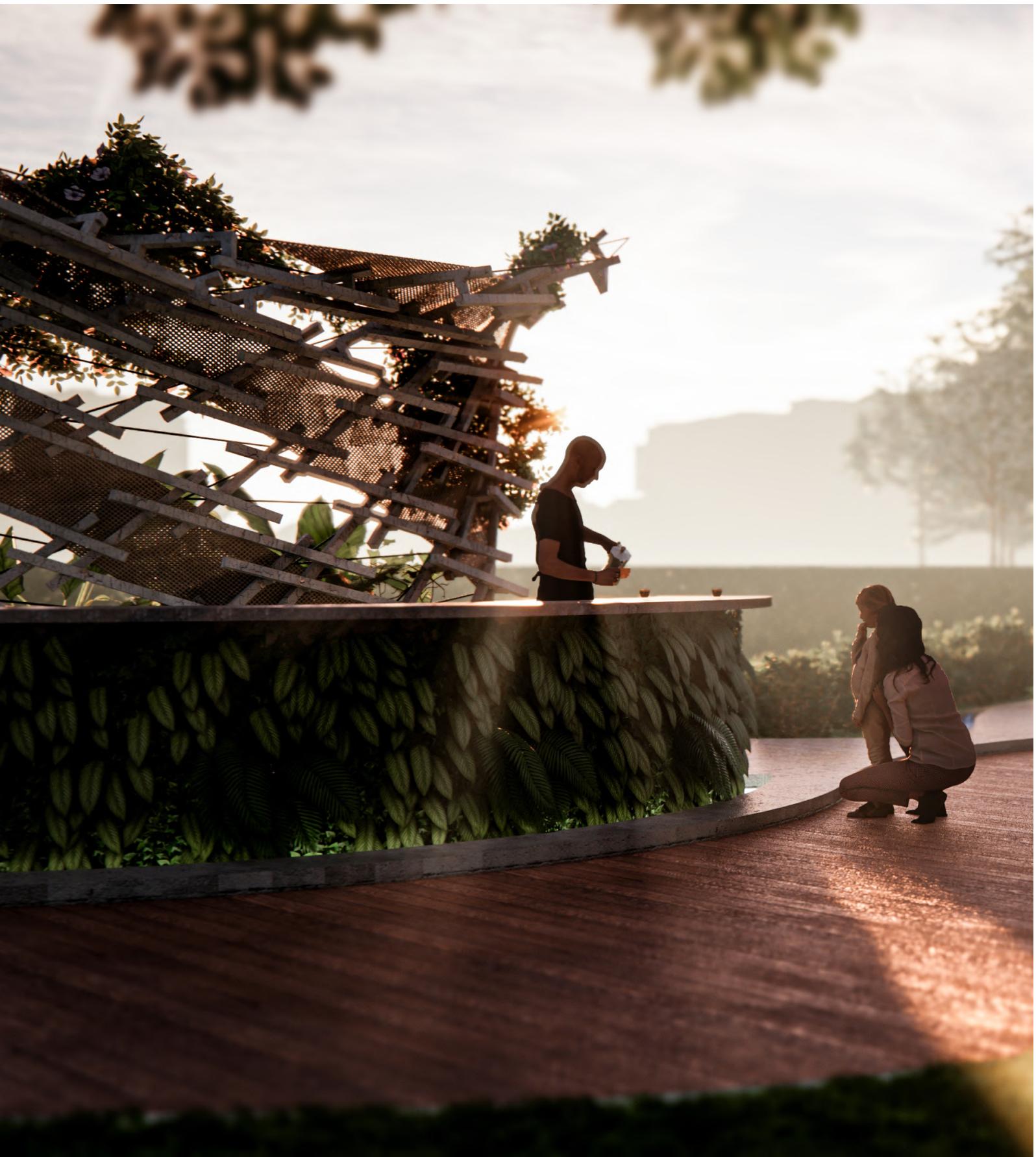


Figure 4.10: Design Render



4.4 Digital Fabrication Methods

Following this design strategy, most of the pre-assembly work was encapsulated in the connector pieces, as they contained most of the essential measurements of the design. A fabrication system was chosen, such that theoretically all necessary operations could be carried out by a human using normal woodworking tools, while the robotic presence could be easily CAM-programmable, would increase speed and safety, and ensure accuracy. This was accomplished by encoding the connection vectors in a specific method in each connector.

Projection Assisted Annotation

While each structure component contained a degree of variability in its total length, the intersection and connection points within its length were highly specific. This information was applied directly on to each component using a projector system. The projector was mounted on a metal frame at a height roughly 2 meters above the work surface, and calibrated to display a known pixel-to-mm ratio at the height of the work surface plus the component height.

The projection displayed the minimal size for each component, the maximum dimensions within the threshold, and the location, id, and orientation of each connector piece. The connector locations were traced in for all primary-layer components, and each connector was additionally pre-glued in place for each secondary-layer component.

Each layer component was assigned an incrementing id based on its layer (e.g. P2 or S5). Each connector piece could then be uniquely identified by which layer components it attached to, (e.g. 2|5). Each connection point on a layer piece was also identified by the id of the opposite layer piece it connected to. See 4.4 regarding the outcome of this.

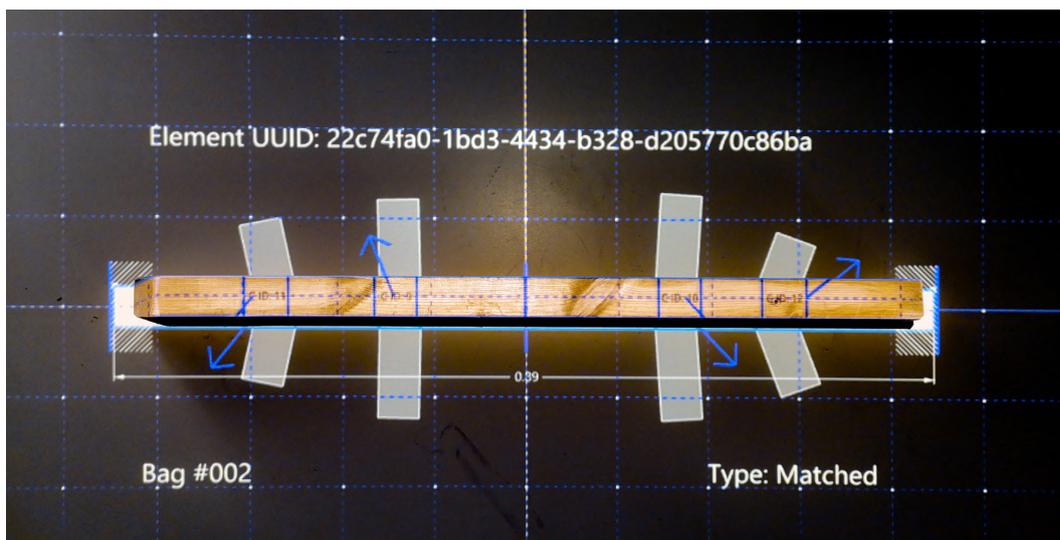


Figure 4.11: Rhino-based overlay for each piece



Figure 4.12: Pre-glueing connectors for all secondary-layer structure components



Figure 4.13: Stock of pre-assembled secondary-layer components

Connector Design

When designing connectors, each intersection was initially approached using the center-lines of each layer piece (see Figure 4.14). The closest point on each line to the other was calculated, finding the endpoints of an ideal/naive connector. Where this line intersected the outer face of the element, the angle to be cut in the connector could be calculated. However, this meant that the face of each connector was arbitrarily oriented compared to the normal of the connector's center. While this could be programmed for with CNC tools, it created a less predictable workcell, and was more difficult to automatically detect and account for possible collisions. It would also have been much more difficult to produce by hand, which remained a desirable design constraint. Alternatively, connections could be designed by first only considering the two faces to be connected. Orienting the geometry such that the plane of the primary layer lies on the world plane, the angular relationship could now be described by a single vector. This vector can be decomposed into two simple on-axis rotations, which were applied at each end of the connector (see Figure 4.14). Because of this however, the connector no longer followed the ideal-shortest path between the layers, and instead touched each main piece at a location to best fit with its end rotations. The connector's intersection point with the secondary layer was now defined by where it intersected a vertical plane projecting from the primary-layer piece. It then touched the primary-layer piece at the necessary location given the encoded angles.

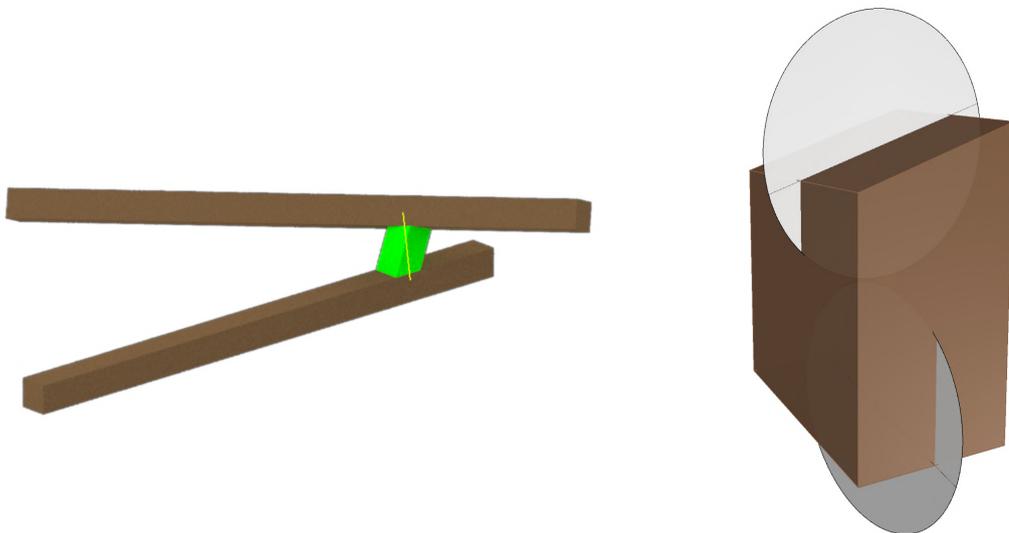


Figure 4.14: The faces of each connector only contain simple rotations to encode the overall vector being connected

Connector Fabrication

This simplified connector system allowed each piece to be fabricated by hand if necessary using a simple miter saw or adjustable miter box. To produce at scale however, we ran this process using a robotic workcell, consisting of a Kuka robotic arm with pneumatic grippers, and a stationary circular saw. Due to the methodology defining the connector's faces, the robot's approach for all pieces now existed in limited arcs, ensuring easier collision avoidance (see Figure 4.17).

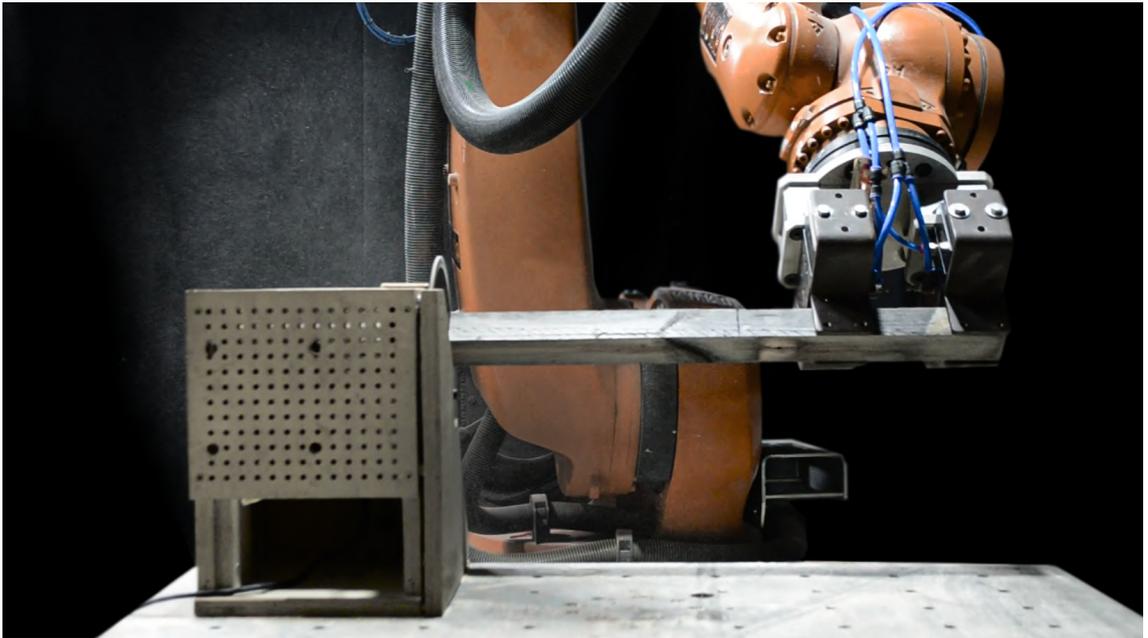


Figure 4.15: Connectors being cut using the robotic setup



Figure 4.16: Stock of connectors cut from a single input element, with minimal waste chips

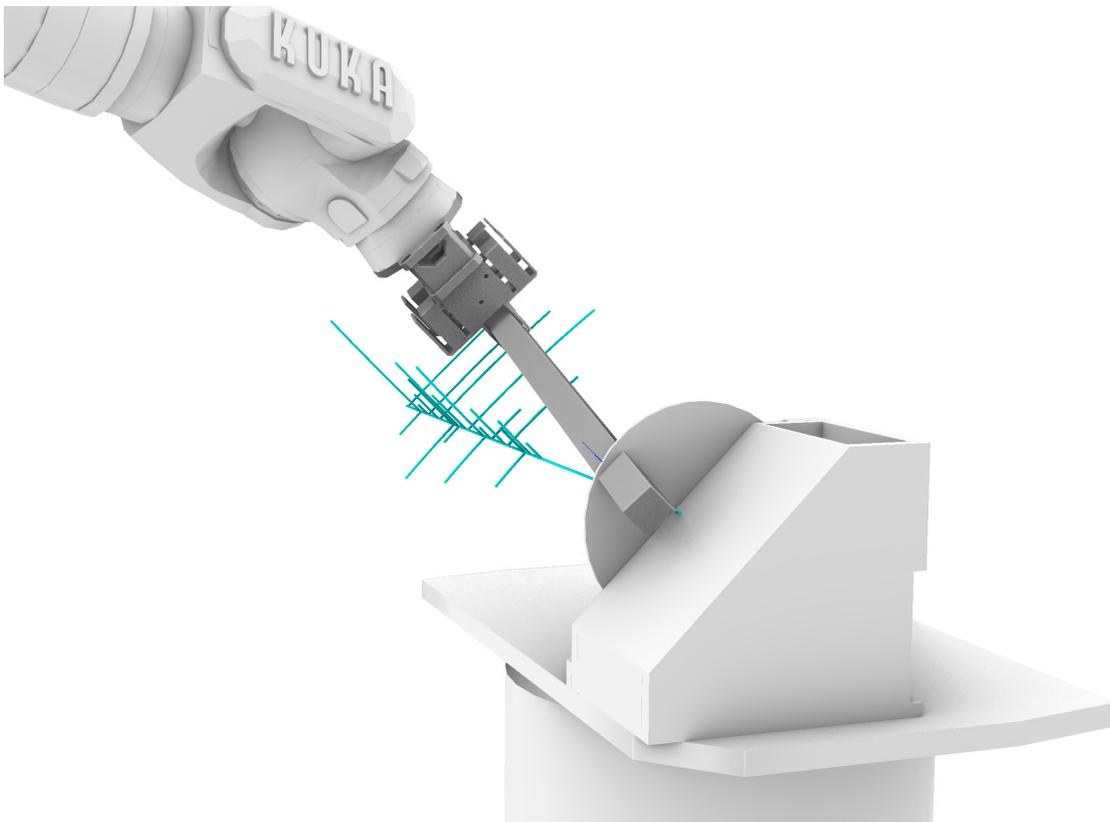


Figure 4.17: While cutting connectors, the robot's Tool Center Point will only approach from a limited series of arcs, reducing likelihood of collisions

Assembly

Final assembly was performed by hand, using wood screws for all connections. Starting from the bottom diamond required multiple workers and clamping to hold the slightly twisted quad in place. However from there, each layer of diamonds was added progressively, with the screws at the top of each diamond added last in order to pull tension into the system if necessary. Even with two screws at each layer-connector point, some bending did occur as more mass was added to the structure. However, this never became extreme, and remained within the bounds of being accounted for as assembly continued. At this stage, components could be added by a single worker if necessary.

Due to the density of labelling applied during the setup phase, upcoming pieces and their orientation could be found by reading the current raw edge of the structure, reducing the need to review against the computer model often.

For stability, a simple two-layer raft was built as a base to the structure. Due to how the reciprocal plugin created forms, the bottom pieces of the surface were not flat the ground, and would not have sat with stability. The base caught the slightly differing heights, as well as allowing the main vertical section of the surface to bear directly on



Figure 4.18: The initial diamond module, requiring the most manual labor

the ground rather than through the lower curve.

Due to rotation at the joints, for stability during transport and demonstration, cables were also added for the major curvature areas. These attached the top curve to the base, to handle bend in the lower curve, as well as between the longest straight ends of the upper curve.

After assembly, the final construction was digitized using photogrammetry techniques. This resulting point cloud was compared against the original design model using Cloud-Compare, to determine the amount of error introduced (see Section 6.3).

Chapter 5

Business Case Investigation

The fabricated demonstrator tested the viability of the technology developed as it pertained to the physical resources of the circular economy. Concurrently with this, we also explored the necessary factors to allow the system to be applied and match the economics of the current marketplace.

This was based on interviews and discussion with relevant actors in the field, including in circular deconstruction, material sales, high-quality site digitization, construction economics, and the practices of contemporary demolition.

There were two primary ideas that resulted from these conversations. First, was considering the nature of the product this technology would represent and balancing detail vs time. The development of the technology focused foremost on obtaining highly detailed and highly accurate reconstructions of sites for analysis. However for some marketplaces, it was much more valuable to know rough estimates on a quick turnaround, to quickly connect with future buyers with minimal storage. While useful at many stages, throughout we proposed the initial deconstruction companies as the first clients of this technology.



Figure 5.1: An accurate but expensive mounted LIDAR model

Secondly was the constant underlying question of cost. The photogrammetry and mobile LIDAR technology tested was initially approached as a immediate and efficient method of approaching site digitization for rapid iteration, and there was an underlying assumption that modern tripod-LIDAR technology would take their place in a full product. However, considering both the avenues of direct purchase of LIDAR hardware (and accompanying training), and alternatively the cost of scanning using third party groups, it was found to be untenable. For this reason, the photogrammetry methodology became the main intended method. Additional domain-specific costs also include the ramp up of computation and storage. Development was able to be performed on laptops, and only covered a few fully scanned and reconstructed sites. In a scenario with multiple full-sized buildings being processed a month, it quickly become necessary to offload computation onto more powerful hardware, or into AWS or other cloud-based ecosystems.



Figure 5.2: Common sources of early-costs

Chapter 6

Results

6.1 Comparison of Reality Capture Methods

For all measures of efficiency and accuracy, Metashape was found to be the best photogrammetry method. Execution time and resulting point counts are shown in table 6.1. For the Barcelona site, Metashape found at least 5 million additional points in its dense cloud, in at least 2 hours less time.

Additionally, Metashape was found to produce the least misalignment and false positives. While Metashape created one truss in error in site BCN1, it produced an accurate version of site IAAC1, both of which Meshroom and Colmap failed to align entirely. While with very heavy photo coverage Colmap could also produce usable results, its extremely long processing times prevented it from being a viable option.

Metashape sometimes required more manual work when processing; as some photos would fail to automatically align and had to be returned to after the fact. However, this may have contributed to the lack of registration errors, as the system was not trying to align cloud segments with unsure local positions.

| Site | BCN1 | | | IAAC1 | | |
|----------------|------------|-----------|------------|------------|-----------|------------|
| Image Count | 164 | | | 379 | | |
| Software | MS | MR | CM | MS | MR | CM |
| Aligned Images | 160 | 164 | 164 | 379 | 379 | 379 |
| Sparse Time | 0:05:40 | 1:12:00 | 5:40:00 | 0:17:00 | 1:53:00 | 14:30:00 |
| Dense Time | 0:34:00 | 2:29:00 | 5:40:00 | 1:01:00 | 0:09:12 | 14:30:00 |
| Sparse Count | 121,860 | 222,964 | 210,770 | 258,676 | 425,039 | 352,934 |
| Dense Count | 17,026,542 | 1,419,703 | 12,030,774 | 44,601,867 | 2,105,644 | 22,034,220 |

Table 6.1: Run time and point counts for tested photogrammetry softwares



Figure 6.1: An example of reconstruction error produced by Metashape in site BCN1. The truss in red does not in fact exist.

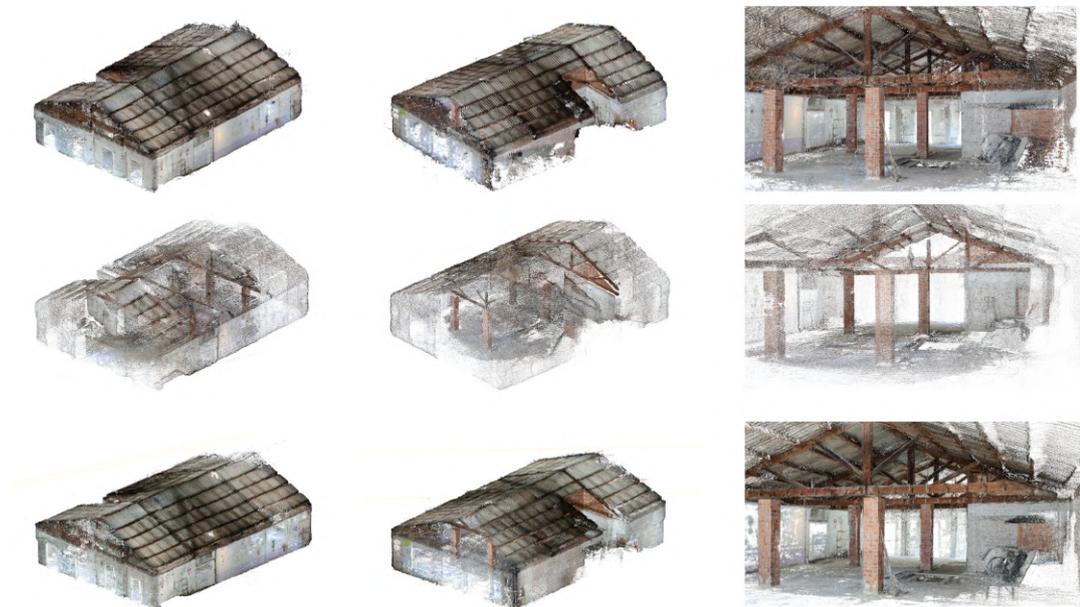


Figure 6.2: Whole site and interior reconstructions using tested software. Row 1 : Metashape, Row 2 : Meshroom, Row 3 : Colmap

Photogrammetry was found to be the most accurate method for reconstruction. Although mobile Lidar methods produced very fast results, they introduced much more noise and false contouring.



Figure 6.3: Comparison of colored results from various capture methods.
Row 1 : Photogrammetry, Row 2 : Stationary Lidar, Row 3 : Mobile Lidar

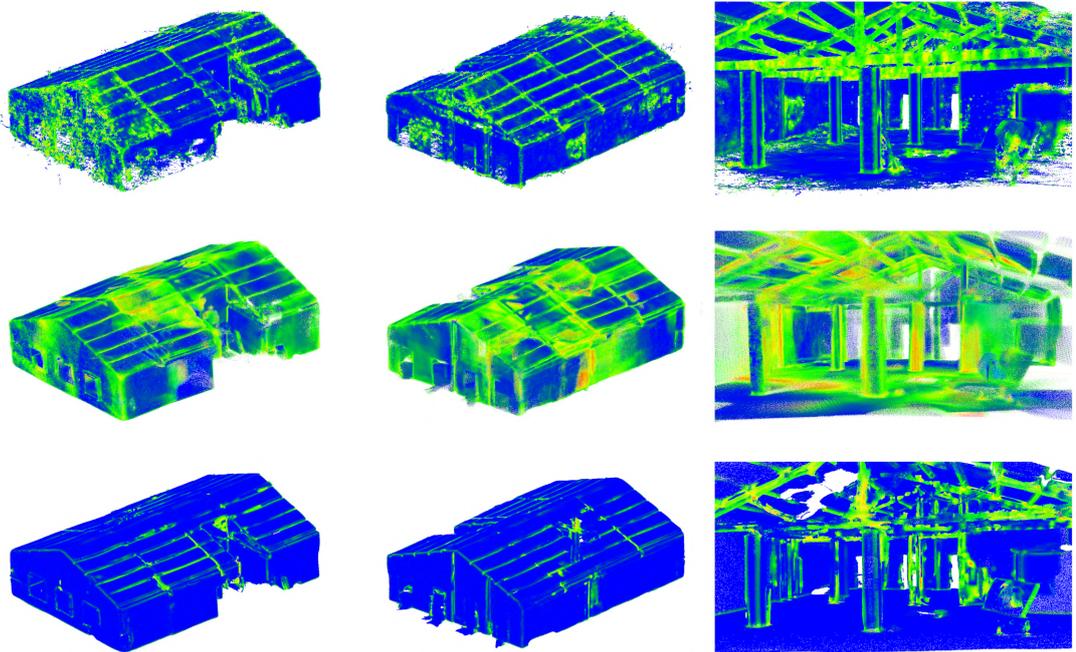


Figure 6.4: Comparison of local curvature resulting from various capture methods. Row 1 : Photogrammetry, Row 2 : Stationary Lidar, Row 3 : Mobile Lidar

6.2 Localization Performance and Generalizability

During training, the input vectors for each of the SVM-based classifiers were examined using t-distributed stochastic neighbor embedding (t-SNE), a dimensionality reduction technique for visualizing high-dimensional data as two-dimensional graphics. van der Maaten & Hinton (2008) These results are shown in figure 6.5. While this method shouldn't be viewed literally, it does present several broad observations about the methods used. The BOVW method generally shows an outer 'ring' of outlier samples, which may indicate that too many clusters are being used. The local binary points method is weakest, showing only strong groupings in the Metal class. The HUE/SAT method shows decent groupings though without much separation, with an expected overlap between the Brick and Wood classes. Finally the stacked results of the previous classifiers show relatively good clustering and separation.

Efficiency of Classifiers Used

Although localization time varied between hardware setups and input resolutions, the relative time used by each of the subclassifiers was measured. While important for the overall accuracy, the CNN classifier was found to take the most processing time by far, and would be the obvious first step for future optimization.

| | BoVW | LBP | HS | CNN | Stack |
|--------|-------------|------------|-----------|------------|--------------|
| Time % | 15 | 9 | 3 | 71 | 2 |

When testing patch combination, the harmonic mean was significantly the least accurate, with the geometric mean negligibly but still consistently more accurate than the arithmetic mean. Applying a falloff was not shown to increase accuracy significantly, however using local differences gave the best results.

| | Arithmetic Mean | Harmonic Mean | Geometric Mean |
|--------------------------------------|------------------------|----------------------|-----------------------|
| Raw Mean | 0.867 | 0.780 | 0.870 |
| Falloff Applied | 0.869 | 0.781 | 0.869 |
| Local Difference Applied | 0.951 | 0.856 | 0.959 |
| Falloff and Local Difference Applied | 0.910 | 0.849 | 0.914 |

Table 6.2: Accuracy Results for Patch Combination Methods

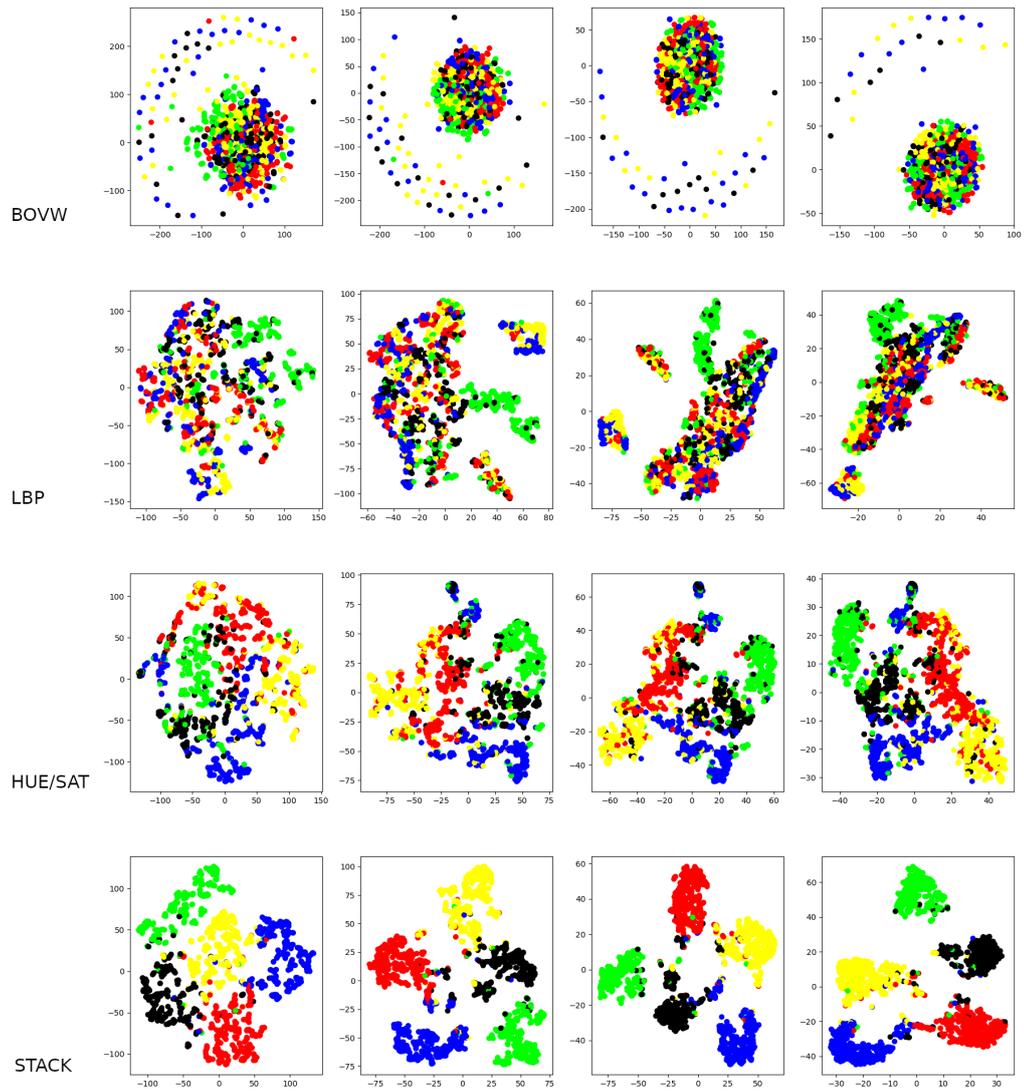


Figure 6.5: t-SNE visualization for the training dataset in the input form for each of the SVM-based classifiers

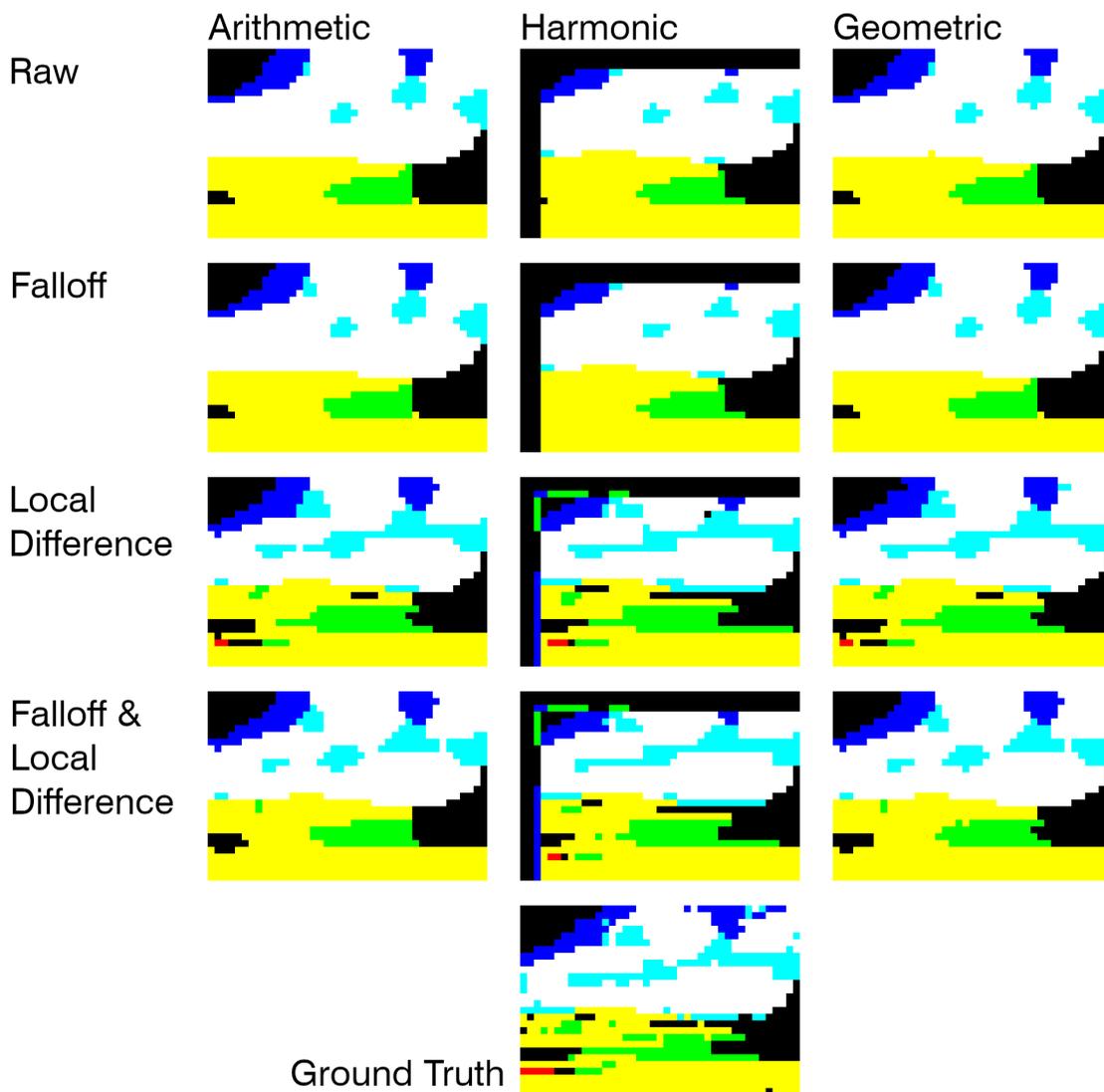


Figure 6.6: Results of patch combination on a test image of a landscape.

6.3 Fabricated Demonstrator

The strategy for fabrication was shown to generate relatively little waste at the time of new fabrication. After optimization, 85% of the layer components were well matched with structurally viable elements from the database, with 15% needing to be obtained from 'new' material. Of the matched elements, only 10% needed to be cut to size.

As the main layers components required little additional alteration, most of the waste would be generated while producing the connectors. Calculating the volumes, it was found that 8% of the volume of the input elements was lost to kerf and chips. However, for each piece, 27% was not used and effectively 'downcycled' into smaller dimensional elements. This was primarily due to the CNC methodology rather than the design, and could be optimized further.

After the system was set up, connector fabrication took place over 10 batches, each taking roughly 20 minutes of pure machine time to run. This was mostly due to relatively slow speeds being programmed for safety, for large scale production this could be optimized heavily, as the robot needs to make minimal positioning alterations during the run (see Figure 4.17).

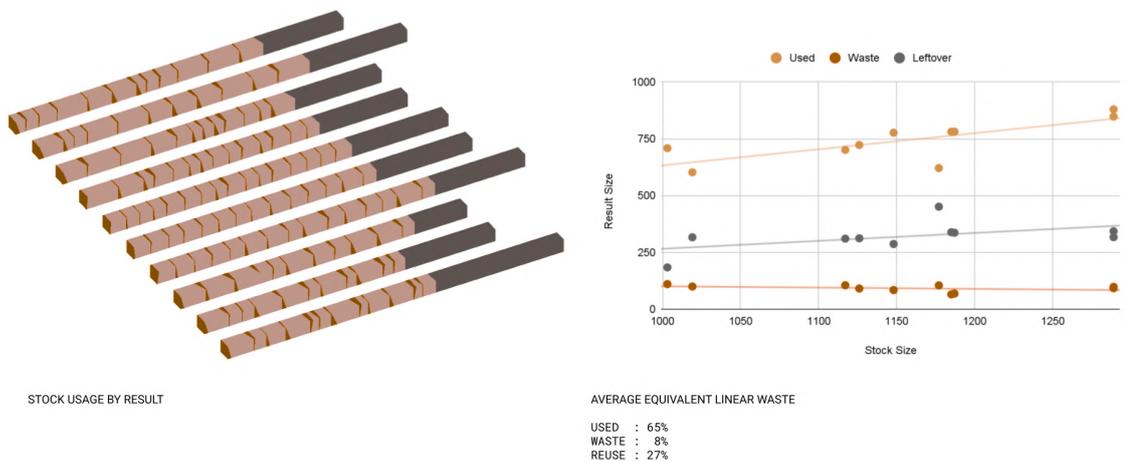


Figure 6.7: Calculate waste from fabrication process for connectors



Figure 6.8: Shots of the demonstrator under construction

The final construction showed high accuracy when compared to the digital model. From these results, 85% of the assembled pieces were within 10mm tolerance of their intended position. The worse areas were at the very top and bottom likely due to the higher torque forces present at those locations.

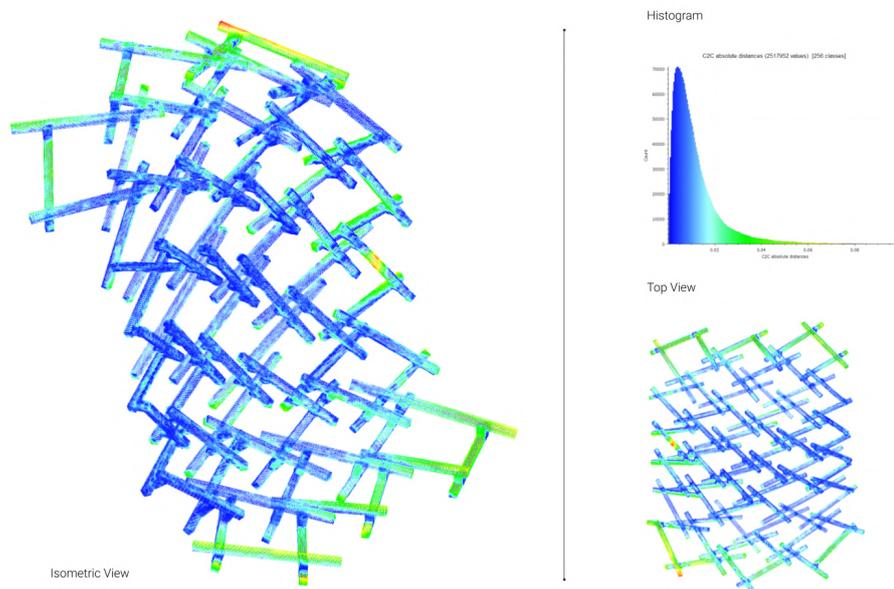


Figure 6.9: Results of comparison between photogrammetry-scanned construction and digital model

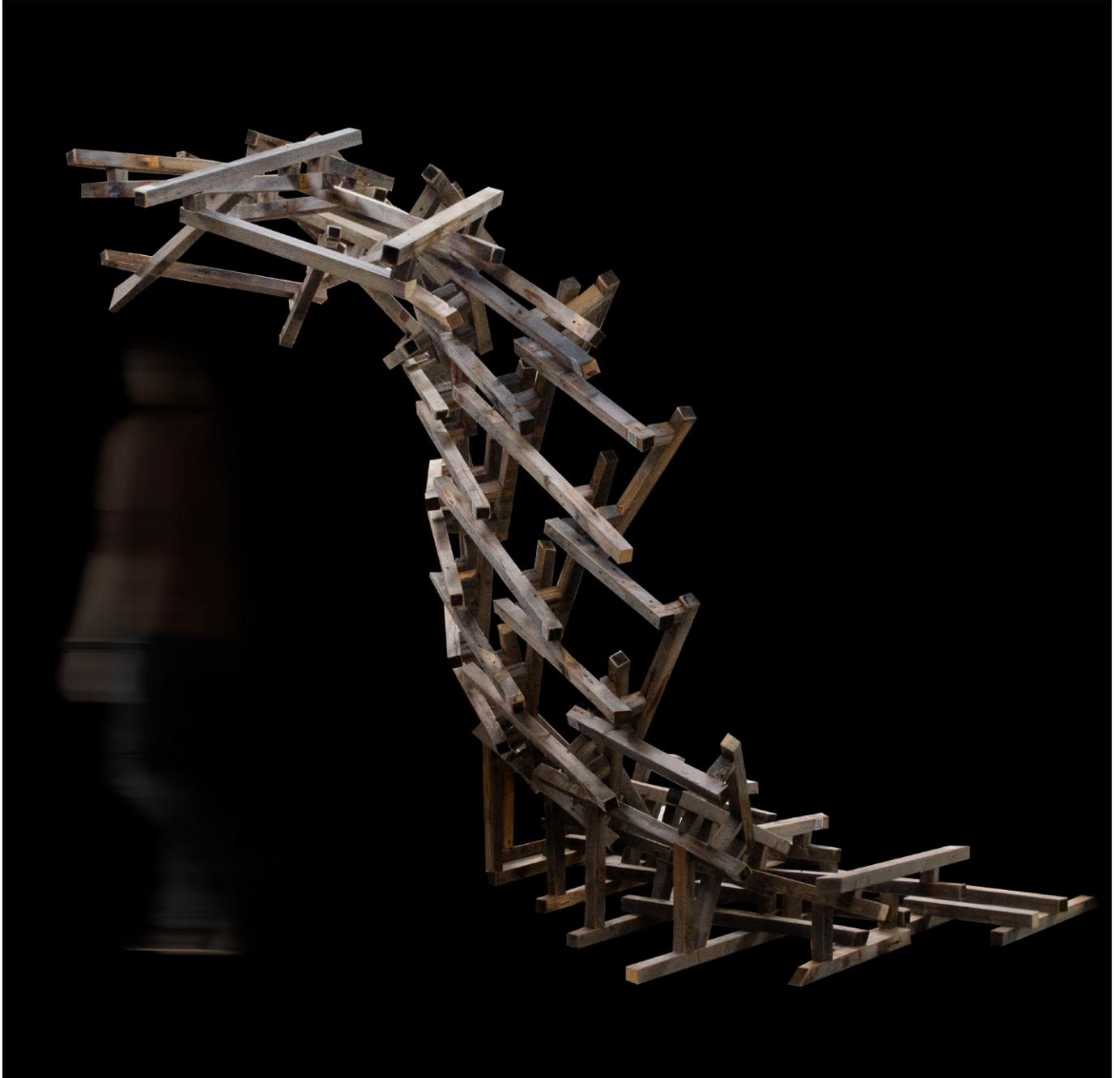


Figure 6.10: The completed demonstrator

Chapter 7

Conclusions

This project has developed a total-lifespan set of methods and technologies for applying digitization and machine-learning to mass re-use of construction and demolition waste. Each stage contains both a working base for new value or greater efficiency, as well as a multitude of opportunities for further development.

7.1 Site Digitization

With the reconstruction results obtained, we demonstrated the viability of low-cost capture methods for digitizing a new niche in the building lifespan. While industrial LIDAR systems remain the necessary state of art for ensuring new-construction, the level of detail to understand existing building stocks can be obtained by everyday operators using mobile devices. Looking forward, while photogrammetry proved to be the best intersection of accuracy and affordability, this decade will likely see a reversal of this. With a version of the accurate-but-expensive Lidar technology now being included in major mobile device lines, these will likely take the value lead in the next years of development.

7.2 Material Identification and Reconstruction

Across our dataset, the classifier systems were able to achieve 94% accuracy, showing the choice of classifiers and descriptors were well suited to the particulars of on site materials. With this setup, new useful classes to recover can be added with minimal reconfiguration. Fully proving the generalizability of these models across arbitrary demolition sites will be the ongoing test for its ultimate practical use. The methodology for synthesis of classification and reconstruction is novel in the site digitization space, bypassing the issues of direct point cloud analysis and classification.

Having taken this computer vision methodology to a full workflow, there were several conditions that ultimately became poor design choices, or that indicate promising areas of further research.

Choice of Classes

In a machine learning problem, choices of how to represent data, and what questions to ask, are often more fundamentally important than the implementation details. Here, there were several instances of confusion arising from the chosen set of classes.

Firstly, some methods of concrete casting using plywood leave the grain pattern on the concrete surface, which may texturally confuse the classifier.

Additionally, common misclassifications indicate the system should adopt a concept of subclasses. For instance, the features learned for the Brick class may appear strong in real instances of brick, CMU, stone masonry, and certain tilework. As the reuse methods and markets for these materials will be largely different, they should ultimately fall into separate classes. However, it will be useful to maintain the larger classifier to separate them first, then use a more specific classifier to assign them individually, dealing with crossover from concrete features etc.

The Problem of Visibility

Even for well-defined classes, the methodology of classification suffers from being based entirely on the visual details of the elements. In many cases, paint or other finishing will obscure or entirely cover the material. While estimations can be made from the element type, dimensions, and building context, certainty would be obtained through a mixture of manual spot-checking, and inclusion of additional data.

In most cases, the reconstruction and identification won't be able to operate with total confidence, so there will always be a need for a limited amount of spot checks by domain experts. This would operate by a system of clustering, where e.g. a collection of similarly painted columns would be internally associated, and only one would be marked for checking.

One method for aiding in this would be the inclusion of Ground Penetrating Radar (GPR technology). Using the reflected signals of radar pulses, the technology can detect both changes in dielectric density (thus materiality), as well as cracks and other deformations.

Integration of Depth Data

The majority of computer vision methods treat an image as inherently 1 or 3 dimensional at its lowest level (grayscale or rgb), although some work (He 2017) has shown improvements in classification accuracy even when using estimated depth maps. As our system already associates each input image with a known position, depth maps can be generated from the point cloud using several methods. This would allow the level of detail and texture visible at different distances to be more directly associated. It would also allow a natural cutoff to be applied, where the system won't attempt to classify parts of an image depicting areas too far away, thus without sufficient detail to be analyzed.

Post Processing the Material Map

When reviewing the classification maps or applying them to reconstructed geometry, the geometric reality of their patch-based form leads to visual confusion and 'spillover' points cast onto the wrong geometry. One method of smoothing, upscaling, (and perhaps better approximating the contour borders of) two-dimensional grid data is the Marching Squares Algorithm. Given the possible values at the corners of each cell, the algorithm chooses from a predetermined set of border lines. However the base version works on a single field value with a threshold applied, thus would need adaptation for a 6-class system. (Lorensen & Cline 1987)

7.3 Fabricated Demonstrator

Adding to the contemporary corpus of recovered-material based constructions, below the surface the demonstrator shows the value of the specific design and fabrication strategies. Its curved form is the first aspect noticed, the assembly of which from straight components implies the importance of the relatively small connector pieces. A close inspection also reveals the low amount of sawn edges, leaving the majority of the construction's texture to be determined by the material's own weathering, an effect made possible by the underlying design optimization system.

Ultimately, the demonstrator was a very specific case, entirely focused on producing the form and aesthetic. Moving forward, application and testing of these technologies in progressively more practical constructions will uncover further structural, spatial, and regulator constraints that the optimization systems will address.

Appendices

Appendix A

Grasshopper for Large Scale Data Processing

Grasshopper is often the first introduction to concepts of algorithmic thinking for those in architecture and design fields. As a programming language analogue, Vanilla (without additional plugins) Grasshopper occupies a particular and at times odd set of paradigms. Within a single execution, the method of calculation based on composing the inputs and outputs of many atomic functions, without true variables or other flexible persistent data storage, brings to mind concepts of functional programming. Likewise, operations are applied without loops, or even recursion, but instead through invisible map and reduce operations based on the environment's universal tree-based data structure.

The end result of these quirks are consistent areas of friction when operating at certain scales or levels of automation, a friction felt throughout the development of various reconstruction components of this project. As the advantages of geometric processing environment make it untenable to offload to a more traditional programming environment alternative methods must be utilized.

Data Trees and Memory

One major consequence of the dataflow style environment is the lack of scope. All results from previous calculations are available as inputs further on, and thus not discarded as the local variables of a function might be.

Persistent Data via `scriptcontext`

The Python component contains one method for persistent data, in the form of a single universal (shared by all Python components) dictionary that is empty upon opening and maintains its contents across execution. However, one drawback to this approach stems from the translation of data in and out of the Python environment, which can be slower than inputs to normal components, as well as certain datatypes not exportable from the environment. Additionally, it can significantly reduce the readability and comprehension

of the resulting script. Variables have to be initialized, checked, and reset separately, and if accessed from multiple components, care need to be taken that there is a direct downstream connection (as 'cousin' components do not have an explicit execution order).

Persistent Data via Baking and Pipelines

Depending on the datatypes involved, geometry data can be automatically baked into the associated Rhino file, and automatically imported via pipeline components. However, this can require careful attention by the user between different runs, to keep track of which elements have been baked, as it may be very difficult to edit out unrequired geometry afterwards.

Persistent Data via File Saving

For other data types, data can be written to text files for external processing. This was used here to access Open3D while splitting and joining element-clouds. This requires there to be an efficient textual representation of the data, and is fragile against inconsistencies in the methods of formatting ascii data (e.g. an malformed csv header will crash Grasshopper point cloud loading plugins).

Live Data

These issues are exacerbated when plugins enabling Live data processing are included, and specific execution signals are exchanged (e.g. as part of this project's system for recovered element scanning). As Grasshopper by default does not indicate which components have been updated, errant execution can easily quickly store overly large amounts of data in Recorder components or methods discussed above.

Chapter 8

References

- Bello, S. A., Yu, S. & Wang, C. (2020), ‘Review: deep learning on 3d point clouds’.
- Bertin, I., Mesnil, R., Jaeger, J.-M., Feraille, A. & Le Roy, R. (2020), ‘A bim-based framework and databank for reusing load-bearing structural elements’, *Sustainability* **12**(8).
URL: <https://www.mdpi.com/2071-1050/12/8/3147>
- Brodu, N. & Lague, D. (2012), ‘3d terrestrial lidar data classification of complex natural scenes using a multi-scale dimensionality criterion: Applications in geomorphology’, *ISPRS Journal of Photogrammetry and Remote Sensing* **68**, 121–134.
URL: <https://www.sciencedirect.com/science/article/pii/S0924271612000330>
- Cignoni, P., Rocchini, C. & Scopigno, R. (1998), ‘Metro: Measuring error on simplified surfaces’, *Computer Graphics Forum* **17**(2), 167–174.
URL: <https://onlinelibrary.wiley.com/doi/abs/10.1111/1467-8659.00236>
- de la Cruz, R. & Kreft, J.-U. (2019), ‘Geometric mean extension for data sets with zeros’.
- Deloitte (2017), ‘Resource efficient use of mixed wastes improving management of construction and demolition waste’.
- Design, M., Clifford, B., Mcgee, W. & Stone, Q. (2019), *Cyclopean Cannibalism, 2017*, pp. 32–33.
- Dimitrov, A. & Golparvar-Fard, M. (2014), ‘Vision-based material recognition for automated monitoring of construction progress and generating building information modeling from unordered site image collections’, *Advanced Engineering Informatics* **28**(1), 37–49.
URL: <https://www.sciencedirect.com/science/article/pii/S1474034613000943>

- Ding, F., Zhuang, Z., Liu, Y., Jiang, D., Yan, X. & Wang, Z. (2020), ‘Detecting defects on solid wood panels based on an improved ssd algorithm’, *Sensors* **20**(18).
URL: <https://www.mdpi.com/1424-8220/20/18/5315>
- Economidou, M. (2011), ‘Europes buildings under the microscope : A country-by-country review of the energy performance of buildings’.
- Ellen MacArthur Foundation (2019), ‘Towards the circular economy’.
URL: www.ellenmacarthurfoundation.org/business/reports
- Ellen MacArthur Foundation (2021), ‘Circular economy diagram’.
URL: <https://ellenmacarthurfoundation.org/circular-economy-diagram>
- Ester, M., Kriegel, H.-P., Sander, J. & Xu, X. (1996), A density-based algorithm for discovering clusters in large spatial databases with noise, *in* ‘Proceedings of the Second International Conference on Knowledge Discovery and Data Mining’, KDD’96, AAAI Press, pp. 226–231.
- Han, K. & Golparvar-Fard, M. (2017), ‘Crowdsourcing bim-guided collection of construction material library from site photologs’, *Visualization in Engineering* **5**, 1–13.
- He, K., Gkioxari, G., Dollár, P. & Girshick, R. (2018), ‘Mask r-cnn’.
- He, Y. (2017), ‘Estimated depth map helps image classification’.
- Herczeg, M., McKinnon, D., Milios, L., Bakas, I., Klaassens, E., Svatikova, K. & Widerberg, O. (2014), ‘Resource efficiency in the building sector’.
- Lorensen, W. & Cline, H. (1987), ‘Marching cubes: A high resolution 3d surface construction algorithm’, *ACM SIGGRAPH Computer Graphics* **21**, 163–.
- Maalek, R., Lichti, D. & Ruwanpura, J. (2018), ‘Robust segmentation of planar and linear features of terrestrial laser scanner point clouds acquired from construction sites’, *Sensors* **18**, 819.
- Macher, H., Landes, T. & Grussenmeyer, P. (2015), Point clouds segmentation as base for as-built bim creation, Vol. II-5/W3.
- Reflow (2021), ‘ReflowOS, Architecture and Manual for Distributed Network Setup and Maintenance’.
URL: <https://reflowos.dyne.org>
- Schnabel, R., Wahl, R. & Klein, R. (2007), ‘Efficient ransac for point-cloud shape detection’, *Computer Graphics Forum* **26**(2), 214–226.

- Siripurapu, M., Jun, J. W., Llubia, J. A., Haddad, Y., Najafi, M. M., Shakir, F. S. & Shalaby, N. (2016), ‘Digital urban orchard: robotic manufacturing of complex shape’.
- UN Directorate-General for Environment (2008), ‘Eu waste framework directive’.
- van der Maaten, L. & Hinton, G. (2008), ‘Visualizing data using t-sne’, *Journal of Machine Learning Research* **9**(86), 2579–2605.
URL: <http://jmlr.org/papers/v9/vandermaaten08a.html>
- WEF (2016), ‘Shaping the future of construction’.
URL: <http://www.fas.org/sgp/crs/intel/RL33539.pdf>
- Wijmans, E. & Furukawa, Y. (2016), ‘Exploiting 2d floorplan for building-scale panorama RGBD alignment’, *CoRR* **abs/1612.02859**.
URL: <http://arxiv.org/abs/1612.02859>
- Xiao, K., Engstrom, L., Ilyas, A. & Madry, A. (2020), ‘Noise or signal: The role of image backgrounds in object recognition’.
- Yuan, L., Guo, J. & Wang, Q. (2020), ‘Automatic classification of common building materials from 3d terrestrial laser scan data’, *Automation in Construction* **110**, 103017.
URL: <https://www.sciencedirect.com/science/article/pii/S0926580519308210>

|