

Human-in-the-loop Fabrication of 3D Surfaces with Natural Tree Branches

Maria Larsson

The University of Tokyo
ma.ka.larsson@gmail.com

Hironori Yoshida

The University of Tokyo
Preferred Networks Inc.
hyoshida@hy-ma.com

Takeo Igarashi

The University of Tokyo
takeo@acm.com

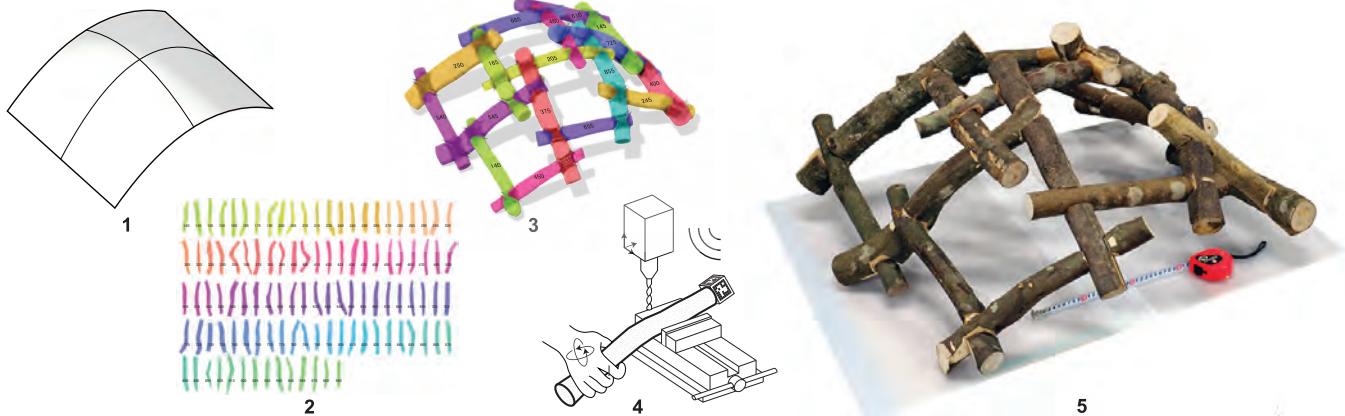


Figure 1: System overview. 1: Input surface. 2: Set of scanned real branches. 3: Best found fitting branch assigned to a reciprocal pattern of target curves. 4: Joints fabricated by audio-visual guided positioning by human and a computer numerical control (CNC) router. 5: Fabricated and assembled result.

ABSTRACT

We propose a design and fabrication system leveraging the diverse shapes of natural wood. With our system, precise geometries are fabricated from non-standardized, naturally curved branches. In this way, low-valued tree branches are up-cycled to a construction material. The process is implemented as follows. Taking a set of branches and a user-defined parametric target surface, the system makes a reciprocal pattern of curves that lie on the surface. Then, it automatically matches the shapes of the scanned branches with these curves. There was a similar attempt to build a large beam structure from tree trunks using industrial equipment such as cranes and robot arms. In contrast, we target smaller surface structures from tree branches using more accessible tools (2.5D CNC) with human-in-the-loop operation. We ask the user to manually place a branch to a defined orientation with the help of audio-visual guidance. Then a CNC-machine can mill out the joint. Finally, the structure is assembled by hand without screws or adhesives.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

SCF '19, June 16–18, 2019, Pittsburgh, PA, USA

© 2019 Copyright held by the owner/author(s). Publication rights licensed to ACM.
ACM ISBN 978-1-4503-6795-0/19/06...\$15.00
<https://doi.org/10.1145/3328939.3329000>

CCS CONCEPTS

• Applied computing → Architecture (buildings); Computer-aided design; Computer-aided manufacturing;

KEYWORDS

guided fabrication, audio-visual guidance, adaptive fabrication, 3D scanning, architecture, wood, joints

ACM Reference Format:

Maria Larsson, Hironori Yoshida, and Takeo Igarashi. 2019. Human-in-the-loop Fabrication of 3D Surfaces with Natural Tree Branches. In *SCF '19: Symposium on Computational Fabrication (SCF '19), June 16–18, 2019, Pittsburgh, PA, USA*. ACM, New York, NY, USA, 12 pages. <https://doi.org/10.1145/3328939.3329000>

1 INTRODUCTION

The unique shapes and textures of wood are aesthetically favored for furniture and architectural design. At the same time, the preference for industrial wood production is even-aged and monospecies conifer forestry. Such forests produce straight tree trunks which are suitable to cut into standardized pieces. The broadleaved forest, on the other hand, produces small tree trunks and branches that are typically just chipped and burned. Their unique curvy shapes make them difficult to systematically use for higher value applications in construction and manufacturing.

Our goal is to develop a computer-aided system that can handle these curvy tree branches. We propose a workflow leveraging digital fabrication techniques. To make a structurally sound structure

out of relatively thin and fragile branches, we choose to organize them in a reciprocal pattern. Such pattern effectively distributes forces across many components. Furthermore, we can easily map this pattern to a user-specified target surface. The curves lying on the surface are then matched with a set of scanned branches. We analyze each individual branch shape to optimize its placement in the design. For this purpose, we have developed a matching algorithm that minimizes the total distance between the branch skeletons and target curves. A method for matching naturally shaped wood to target curves was already developed for the experimental pavilion project *Wood Chip Barn* [Mollica and Self 2016]. Our algorithm is, however, more effective and efficient. The short computation time of our matching algorithm makes it possible to search among a relatively large library of branches. This increases the chance of good matches, and therefore the quality of the outcome. We demonstrate the effectiveness of the matching algorithm by analyzing outputs from a variety of input surfaces.

To materialize the design, the branches are connected with detachable cross lap-joints. These joints are all uniquely adapted to the various radii and aligned with the 3D skeletons. Fabricating such 3D joints is challenging since joints appear on various sides of the branch and at various angles. Rather than turning to robotic fabrication, we use a human-in-the-loop system with accessible hardware requirements. A user positions a branch with the help of audio-visual feedback. Voice commands, a beeping sound, and visual feedback guide the user to orient the branch correctly before fixing it in a vise. Then, a 2.5D CNC machine mills out the joint. We show the feasibility of the fabrication system by fabricating a dome-shaped surface (Fig. 1). Further, we evaluate the structural performance by several crush tests, exhibiting the structural capacity of the reciprocal pattern.

Our workflow enables geometrically precise fabrication of natural materials. With this system, low-value tree branches from the broadleaved forest are transformed into structural elements. Due to manual labor involved in the system, economical efficiency can be questioned compared to using engineered wood. However, we believe that there are aesthetic and social values to build one-of-a-kind structures from local materials. The system is most suitable for DIY applications, creating furniture, small architectural structures, sculptures, educational toys, and more.

In summary, our key contributions are:

- A process to match branches with target curves, and negotiate between existing materials and ideal shapes.
- An audio-visual guidance system for fabrication that helps a user to fine-tune the pose of a piece of material.

2 RELATED WORK

2.1 Design with Natural Wood

There is previous work that specifically focuses on digital design and fabrication with natural wood. These works are generally from the field of architecture and design, and tends to present full design systems from material collection to built object. Schindler scans tree branches and creates experimental furniture designs [Schindler et al. 2014]. In the project *BranchConnect*, Yoshida uses tree branches to construct a 2D screen wall [Yoshida et al. 2019]. *Bolefloor* fits naturally curvy wood planks next to each other to create a floor

[Ojamaa et al. 2013]. The experimental architectural design pavilion *Wood Chip Barn* is a 1:1 scale truss structure built from tree trunks [Mollica and Self 2016].

The design challenge how to arrange unique components is shared by all these works. They take different approaches. Schindler let users "manually" design furniture with commercial 3D modelling software [Schindler et al. 2014]. *BranchConnect* takes a crowdsourcing approach, creating an online game for multiple users to arrange branches together. *Bolefloor* and *Wood Chip Barn* develops algorithms for automatic arrangement of components. The optimization goal of *Bolefloor* is to pack the curvy planks next to each other to minimize the gaps between planks. For *Wood Chip Barn*, the optimization goal is to minimize the total distance between target curves of a truss, and the skeletons of forked tree trunks [Mollica and Self 2016].

Our work has the same optimization goal as *Wood Chip Barn* - to fit natural material to an ideal shape. They use an evolutionary optimization approach. They reportedly optimize over 56 variables to both globally swap places of components, and to update the local placements [Mollica and Self 2016]. They do not report on the computation time, but judging from the setup the computation time is long and does not go through all the possibilities. We share the same problem as *Wood Chip Barn*, but solve it in a different way. Rather than using a probabilistic technique, we use combinatorial optimization. With our matching algorithm, we can go through all possibilities and generate results within a few minutes for a structure with a similar number of components.

On a conceptual level, the difference between our work and *Wood Chip Barn* is that their target is a large structure requiring expert skill and professional tools such as crane and industrial robot arm. Our work, on the other hand, leans towards personal fabrication. The tree branches can be handled by humans during both fabrication and assembly. And our hardware setup is more basic. Furthermore, the difference in choice of material also leads to different structural strategies. A truss structure is appropriate for forked tree trunks. But it is not ideal for branches, mainly because it requires end-to-end joint connections. It is difficult to make such joints strong enough to carry load between the relatively thin branches. Our reciprocal pattern with cross joints is more suitable for branches.

2.2 Human-in-the-Loop Fabrication

As for fabrication with natural wood, previous work generally uses 2.5D CNC-machine milling for 2D structures [Ojamaa et al. 2013; Yoshida et al. 2019], or 6-axis robot for 3D structures [Mollica and Self 2016]. Our Human-in-the-loop fabrication system makes it possible to fabricate 3D structures with a 2.5D CNC-machine. We cover a larger space of fabrication possibilities, with less hardware.

There is further a category of related work augmenting CNC routers with human mobility. Rivers developed a mobile router with an actuated milling bit where a user roughly places the router on the material while the router fine-tunes the milling bit position in 2D [Rivers et al. 2012b]. Zoran presents a 3D-sculpting tool where the user move the router freely and the milling bit automatically stops when the user carves too much [Zoran and Paradiso 2013]. These two projects create powerful application by combining the

reachability of the human with the precision of the CNC-machine. We are inspired by these works, and try to combine the flexible way in which a human hand can orient a piece of material with the power of the machine. But rather than using the machine to "correct" the sloppiness of the human, we let the computer guide the human to precision.

There are precedent work also on the topic of computer guided fabrication. It includes another project by Rivers who presents a fabrication method for 3D-sculpting aided by computer feedback, where a camera-projection system analyzes the current shape compared to a target, and projects colors that indicate where material should be added and subtracted [Rivers et al. 2012a]. Yoshida uses the same principle as Rivers, but applies it to a building scale [Yoshida et al. 2015]. They guide construction workers to build a free form shape by pouring chopsticks and glue. However, in our case, the human does not have to work constantly, but only spot-wise when re-positioning is necessary. Furthermore, we try a different mode of guidance compared to precedent work - namely audio-visual. It is, to our best knowledge a novel approach in the field of fabrication; although, it should be noted that audio guidance has been extensively used for other applications such as aircraft operation [DR et al. 1996], surgical tools [Rissanen et al. 2006], and aids for people with impaired vision [Katz et al. 2012].

2.3 Reciprocal Frame Structures

The target pattern we use as a basis for matching branches, is a reciprocal frame pattern. This type of structure has a long history and has been widely studied. Recent work proposes computational design tools to create and construct reciprocal frame structures [Song et al. 2013; Tai 2012]. There is also recent work on the optimization of closely resembling cell packing structures [Pottmann et al. 2015]. Our target pattern is similar to these works, with some modifications to fit our context. Specifically, the components that make up the reciprocal frame are, in our case, curvy branches rather than straight engineered wood.

3 WORKFLOW

We present a workflow from raw material collection to planning and fabrication (Fig. 2). First, branches are collected from the environment and trimmed down to a manageable length (Sec. 3.1). A unique marker cube is attached to each branch. Then, it is scanned. Skeleton and other key features are extracted from the scanned data. Parallel to collecting and analyzing existing material, a free-form parameterized surface is created and set as target shape (Sec. 3.2). A pattern of target curves is fitted to this shape. Then, our matching algorithm tries to fit the real branches to the target curves (Sec. 3.3). It calculates the geometric difference of each skeleton-target pair and decides which branch goes where. The positions of the branches are further optimized by relaxing the assembly to minimize the distances where branches are supposed to intersect. After defining which branch goes where and optimizing their positions, joint geometries are automatically generated at the intersections (Sec. 3.4). The joints are fabricated by a human-in-the-loop process (Sec. 3.5). With the help of audio-visual guidance, a user positions a branch in a vise so that the intended cutting area faces upwards. Then, the joint can be milled out by a 3-axis CNC machine. Finally,

to predict the structural performance of a reciprocal structure of branches, we performed a component crush test (Sec. 3.6).

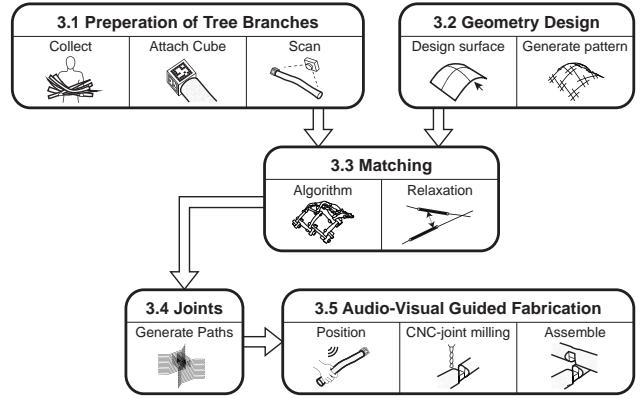


Figure 2: Work-flow diagram

3.1 Preparation of Tree Branches

3.1.1 Material Preparation. Branches are collected from the environment: either leftovers from broadleaved forest maintenance and lumber production or fallen branches from parks. We found that branches with a diameter ranging from 4 to 7 [cm] are thick enough to be strong, and thin enough to be light and easy to work with. The branches are cut into components with a manageable length of 40-60 [cm]. Finally, a cube with five unique markers is attached to the end of each branch (Fig. 3). The marker cube makes it possible for the system to identify each branch and track its pose during fabrication. We also tested markerless tracking, but it was not reliable.

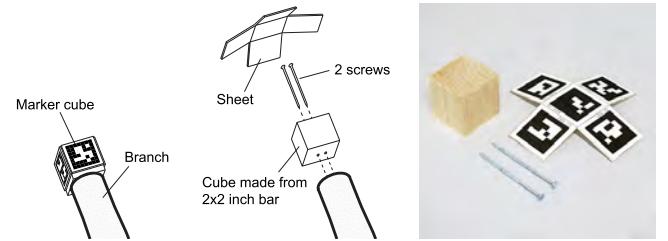


Figure 3: Marker cubes attached on the branches.

3.1.2 Scanning and modeling. A branch is placed on a rotating table with AR markers (Fig. 4). The user manually rotates the table while an RGBD camera takes a series of images (about 50-100). The branches are scanned one at a time, taking 30 seconds per branch. A point-cloud is reconstructed from the multiple captures by truncated signed distance function (TSDF) implemented in *Open3D* [Zhou et al. 2018]. Then the acquired point-cloud is then further processed by running a MeshLab script to segment and downsample the branch. The branch is segmented from the background by a z-height condition. Then the point-cloud is down-sampled

with the "Point Cloud Simplification" filter, and the mesh is constructed with the "Ball Pivot" function. Since the branch lies on a table when it is scanned, the side that faces the table is not visible to the camera. Therefore, the mesh is incomplete.

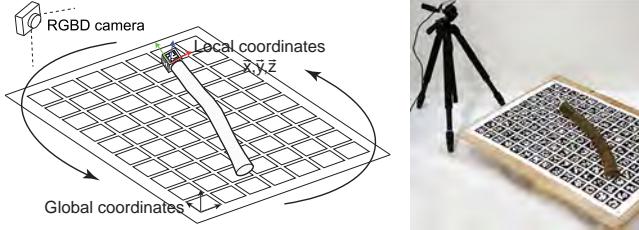


Figure 4: Scanning setup.

3.1.3 Skeleton extraction. For the further processes we need the branch skeleton and radius. The skeleton is used for matching real branches to target curves, and the radius is necessary to calculate the geometries of the customized joints. Skeleton and radii are automatically extracted from the mesh (Fig. 5). First, the 2D skeleton is created by filtering and sorting the midpoints of the Delaunay triangulation lines of the 2D contour in top view (Fig. 5-1). At every skeleton point (p_i), the local radius (r_i) is defined as half of the diameter of the 2D contour (Fig. 5-2). Until here the process is the same as precedent work on branch structures [Yoshida et al. 2019]. Then, the 3D skeleton is created by projecting the 2D skeleton to the top surface of the mesh, and shifting each point down by the local radius (Fig. 5-3). Finally, the whole shape of the branch is approximated as a generalized cylinder with circular cross sections (Fig. 5-4). This is a simplification. In reality, some branches have more elliptical cross-sections and local irregularities caused by nodes. But an advantage is that this process works although the mesh is not complete. The skeleton extraction and following steps of the system can handle smooth branches with 3D curvatures. It does not support spiraling or grafted branches (Fig. 6).

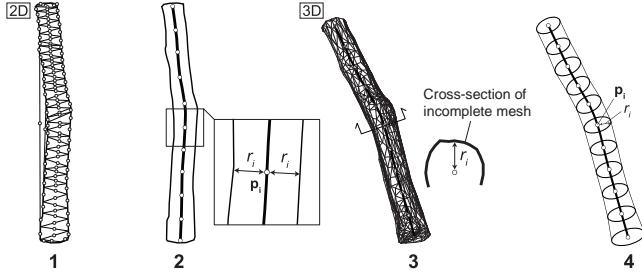


Figure 5: Skeleton extraction from incomplete mesh.

3.2 Geometry Design

The input is given as a smooth parametric surface. It is converted to a network of target curves to be assigned with branches. This pattern is created as illustrated in figure 7. The input surface is first divided in U and V direction (Fig. 7-2). The number of divisions are

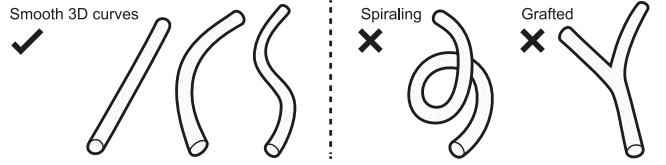


Figure 6: Supported and unsupported branch types.

automatically set so that all final target curves are shorter than the median length of the available set of branches. Each line of the grid is rotated 20 degrees (Fig. 7-4). Then, each line is projected to the surface in the direction of the surface normal at its midpoint (Fig. 7-5). Finally it is extended so that it intersects with the adjacent curves (Fig. 7-6). This pattern makes a structurally sound reciprocal structure. It also has the advantage that it makes "loops". When a loop is completed, it tends to fix the joints (Fig. 8). Therefore, joints can be fabricated to fit loosely for ease of assembly, and to absorb some degree of fabrication error.

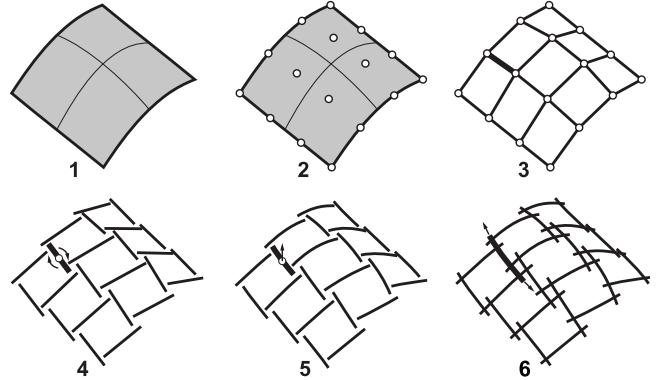


Figure 7: Target curve pattern generated from input surface.

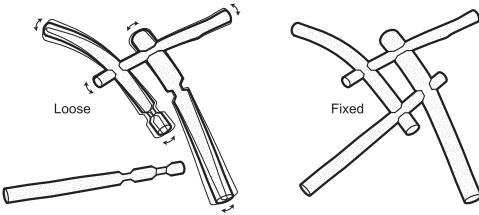


Figure 8: Assembly is fixed when the loop is completed.

The input surface of our system should be a single four-sided NURBS patch and cannot be self-intersecting or trimmed. The maximum curvature of the surface should be smaller than the maximum curvature of the available branches. It should also have a reasonable size to be fabricated with the branches of the given length, roughly 40-60 [cm].

The system does not dictate whether the target surface or branches comes first. In case the user choose to design the input surface first, it is possible to visualize the ideal branch shapes for constructing

that particular geometry. This can then be used as a rough guide when collecting branches. Vice versa, if the material comes first, it is possible to explore various design options while checking if the target curves are similar to the collected branches. The difference can be observed by monitoring the cost for various input surfaces (reported on in the next section, Sec. 3.3).

3.3 Matching

The goal is to match real branches to target curves. Our strategy is to search for the best pose of every branch to every target and calculate the cost as the difference between them. Then, we assign a unique branch to each target, so that the sum of the costs is minimized.

The computation of a cost between a target curve T_i and branch skeleton S_j proceeds as follows. We produce a number of candidates and pick the one with the lowest cost. We first consider the length of the branch skeleton. For each skeleton S_j we pick segments S_k that has the same length as T_i , as long as the branch is longer than the target. We first coarsely try at every 5 [cm], and then refine the segmentation at every 1 [cm] (Fig. 9). There is also a choice of direction of alignment, thus we try both curve directions of S_k .

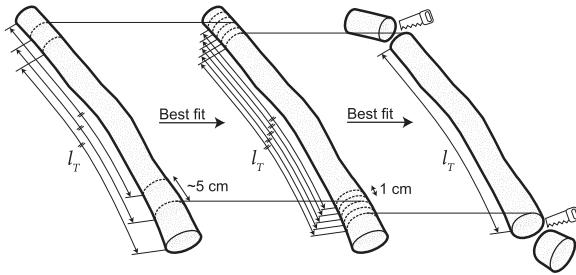


Figure 9: Branch segmentation coarsely tested at every 5 cm and refined at every 1 cm

To calculate the difference between the curves, the branch skeleton S_k is oriented to the target curve T_i so that the branch skeleton matches the target curve as much as possible. To do so we introduce two criteria:

- the directions of the curves should be aligned
- the strongest curvatures should lie in the same plane.¹

and extract two key features for each curve:

- joint intersection points P
- plane B of strongest curvature

For the target curve T_i the joint points are simply the intersections with other target curves in the network pattern (Fig. 10-left). Given the pattern, the length of P is either 2 or 4. To get the corresponding points on the skeleton, each segment S_k is evaluated at the normalized parameters of the joint points on T_i (Fig. 10-right). The plane B is formed by analyzing the strongest curvature. It is done by dividing the curve at every 2 [cm] and fitting a plane to those points.

¹It is not necessarily possible to fulfill the two criteria at once. Since the branches have three-dimensional curvatures, the curve direction might not lie in the plane of the strongest curvature. Priority is given to the first criteria.

Using the two criteria and curve features we calculate local coordinate vectors (\vec{v}, \vec{u}) with local origin (O) for each curve, and then orient S_k to T_i by a rigid transformation. The local vector space is calculated as follows. The first criteria - to align the directions - defines the first basis vector (\vec{v}) . When the length of P is 2, it is the vector between these two points. In the other case, when the length of P is 4, it is the vector from the average of the first two points to the average of the last two points. The second criteria - that the strongest curves should lie in the same plane - is used to calculate the second basis vector (\vec{u}) . The second basis vector \vec{u} is orthogonal to \vec{v} and to the normal vector \vec{n}_B of B . We try the orientation with two opposite directions of vector \vec{u} , because both directions fulfill the criteria to be perpendicular to \vec{n}_B and \vec{v} . Finally the coordinate origin (O) , is simply defined as the average of the curve joint points P . After defining the local vector spaces for each curve, we can perform the rigid transformation for each candidate and calculate the cost.

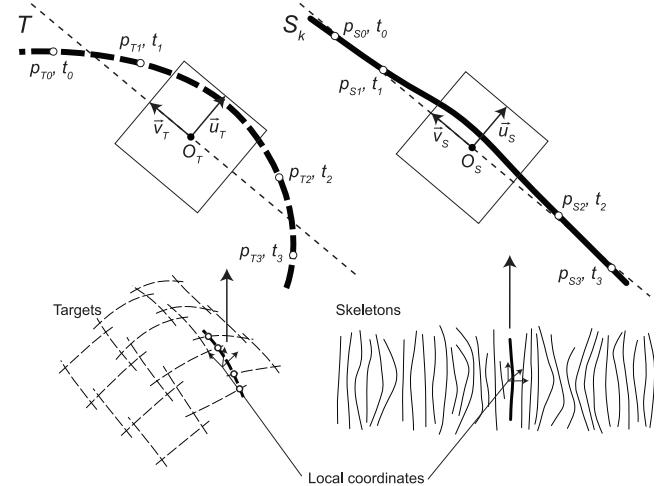


Figure 10: Defining the local vector spaces for target curve T and skeleton S .

3.3.1 Cost. The difference between the skeleton and the target curve is evaluated at a number of points on the target curve, referred to as the evaluation points Q . To save computation time, we chose the evaluation points carefully. We first pick the intersection points P , and then recursively add a new point between two adjacent points until distances between adjacent points becomes lower than 15 cm. Then the cost (c) is calculated as:

$$c_{i,j} = \sqrt{\frac{1}{n} \sum_{i=1}^n (Q_i - Q'_i)^2} \quad (1)$$

where Q'_i is the closest points to Q_i on the oriented skeleton curve S' . (Fig. 11). We compute the cost for every pair of target curves and branches, obtaining a full matrix of lowest found costs for possible alignments.

$$C = \begin{bmatrix} c_{0,0} & \cdots & c_{0,m} \\ \vdots & \ddots & \vdots \\ c_{n,0} & \cdots & c_{n,m} \end{bmatrix}$$

where n is the number of target curves and m is the number of real branches. The computation time for this matrix is roughly 5-10 seconds for a 20×100 matrix, i.e. when we have 20 target curves that we are trying to match with 100 collected branches. Although we test many candidates the computation is relatively quick because the algorithm is based on simple but effective geometrical calculations. The computation time does, however, increase exponentially when the number of branches or target curves increases.

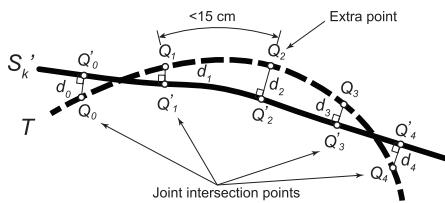


Figure 11: Evaluation points for cost calculation.

3.3.2 Assigning. The next step is to choose which branch goes where, while minimizing the sum of the cost values. We implement the Hungarian Algorithm to solve this problem [Kuhn and Yaw 1955]. The computation time for the Hungarian Algorithm is very short; it just a fraction of a second for the normal matrix size that we are using.

3.3.3 Branch Structure Relaxation. The previous step aligns each branch to each target curve independently, so some branches are far apart at some joints. That is a problem for fabrication. So, in this step, we apply force-guided relaxation adjusting the positions and orientations of branches so that joints are tightly connected. This is similar to precedent work on reciprocal structures which also applies relaxation [Song et al. 2013]. In order to increase the chances of making tight intersections, our model allows for the branches to slide along the intersecting curve within a range of 4 [cm] (Fig. 12). Springs are applied at the shortest distance between the two sliding lines. To avoid too much deviation, we also apply strings to the original target joint point. The optimization runs as long as the result is improving, usually for 50-100 steps, taking 5-10 seconds. This method does not guarantee convergence, but is has been sufficient for our purposes.

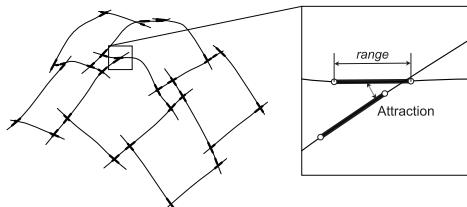


Figure 12: Force-guided relaxation for optimizing joint tightness.

3.4 Joints

The geometries of cross lap joints are calculated at each intersection. Each joint is customized to the skeleton intersection angle and the local radii. It is given by the reciprocal pattern that every other joint should go over/under (Fig. 13-left). We use two different joint types for the loop and edges of the reciprocal pattern (Fig. 13-right). By experience we found that where joints on opposite sides of one branch are far from each other, as on the edge of the reciprocal pattern, it is possible to connect the branches with standard cross-lap joints, referred to as joint type 1 (Fig. 14-top). This joint type is meant to be assembled by pushing it straight in. It is similar to the joint of precedent work on branch structures [Yoshida et al. 2019], with the only difference being that they calculate the joint in 2D while we orient it in 3D. Where joints on opposite sides of one branch are close to each other, as in the "loop", it is, however, impossible to assemble the branches with joint type 1. Therefore, we developed a joint shape that can be rotated into place, referred to as joint type 2 (Fig. 14-bottom).

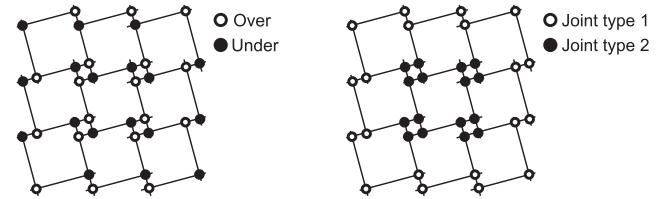


Figure 13: Joint side and type allocation on reciprocal pattern.

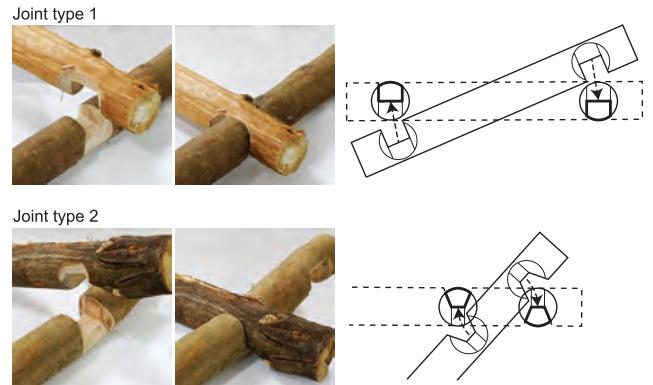


Figure 14: Joint types.

The calculation of the geometry and cutting paths of joint type 1 is illustrated in Fig. 15. Key vectors v_0 and v_1 are calculated as:

$$\vec{v}_0 = r_1 * \vec{t}_0 \quad (2)$$

$$\vec{v}_1 = r_0 * \vec{t}_1 \quad (3)$$

where r is the local radius and \vec{t} is the unit tangent vector of skeleton S at intersection point p (Fig. 15-1). These vectors are added and subtracted from p to define eight control points (A, B, C, D, a, b, c, d):

$$A = p + \vec{v}_0 - \vec{v}_1 \quad (4)$$

and so on.

$$a = p + \frac{1}{2}(\vec{v}_0 - \vec{v}_1) \quad (5)$$

and so on. The control points are connected to define the outlines of *side cuts* and a *center cut*. The *side cuts* are extended 5 [mm] outside the virtual model of the branch to account for fabrication tolerances and in case there is a local irregularity on the real branch that protrudes outside the assumed circular cross-section. The cutting paths are made from these outlines, with the milling bit radius (r_{mill}) and as a variable parameter (Fig. 15-2). The cutting paths are extruded in depth, double the local radius ($2r_0$) for the *side cuts* and one time the local radius (r_0) for the *center cut* (Fig. 15-3). The milling path of joint type 2 is calculated in a similar parametric manner. All customized milling paths are automatically generated and exported as GCode.

The cutting axis \vec{v}_{cut} is defined as follows:

$$\vec{v}_{cut} = \vec{v}_0 \times \vec{v}_1 \quad (6)$$

This vector should be aligned with the milling bit of the CNC router during fabrication.

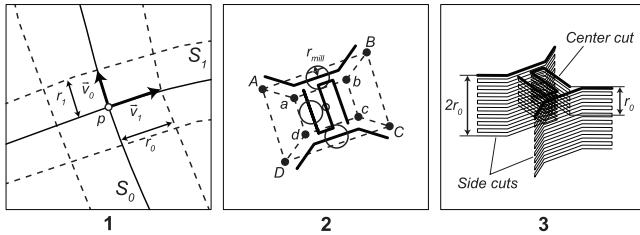


Figure 15: Illustration calculation of the custom joint geometry and milling path (type 1).

3.5 Audio-Visual Guided Fabrication

Each joint on one branch has a unique cutting direction defined by vector \vec{v}_{cut} (Fig. 16). As long as we use a 3-axis CNC router, it is then necessary to position the branch for every joint cut. We take a human-in-the-loop approach for this positioning task, asking the user to place the branch at a specific orientation. More precisely, the branch needs to be oriented so that \vec{v}_{cut} is aligned with the Z-axis of the CNC router. The "roll" and "pitch" angles need to be set by the user. The "yaw" angle, which is effectively the rotation around the Z-axis of the machine, and translation do not need to be set by the user - the milling path is automatically adjusted to those parameters after reading the pose of the fixed branch. To place the branch correctly, the user receives audio-visual guidance in the form of speech hints and a stepped sound with a variable frequency. Furthermore, the user is asked to do a fine calibration of the milling bit position on the physical branch during fabrication. The reason is that the automatized pose recognition system with AR markers has an accuracy in the range of centimeters, but millimeter accuracy is required to achieve desired fabrication quality.

As for the audio-visual guidance system, we choose this modality for a variety of reasons. The fabrication setup poses some limitations. The user holds the branch in one hand, and the lever of the vise in the other. So, the user needs to have two hands free. The user

also needs to keep visual attention at the fabrication scene. Considering these constraints, we ruled out options including holding a smartphone and wearing a VR headset. AR glasses, projection mapping, and audio feedback were identified as suitable options. We choose audio over the other modalities. AR glasses were expected to be lagging. A projection system was expected to be slow at launching. Since the system is used for just a few minutes at the time, with up to an hour break in between (while the CNC-machine is milling), it would be tedious to turn on a projector and wait for it to warm up every time. This can be solved by using a laser projector, which turns on and off instantly. However, we found the more basic setup using audio feedback sufficient for the task. We simply use the built-in speaker of the PC on which the pose-tracking system is running. With this mode of guidance, the user has both hands free and can look at the branch while placing it. It instantly turns on and off. And even when there is some latency, audio lag is less irritating than visual lag, because it is not obviously detectable. Later, we did however extend the system by adding visual guidance on the PC screen, so that the user can glance over to the screen if he or she feels lost. The result is an audio-visual guidance system.

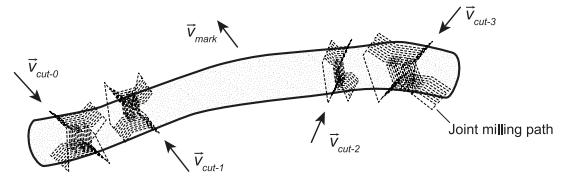


Figure 16: Example of a branch with unique cutting angles for each joint.

1

3.5.1 Fabrication setup. The fabrication setup is a CNC router with a vise to fix branches for milling (Fig. 17). A board of AR markers is installed on the work surface of the router. These markers correspond to the coordinates of the CNC machine. A camera captures the scene and reads the pose of the marker cube on the branch in relation to the machine.

3.5.2 Fabrication workflow. To fabricate the joints on one branch, all joint locations are first marked in one go. That ensures that the relationship between multiple joints on one branch is precise. Then, joints are milled out one by one, using the marks as references for joint center calibration. Fig. 18 illustrates the fabrication workflow in detail. The user first places the branch in the vise according to the audio-visual guidance (Fig. 18-1). At this point, the system relies on the AR markers to compute the position and orientation of the branch. After the user successfully fixes the branch, the system goes into calibration phase (Fig. 18-2). The milling bit first moves to the start point of the branch. The user corrects the start point by looking at the branch, and moving the milling bit with a joystick in X, Y, and Z directions so that the milling bit is positioned exactly at the center of the top end of the branch. The adjustment is usually within a 1 [cm] range. Then, the machine moves to the endpoint, and the user makes the same manual adjustment. According to these two positions, the system obtains a more accurate mapping between physical coordinates and virtual coordinates. The system

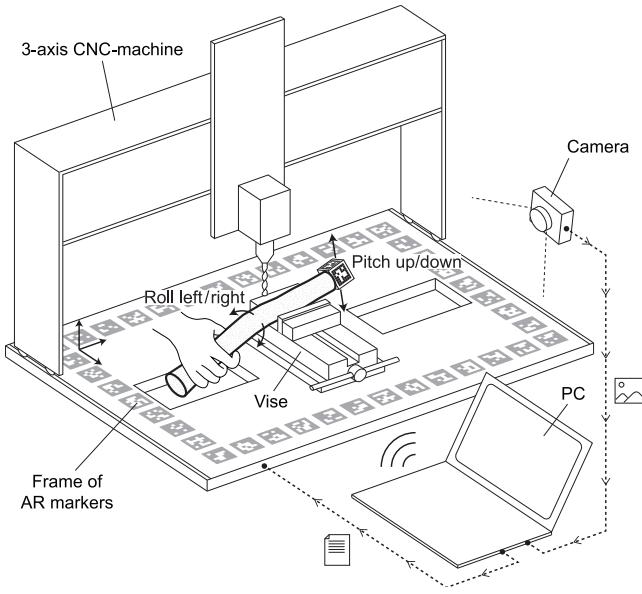


Figure 17: Fabrication setup. Receiving image input, a PC (personal computer) gives audio-visual guidance to a user. The PC also controls the CNC router.

then mills joint marks on the sides of the branch based on the accurate mapping (Fig. 18-3). The user now moves on to the actual fabrication of each joint. The user first re-positions the branch in the vise according to the audio-visual guidance (Fig. 18-4). Here, the system guides the user to rotate the cutting vector (v_{cut}) of the joint so that it points up, aligning with the z-axis of the milling machine. After the user successfully fixes the branch again, the system goes into calibration phase for the joint (Fig. 18-5). The milling bit automatically moves to the center of the joint, based on the AR-markers. Then, the user again calibrates the translation by manually adjusting the milling bit so that it is perfectly centered on the joint mark. The CNC-machine finally mills the joint based on the accurate mapping (Fig. 18-6). Step 4-6 is repeated for each joint.

3.5.3 Positioning with Audio-visual Guidance. In the positioning, the user adjusts roll and pitch angles. Roll is the rotation around the axis of the branch itself, and pitch is the angle set by tilting the wrist up and down. On the computer screen, the user sees the scene captured by the camera overlaid with visual guidance showing the current orientation of \vec{v}_{cut} (yellow) and target \vec{v}_{cut} (cyan), as well as roll and pitch angles separately (Fig 19-left). The user is first asked to set the roll angle, and is given speech feedback to "twist right" or "twist left" (Fig. 19-right). When the correct roll angle is close, the system switches from speech to beep mode. The distance to the target angle is proportionally described by the frequency of the sound, until the target roll angle is reached. Once the current angle satisfies within the threshold around target angle, the system provides sound as well as visual effect. Then, the user is asked to set the pitch angle in a similar manner, while given hints to "move up" and "move down". While adjusting the pitch angle, feedback for roll angle is not provided. After satisfying the pitch angle, the

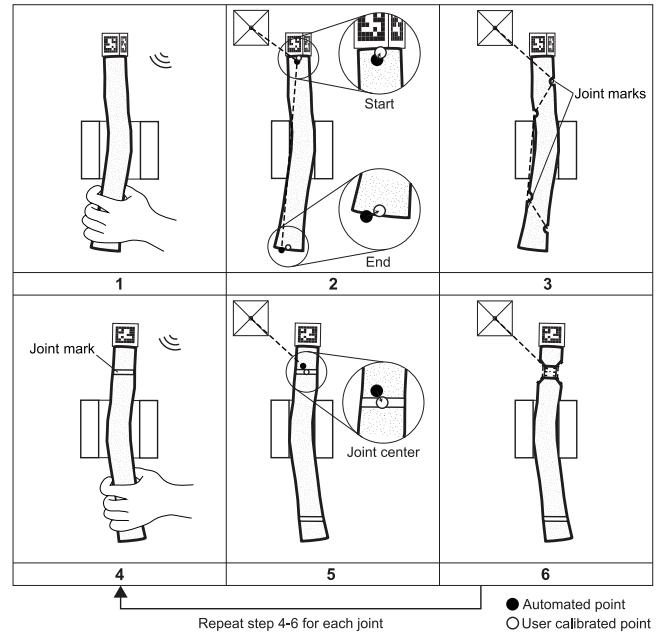


Figure 18: Detailed fabrication workflow including calibration steps (2 and 5).

system moves to fine-tuning mode. In this mode, roll and pitch angles are combined and the threshold is more strict. When both roll and pitch angles are reached, an appraisal sound is played, and the user fixes the branch in the vise.

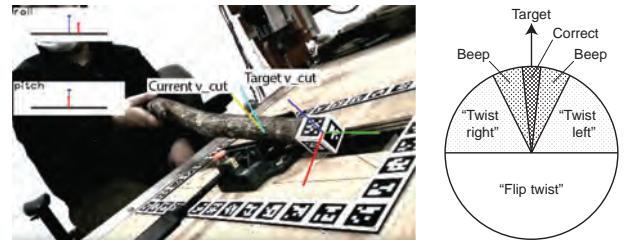


Figure 19: Left: Visual guidance interface for positioning. Right: diagram of audio hints for setting the roll angle.

3.6 Structural Testing

To evaluate the structural behaviour of the reciprocal pattern of branches, we conducted flexural tests on two basic loop components of 4 branches. The experimental setup is shown in Fig. 20-1 and 2. Fig. 21 shows how the two samples deform with increased input load until they break. In the beginning, the plot is oscillating as the structure settles down with increasing input load. The samples yielded at 2.3 and 2.4 [kN], respectively. Both samples were yielded in the same way - the shortest span between joints split along the direction of the fibers (Fig. 20-3). Another finding is that the settlement of the structure was relatively smooth, demonstrating that the joints were precisely fabricated.

We also modeled the sample and simulated in a finite element analysis (FEA) with the software *Karamba Physics*. The joints were modeled as continuous nodes whereas in the reality joints are disconnected resulting in more displacements than the simulation. Taking the parameters we obtained through this test, we modeled the fabricated dome and conducted the compression test (presented under results in Section 4.4).



Figure 20: Flexural test. 1,2: Experimental setup. 3: Failure mode.

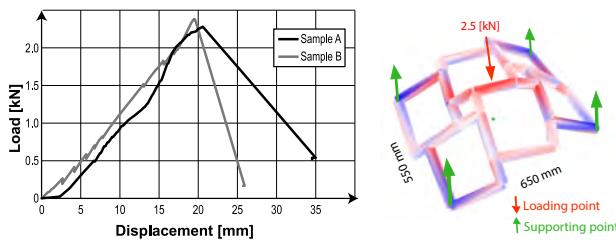


Figure 21: Flexural test plot (left), and the modeled dome for FEA based on the test (right).

4 RESULTS

4.1 Matching Evaluation

Fig. 22 and Table 1 show 7 examples of input surfaces matched with the same set of branches we used to for the fabricated result (reported in the next section). The table shows computational time², and joint distances before and after relaxation. This result shows that the matching algorithm finds branches that fit the target curves, and that the relaxation tightens the joint connections for examples where there are large distances. To further demonstrate the effectiveness of the matching algorithm, we present in Fig. 23 a comparison between our matching result and two other scenarios where matching error is maximized (as opposed to minimized), and using straight rods for matching (as opposed to curved branches). This comparison shows that our matching algorithm effectively finds branches that follow the input surface, resulting in tight intersections.

4.2 Fabrication Evaluation

We tested the system by fabricating a physical dome out of 17 branches with 24 joint connections (Fig. 1). We obtained a bulk of branches of various tree species from the broad-leaved forestry with a length of roughly 50 [cm] and diameters varying from 4

²Computation time is measured on a laptop computer with CPU 2.60 GHz and RAM 16.0 GB

to 7 [cm]. We attached marker cubes and scanned 105 branches to be used for matching. We designed an input surface measuring 70 × 60 [cm]. The system generated a reciprocal pattern giving 17 target curves. We ran the matching algorithm which selected corresponding branches. We fabricated the joints with our human-in-the-loop system using one person and one 2.5D CNC-router. After all the joints were milled out, the dome was assembled by hand, while looking at a plan of which branch goes where. It was the same one person who fabricated the joints that performed the assembly. Later, the joints were color coded with 24 unique colors for each joint, so that the dome could be easily reassembled for transportation and demonstration purposes. Fig. 24 shows a scan of the fabricated dome overlaid with the planned structure. Fig. 25 shows photographs of some fabricated branches next to their virtual models, skeletons, and target curves.

Fabricating the dome took one week. To fabricate one joint took on average 40 minutes, divided by task as follows: positioning the branch 5 minutes (12%), milling mark 5 minutes (12%), and milling joint 30 minutes (75%). Most of the fabrication time was spent for joint milling. A joint is milled out in layers in steps of 1 [mm], at an average of 30 layers, so the CNC-milling is time consuming. It is, however, automated so the user can do something else in the meantime while just keeping an eye on the machine for safety. Sometimes we had to manually lower the cutting speed when the wood was exceptionally hard. Some branches we used were oak, a very dense and hard type of wood, and sometimes the joint were located at a locally hard node.

4.3 Fabrication without Audio-Visual Guidance

Prior to developing the audio-visual guidance system, we fabricated another prototype (Fig. 26). Here the approach was to plane the sides of the branches, so that they could be positioned in a stable pose, letting the flat side rest on the table. This ensured a correct roll angle. To set the pitch angle, we used a water pass as a guidance to position the branch so that the joint area was leveled. This prototype took almost 3 weeks to fabricate. The reason for the long fabrication time was the milling time spent for planing the sides of the branches. This experience motivated us to develop the audio-visual guidance system, so that materials could be accurately positioned without planing the sides. Furthermore, we used branches of radii as small as 2 [cm] for the prototype. Some of them broke during handling. This lead us to define the appropriate branch radius to 4-7 [cm].

4.4 Structural Evaluation

The structure of the fabricated dome was evaluated with *Karamba Physics* using the experimentally acquired values from the flexural test. It is a rather simple beam model excluding details about the joints and variety of species of wood. The fixed supporting points are set at four corners. The input load was applied as in the right of Fig. 21. The result shows that the structure could take the maximum total load 2.5 [kN/mm²], causing a 19.2 [mm] displacement. The result from the analysis was experimentally verified by performing a crush test on the fabricated dome (Fig. 27). It broke at a load of 2.28 [kN] and with a displacement of 66 [mm] (see Fig. 28). There is quite large difference between the simulated model and the results of the crush test. The purpose for the structural test and simulation

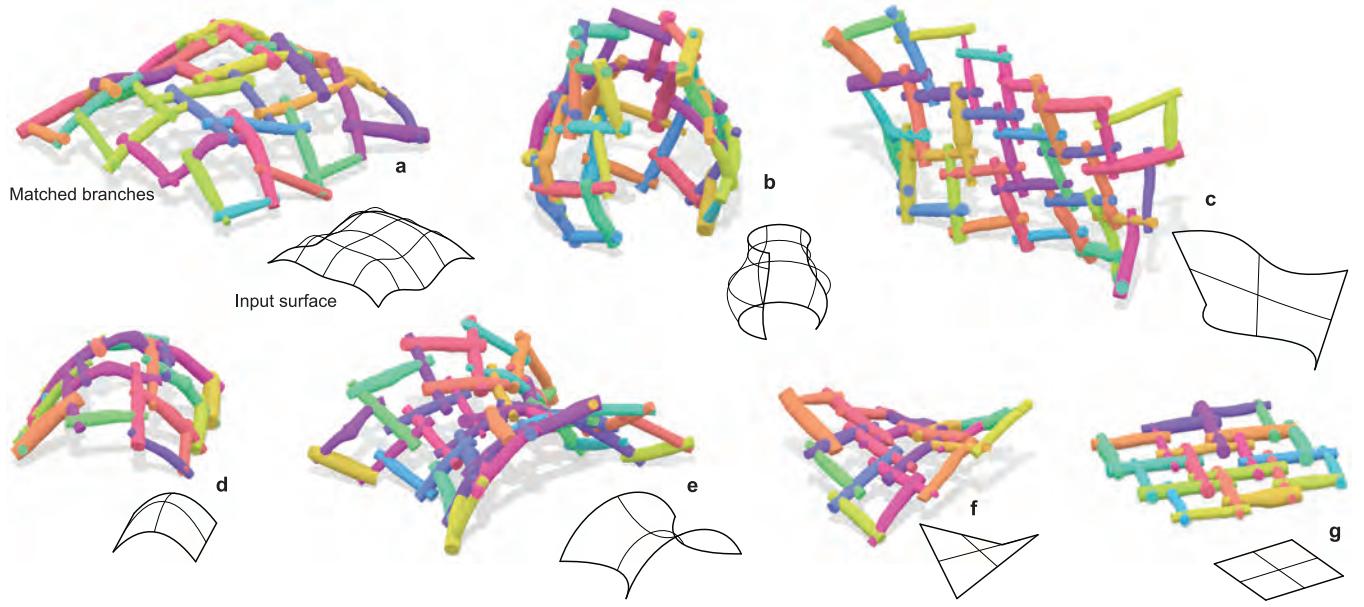


Figure 22: Different surfaces created from a set of 105 branches. Note: rendered results, not fabricated.

Table 1: Matching and relaxation evaluation of surfaces in Figure 22.

Surface	Number of Branches	Matching computation time	Relaxation simulation time	Joint Distance [mm]			
				Before Relaxation		After Relaxation	
				Mean	Max	Mean	Max
a. Quad dome	40	27 s	8 s	1.47	6.46	1.09	4.93
b. Cocoon	45	1 min 12 s	23 s	3.39	9.73	2.15	9.50
c. Sinus wall	45	1 min 36 s	10 s	0.97	3.54	0.78	2.82
d. Dome	24	36 s	3 s	1.97	8.66	2.18	7.53
e. Saddle arch	49	1 min 12 s	12 s	1.12	5.11	0.91	4.04
f. Bent sheet	24	41 s	4 s	0.82	3.57	0.73	2.68
g. Flat sheet	24	32 s	1 s	0.66	1.89	0.66	1.89

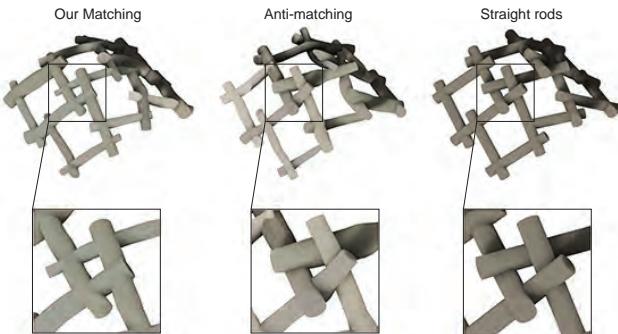


Figure 23: Our matching algorithm compared to other scenarios. In "Anti-matching" we pick the worst branch as opposed to the best one. In "Straight Rods" we use straight rods instead of curved branches.

is to show that the relatively fragile branches can make a relatively strong structure. There is a potential to improve the simulation and integrate the structure analysis feedback in the design process. However, we do not explore that direction.

5 LIMITATIONS AND FUTURE WORK

The matching algorithm has proven sufficient for our purposes. However, we have identified some possible improvements that could be interesting areas for future research. Firstly, the algorithm does not take into account how errors accumulate or cancel out. Secondly, we first divide longer branches into smaller segments before matching them with target curves. This simplifies the problem. But there is a potential to optimize the initial segmentation of longer branches to use for multiple target curves. Furthermore, as for the heterogeneous material properties of branches, we only consider the skeleton curvatures of the branches. There are, however, other characteristics that would be possible to include in the system, such as radius, color, and nodes. Branches with a larger

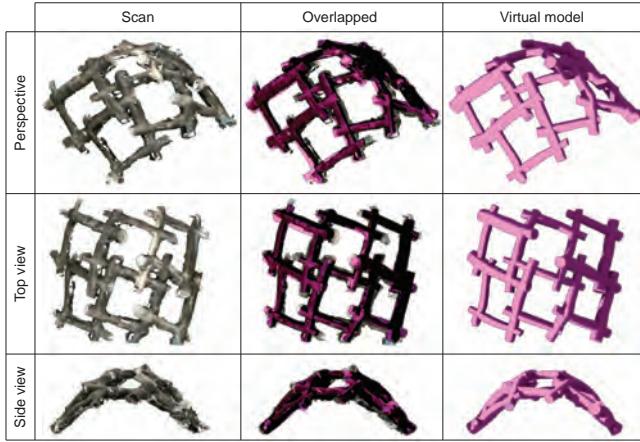


Figure 24: Scan of fabricated dome compared to virtual model.

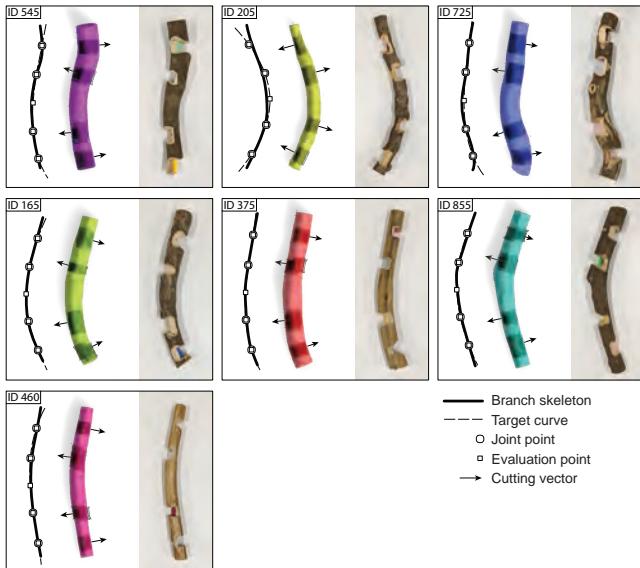


Figure 25: For selected branches (all with 4 joints) of the fabricated dome: branch skeleton overlapped with target curve, virtual model of branch with cutting vector and milling path, photograph of fabricated branch.

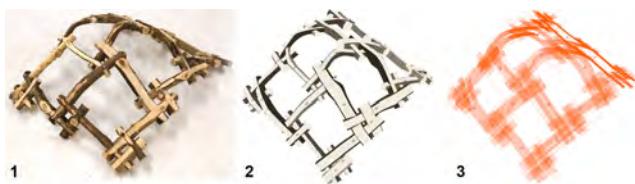


Figure 26: Prototype fabricated without audio-visual feedback. 1. Fabricated result. 2. Virtual model of planned structure. 3. Milling paths of joints and side planing.

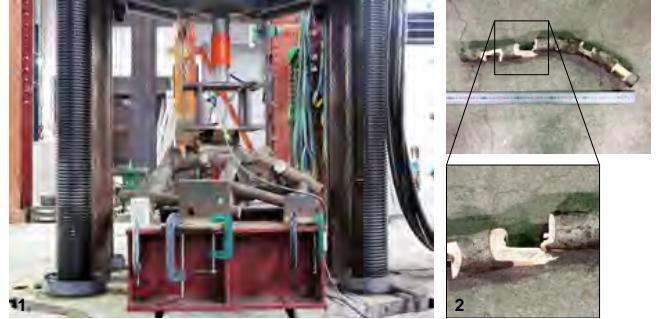


Figure 27: Crush test of fabricated dome. 1. Experimental setup. 2. Failure mode.

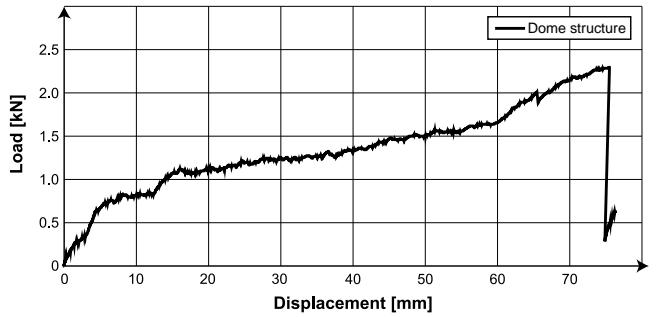


Figure 28: Crush test of dome structure. Load-displacement graph.

radius could potentially be placed at parts of the structure that take more load. And it would be desirable to avoid making a joint at a locally dense node.

We found that it was easy to collect tree branches, and to purchase them at a low cost directly from the forestry industry. But much effort is spent on material handling, i.e. to build and attach each marker cube, and to scan the branches one by one. For the system to scale well, these steps would need to be less of a burden. Scanning a bulk of branches at once, rather than one by one, is technically feasible. As for the AR markers, the process would be quicker if the marker cube was mass-produced instead of "home-made". Or even better, time and effort could be saved if it was possible to do marker-less tracking without compromising on accuracy.

The fabrication itself is rather time-consuming. Most of the time is spent for milling the joints. That is not unique to our system, but equally time-consuming with engineered wood. Using multiple CNC-machines parallel is one possibility to speed up the fabrication. As for the human task of positioning, it is rather quick and, we find, an enjoyable way to partake in the fabrication. The calibration steps after the positioning also rather quick but tedious. The user has to stand up and look closely at the branch from multiple angles to visually assess the position of the milling bit. This step would be easier if the user was assisted by a second camera capturing only the zoomed in area under the milling bit. Then the user could adjust the milling bit position while looking at a screen. With such engineering improvements of our prototypical system, it would be

more feasible to scale up production. As for the fabrication setup, we sometimes experienced a problem attaching the branch between the flat clamps of the vise. Sometimes we added small pieces of wood to find a stable position of the branch. The setup could be improved by using a vise with concave clamps, for example.

Considering the structural tests, the reciprocal pattern was effective to absorb the irregular nature of tree branches. The test with the fabricated dome showed more displacement than the simulated result. This result shows that the modeled dome was not accurate due to the continuous connections at nodes. The branches were weakened by milling out the joints as well. Further study on precise joint modeling is necessary. For architectural applications, there is an abundant diversity of reciprocal and other structurally sound patterns that we have left unexplored. For furniture applications, it would be possible to make furniture such as chairs and tables out of tree branches. It would just require a more diverse portfolio of joint types.

6 CONCLUSION

This paper presents a system that aims to upcycle abundant resources by digital scanning and computational fabrication, specifically focused on tree branches which are currently chipped and burned and not systematically used as a construction material. With our matching algorithm and human-in-the-loop fabrication, we have demonstrated that it is possible to build something geometrically precise and structurally sound out of tree branches, using just a standard 2.5D CNC-machine. Compared to standardized materials, fabricating with irregular material is challenging. Each building block is unique as opposed to interchangeable. To analyze the shape and position of the material, it needs to be scanned and pose-tracked. However, in a near future where scanning and pose-tracking technologies get more sophisticated, and as CNC machines becomes common-place in woodshops, along with an increased awareness of the environmental impact of material consumption, we believe that this project point at an interesting direction for computational fabrication.

ACKNOWLEDGMENTS

This research is supported by JST Act-I Grant Number JPMJPR17UT and JSPS KAKENHI Grant Number 17H00752. We would like to thank Shohei Furuchi and Kenchia Chang of Jun Sato Laboratory at the University of Tokyo for performing the crush tests and for structural consultation. We would further like to thank Georgina Cavalli for design consultation and Shohei Katakura for his support.

REFERENCES

- Begault DR, Wenzel EM, Shrum R, and Miller J. 1996. A Virtual Audio Guidance and Alert System for Commercial Aircraft. In *Proceedings of the Third International Conference on Auditory Display ICADâŽ96*, Kramer G (Ed.). Palo Alto, Calif.
- Brian F. G. Katz, Slim Kamoun, Gaëtan Parselhan, Olivier Gutierrez, Adrien Brilhault, Malika Auvray, Philippe Truillet, Michel Denis, Simon Thorpe, and Christophe Jouffrais. 2012. NAVIG: augmented reality guidance system for the visually impaired. *Virtual Reality* 16, 4 (01 Nov 2012), 253–269. <https://doi.org/10.1007/s10055-012-0213-6>
- H. W. Kuhn and Bryn Yaw. 1955. The Hungarian method for the assignment problem. *Naval Res. Logist. Quart* (1955), 83–97.
- Zachary Mollica and Martin Self. 2016. Tree Fork Truss: Geometric Strategies for Exploiting Inherent Material Form. In *Advances in Architectural Geometry 2016*, Sigrid Adriaenssens, Fabio Gramazio, Matthias Kohler, Achim Menges, and Mark Pauly (Eds.). 138–153.
- A. Ojamaa, V. Kotkas, M. Spichakova, and J. Penjam. 2013. Developing a lean mass customization based manufacturing. In *2013 IEEE 16th international conference on computational science and engineering (CSE)*. 28–33.
- Helmut Pottmann, Caigui Jiang, Matthias Höbinger, Jun Wang, Philippe Bompas, and Johannes Wallner. 2015. Cell packing structures. *Computer-Aided Design* 60 (2015), 70 – 83. <https://doi.org/10.1016/j.cad.2014.02.009> Material Ecology.
- Mikko J. Rissanen, Yoshihiro Kuroda, Naoto Kume, Megumi Nakao, Tomohiro Kuroda, and Hiroyuki Yoshihara. 2006. Audiovisual Guidance for Simulated One Point Force Exertion Tasks. In *Proceedings of the 2006 ACM International Conference on Virtual Reality Continuum and Its Applications (VRCIA '06)*. ACM, New York, NY, USA, 365–368. <https://doi.org/10.1145/1128923.1128989>
- Alec Rivers, Andrew Adams, and Frédéric Durand. 2012a. Sculpting by numbers. *ACM Transactions on Graphics (TOG)* 31, 6 (2012), 157.
- Alec Rivers, Ilan E Moyer, and Frédéric Durand. 2012b. Position-correcting tools for 2D digital fabrication. *ACM Transactions on Graphics (TOG)* 31, 4 (2012), 88.
- Christoph Schindler, Martin Tamke, Ali Tabatabai, Martin Bereuter, and Hironori Yoshida. 2014. Processing Branches: Reactivating the performativity of natural wooden form with contemporary information technology. *International Journal of Architectural Computing* 12, 2 (2014), 101–115.
- Peng Song, Chi-Wing Fu, Prashant Goswami, Jianmin Zheng, Niloy J. Mitra, and Daniel Cohen-Or. 2013. Reciprocal Frame Structures Made Easy. *ACM Transactions on Graphics* 32, 4 (2013), 10.
- Alan Song-Ching Tai. 2012. Design for assembly : a computational approach to construct interlocking wooden frames. *Thesis (S.M. in Architecture Studies)–Massachusetts Institute of Technology, Dept. of Architecture* (2012).
- Hironori Yoshida, Takeo Igarashi, Yusuke Obuchi, Yosuke Takami, Jun Sato, Mika Araki, Masaaki Miki, Kosuke Nagata, Kazuhide Sakai, and Syunsuke Igarashi. 2015. Architecture-scale Human-assisted Additive Manufacturing. *ACM Trans. Graph.* 34, 4, Article 88 (July 2015), 8 pages. <https://doi.org/10.1145/2766951>
- H. Yoshida, Larsson M., and T. Igarashi. 2019. Upcycling Tree Branches as Architectural Elements through Collaborative Design and Fabrication (forthcoming). In *TEI '19—Proceedings of the Twelfth International Conference on Tangible, Embedded, and Embodied Interaction*. Arizona, USA.
- Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. 2018. Open3D: A Modern Library for 3D Data Processing. [arXiv:1801.09847](https://arxiv.org/abs/1801.09847) (2018).
- Amit Zoran and Joseph A. Paradiso. 2013. FreeD: A Freehand Digital Sculpting Tool. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, New York, NY, USA, 2613–2616. <https://doi.org/10.1145/2470654.2481361>