

1      **Hybrid Drawing Solutions in AR: Bitmap-to-Vector Techniques on 3D Surfaces**

2  
3      PENGCHENG DING and YEDIAN CHENG, Northeastern University, Canada

4  
5      Recent advancements in augmented reality (AR) and virtual reality (VR) have significantly enhanced workflows for drawing 3D objects.  
6      Despite these technological strides, existing AR tools often lack the necessary precision and struggle to maintain quality when scaled,  
7      posing challenges for larger-scale drawing tasks. This paper introduces a novel AR tool that uniquely integrates bitmap drawing and  
8      vectorization techniques. This integration allows engineers to perform rapid, real-time drawings directly on 3D models, with the  
9      capability to vectorize the data for scalable accuracy and editable points, ensuring no loss in fidelity when modifying or resizing the  
10     drawings. We conducted user studies involving professional engineers, designers, and contractors to evaluate the tool's integration  
11     into existing workflows, its usability, and its impact on project outcomes. The results demonstrate that our enhancements significantly  
12     improve the efficiency of drawing processes. Specifically, the ability to perform quick, editable, and scalable drawings directly on 3D  
13     models not only enhances productivity but also ensures adaptability across various project sizes and complexities.

14  
15     CCS Concepts: • Human-centered computing → Augmented reality; • Applied computing → Computer-aided design; •  
16     Computing methodologies → Graphics systems and interfaces.

17  
18     Additional Key Words and Phrases: Augmented Reality, Drawing, Industrial Design, Unity, Microsoft HoloLens 2

19  
20     **ACM Reference Format:**

21  
22     Pengcheng Ding and Yedian Cheng. 2024. Hybrid Drawing Solutions in AR: Bitmap-to-Vector Techniques on 3D Surfaces. 1, 1  
23     (August 2024), 23 pages. <https://doi.org/10.1145/nnnnnnn.nnnnnnn>

24  
25     **1 INTRODUCTION**

26  
27     Visualizing complex objects in two dimensions is limiting, as it restricts our ability to perceive accurate spatial  
28     relationships and dimensions. Immersive technologies such as Virtual Reality (VR) and Augmented Reality (AR) are  
29     increasingly being integrated into professional workflows to address these limitations. These technologies enhance  
30     our ability to visualize 3D objects with greater detail and precision, providing experiences that mimic interacting with  
31     physical objects but without the constraints of physical space.

32  
33     Currently, VR and AR technologies are becoming significant in fields such as engineering. Advancements in  
34     photogrammetry[20, 33] have enabled designers to directly create three-dimensional models from photographs, trans-  
35     forming traditional two-dimensional workstations into immersive three-dimensional environments. Augmented Reality,  
36     in particular, facilitates the projection of 3D models into the real world, providing a novel mode of interactive visualiza-  
37     tion that effectively merges virtual objects with the physical environment. This capability significantly enhances the  
38     engineering process by allowing professionals to view and manipulate 3D models in real-time[15], thereby ensuring  
39     more precise outcomes. Consequently, there is a growing demand in the technical community for capabilities such as

40  
41     Authors' address: Pengcheng Ding, ding.pen@northeastern.edu; Yedian Cheng, cheng.yed@northeastern.edu, Northeastern University, Vancouver,  
42     British Columbia, Canada.

43  
44  
45     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  
46     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  
47     of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to  
48     redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

49     © 2024 Association for Computing Machinery.

50     Manuscript submitted to ACM

51  
52     Manuscript submitted to ACM

53 direct drawing on 3D models in augmented reality, highlighting the need for tools that can perform precise annotations  
 54 and drawing on 3D objects to enhance detail-oriented tasks.  
 55

56 Research in AR and VR sketching has developed over time, and a wealth of valuable research already exists, allowing  
 57 us to delve deeper into immersive sketching with these technologies. Early systems like HoloSketch[12] pioneered 3D  
 58 modeling but struggled with more structured and detailed creation . Technologies such as Tilt Brush[40] improved  
 59 control but still lack functionalities like stroke capture[3, 23–25, 37]. Efforts like FreeDrawer[43] and haptic feedback  
 60 integration[26, 27] aim to enhance precision. Arora et al.[3] highlighted the challenges of accurately creating curves  
 61 on virtual surfaces in VR, even on simple planar surfaces. Furthermore, hybridizing 2D and 3D sketching, as seen in  
 62 PintAR[18, 19] and TabletInVR[39], improves 3D surface drawing accuracy. SymbiosisSketch[2] and VRSketchIn[16]  
 63 combine air-pen and tablet inputs, but managing complex surfaces like spheres remains difficult. Advanced projection  
 64 techniques, used in SecondSkin[11], Skippy[29], and CASSIE[45], align drawn trajectories to object surfaces, improving  
 65 mid-air drawing accuracy[4].Using the vector method to draw in has also made some progress. A key study called  
 66 VTracer[36] is a tool that allows for the transformation of raster images into vector graphics. By leveraging this frame  
 67 field, the algorithm creates vectorizations that align closely with the original raster images. These researches are  
 68 priceless resources for us to start our research.  
 69

70 Even with these improvements, modern AR technologies frequently fail to satisfy the particular requirements of  
 71 engineers. No drawing technique currently in use can meet both the needs of quick sketching and preserving accuracy  
 72 and editability in large-scale models.  
 73

74 To overcome the shortcomings of existing augmented reality technologies, our approach builds a custom AR  
 75 framework for engineering applications by combining the immersive features of the Microsoft HoloLens 2[31] with an  
 76 advanced hybrid drawing tool built on Unity. This drawing tool uniquely combines the advantages of rapid sketching  
 77 with the scalability and editability of vector graphics.  
 78

79 In summary, our contributions are as follows:  
 80

- 81   • We have developed a tool for rapid drawing on 3D model surfaces in AR. This innovative tool allows for initial  
   82 bitmap sketches to be made directly within the AR environment, facilitating immediate and intuitive interaction  
   83 with 3D models.
- 84   • To preserve clarity, editability, and scale, we enable the vectorization of sketches. After the initial sketching,  
   85 our tool can vectorize these sketches, ensuring that the drawings maintain their editable, detail, and scale.
- 86   • We evaluated the tool’s usability and efficiency in real-world circumstances. This study contributed to uncover  
   87 practical issues and user preferences, which are critical for iterative development.  
 88

89 This novel framework revolutionizes the conventional drawing process by facilitating a smooth transition from  
 90 bitmap to vector. This conversion enhances the utility of drawings throughout various phases of the engineering  
 91 design process, making rapid prototyping and modifications more manageable. Moreover, our hybrid drawing approach  
 92 ensures that drawings maintain accuracy and precision even at large scales, which is particularly vital for large-scale  
 93 engineering projects where precision and detail are critical.  
 94

## 95   2 RELATED WORK

96 Our work builds upon a substantial body of prior research in the areas of immersive sketching and modeling, integration  
 97 of 2D and 3D sketching techniques, enhancing 3D modeling through projection techniques, and integrating bitmap  
 98

105 drawing and vectorization process for enhanced precision in 3D AR Environments. This diverse foundation informs  
106 our approach to developing advanced AR tools for engineering applications.  
107

## 108 2.1 Immersive sketching and modeling:

109 Immersive sketching and modeling have rapidly evolved through advancements in VR and AR technologies. Introduced  
110 in 1995, HoloSketch[12] utilized a 6-degree-of-freedom wand for creating basic 3D shapes, setting a foundational stage  
111 for interactive 3D modeling. Despite its innovation, early systems like HoloSketch often suffered from poor motor  
112 control, leading to imprecise and noisy mid-air drawings. Subsequent systems have further developed these initial ideas,  
113 delving into the creation of freeform 3D curves and swept surfaces[25, 37]. Technologies like Google's Tilt Brush[21]  
114 have built on these foundations, offering enhanced control and the ability to transform direct 3D inputs into detailed  
115 creative outputs. However, these systems still struggle with the inherent inaccuracies of 3D sketching, especially when  
116 detailed, structured creation is needed. Current immersive tools lack functionalities like stroke capture and real-time  
117 feedback, visual depth cues, and motion parallax, which are standard in traditional 2D sketching[3, 23, 24]. Efforts  
118 like FreeDrawer[43] have tried to improve precision by regularizing 3D curves to existing geometries, and some have  
119 integrated haptic feedback to refine user interaction[26, 27]. Arora et al.[3] highlighted the challenges of accurately  
120 creating curves on virtual surfaces in VR, even on simple planar surfaces. This finding underscores the need for our  
121 research into techniques that project 3D strokes onto surfaces.  
122

## 123 2.2 Integration of 2D and 3D sketching techniques:

124 Enhancing the precision of 3D surface drawings can be significantly advanced through the innovative hybridization of  
125 2D and 3D sketching techniques. Gasques et al.[18, 19] identified the need for this hybrid sketching approach in their  
126 PintAR system, which provides a pen and tablet interface without mid-air interactions, focusing on a more grounded  
127 drawing experience. Similarly, TabletInVR [39] is a recent initiative that creates a design space for VR 3D solid modeling  
128 based on tablet inputs, demonstrating the practical application of tablet-based controls in immersive environments.  
129 SymbiosisSketch[2] combines freeform air-pen sketching in AR with pen-and-tablet inputs, facilitating sketches on  
130 both flat and curved surfaces within AR, thereby enhancing interaction with real-world objects. However, a significant  
131 limitation emerges when the scene exceeds a basic canvas, such as when dealing with continuous surfaces like spheres.  
132 This complexity is highlighted by the challenges of extending beyond traditional surface patch topology to accommodate  
133 surfaces with multiple boundaries or closed forms like spheres. VRSketchIn[16] further explores this concept within its  
134 design space, categorizing prior art and defining metaphor groups of sketching interactions for VR. This system was  
135 implemented and evaluated through usability walkthrough with participants, emphasizing its primary application in  
136 artistic creation rather than technical tasks. The findings suggest that while 2D and 3D integration is beneficial for  
137 artistic endeavors, it remains challenging to manage on larger and continuous surfaces like spheres.  
138

## 139 2.3 Enhancing 3D Model Drawing through Projection Techniques

140 Advanced projection techniques in 3D modeling play a crucial role in enhancing the quality of digital representations by  
141 mapping intricate contours onto object surfaces. Techniques such as SecondSkin[11] and Skippy[29] utilize insights into  
142 the spatial relationships between drawn strokes and the 3D object, inferring 3D curves that accurately follow the object's  
143 surface. In these models, drawn trajectories can be aligned to predefined lines[46] or planes [30], or integrated into 3D  
144 curve networks[45]. Furthermore, CASSIE[45] has developed a 3D optimization approach that enables the automatic  
145 connection and predictive smoothing of curves, enhancing the user's ability to create more coherent 3D structures.  
146

Graffiti-style painting with a spray can is widely recognized as a standard practice in commercial immersive paint and sculpt software like Medium[1] and Gravity Sketch[22]. This method often employs a closest-point projection that simulates drawing on the 3D object without direct physical contact, similar to the method used by the "guides" tool in Tilt Brush[21]. While existing approaches handle each mid-air point separately and lack context, more recent developments, such as those reported by Arora et al.[4], present imitation methods that use anchored projection techniques. However, the mimicry approach uses a 3D mesh builder that attaches volumetric blocks to our model's surface, leading us to opt for Texture Painting[38] instead. While mimicry forms a foundation for our drawing techniques, it struggles with engineering applications that require frequent model scaling. These limitations prompt further consideration and refinement of our techniques.

## 2.4 Integrating bitmap drawing and vectorization for Enhanced Precision in 3D AR Environments

In 2018, Dudley et al.[17] proposed a method for drawing vector lines in 3D mid-air, using a technique known as 'Tapline', which involves plotting points in space but is quite time-consuming. Our solution requires a method for directly vectorize curves on 3D surfaces. Existing 2D vectorization methods developed by Noris et al.[35] and Donati et al.[13, 14] often use 1-skeleton or raw image gradients to highlight important drawing features such as curve endpoints and junctions[7, 13, 14, 34, 35]. Although these methods are effective in clean line drawings, they can suffer from geometric and topological errors under noisy conditions. The introduction of the PolyVector Flow method by Puahachov et al.[6] marks a significant improvement by efficiently converting raster images to vector formats, focusing on tasks such as identifying crucial points, extracting topology, and determining optimal geometries. However, it is time-consuming and primarily suitable for line art. The VTracer method[36] integrates these processes by initially clustering the input image using Hierarchical Clustering[41], then tracing each cluster into a vector. The vector tracing algorithm involves converting pixels into paths, simplifying these paths into polygons, and smoothing the polygons with a curve-fitting technique. This approach is particularly adept at handling complex junctions in noisy environments and underpins our strategy to enhance 3D vectorization. By extending these principles to 3D, our project aims to improve the accuracy and quality of drawing in AR. After the vectorization process, the line records stroke data as mathematical vectors, enabling tasks such as curve adjustments and maintaining clarity and quality when scaled. This makes it suitable for use in other engineering software.

## 2.5 Semi-Structured Interviews as Qualitative Metrics

Semi-structured interviews play a critical role in qualitative research, enriching the data gathered with insights and depth not typically accessible through quantitative methods alone. These interviews typically employ a two-tiered approach to questioning: main themes and further follow-up questions. The main themes serve as the framework for the conversation, addressing the major questions of the research and enabling participants to openly express their opinions and experiences. It is customary to engage every participant in these primary themes[47]. The sequence of these themes is often designed to progress in a logical and orderly manner[28], starting with more general questions to ease participants into the conversation and establish a comfortable atmosphere[10, 44]. The initial questions typically touch on areas familiar to the participants but crucial to the research topic[44]. As the interview progresses, the discussion deepens, moving towards more comprehensive topics[5, 10, 44] before returning to less intense themes towards the end[5]. By incorporating semi-structured interviews into our study, we are hoping to gather an extensive variety of information, from user interactions with technology to the subjective experiences that demonstrate its applicability and user approval.

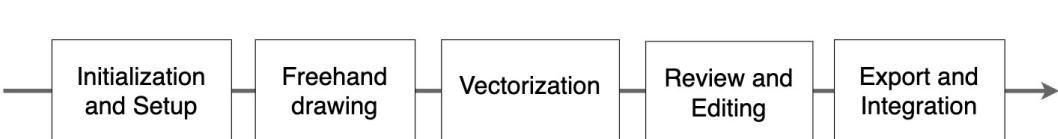


Fig. 1. Draw2Vector Workflow

### 3 SYSTEM DESIGN

To meet the demands of drawing on 3D geographical models, we have developed hand-interaction-based drawing techniques utilizing the Microsoft HoloLens2[31]. This augmented reality headset, equipped with depth cameras and sensors, allows for precise environmental sensing. Our system enables users to draw freely by projecting their fingertips' positions onto 3D surfaces using a ray. The resultant drawings are then vectorized by tracking the points in the stroke data to form vectorized lines, which are then smoothed. This integration is powered by the Unity game engine[42], which facilitates a smooth interaction between drawing functionalities and 3D modeling. The system's innovation lies in its capability to vectorize 3D stroke data. The bitmap drawing method enables users to swiftly sketch designs or annotate directly on models, and the vectorized stroke is scalable and editable, ensuring no loss of detail upon scaling or exporting.

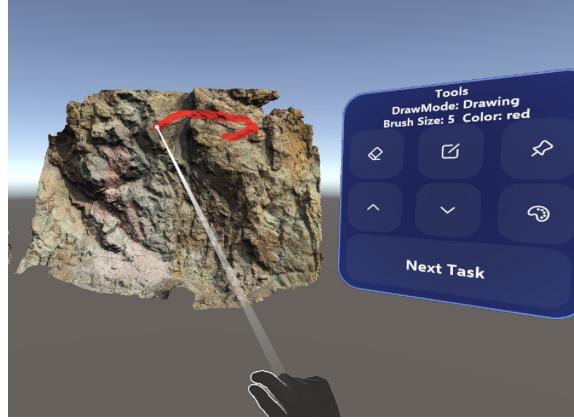
#### 3.1 Workflow Description

The workflow is optimized for user efficiency and accuracy, encompassing several key phases to ensure a smooth operation from start to finish:

- **Initialization and Setup:** To begin, users load a 3D model from the application's cloud-based library. The HoloLens2 headset runs system checks and measurements to ensure proper tracking and rendering based on the user's location.
- **FreeHand Drawing:** Users perform drawing tasks on the surface of 3D scanned models using gesture controls, providing an immersive experience. The system supports various function like changing the size and color of the brush, accessible via a UI panel in the AR view.
- **Vectorization:** Users can convert their freehand sketches into vectorized strokes by clicking the 'Vectorization' button on the UI panel. This process optimizes the paths of the sketches, enhancing clarity and reducing noise to produce high-quality 3D vectorized stroke.
- **Review and Editing:** If the vectorized strokes are not satisfactory, they can readily edit the vectorized drawings. This editing phase includes editing the key points of the strokes, allowing users to achieve the goal they desire for the final output.
- **Export and Integration:** Completed drawings are exported as a CSV file that records the geographic coordinates of each point and includes a texture map of the model. These files are suitable for use in other applications for further editing and refinement.

Each of these steps is designed to ensure that the system is not only user-friendly but also capable of producing high-quality outputs that are useful in professional settings. Next, we will introduce the two most important steps in

261  
262  
263  
264  
265  
266  
267  
268  
269  
270  
271  
272  
273  
274  
275



276  
277  
278

Fig. 2. User interface of Draw2Vector

279 our workflow: Free-hand drawing and the Vectorization process. This workflow 1 provides a visual summary of the  
280 entire process.

281  
282

### 3.2 User interface

283  
284  
285  
286  
287  
288  
289  
290  
291

Upon loading the model, the system automatically enters the freehand (bitmap) drawing mode. A small red sphere, representing the user's hand position, appears on the surface of the 3D model. Drawing begins with a pinch gesture in the air. As the user performs this gesture and move his hand, the red sphere moves along the model's surface, tracing the strokes. The drawing process concludes when the gesture is released. Users can navigate the red sphere to the right-side UI panel2 and pinch again to activate the panel, which appears directly in front of the user. This panel provides several options, including adjusting brush color and size, clearing the canvas, vectorization, and changing the drawing mode.

292  
293  
294  
295  
296

If the user selects the vectorization option, the current freehand drawing is converted into vectorized lines. Subsequently, the sphere changes to green, indicating the Tapline mode (vector drawing mode). If users are unsatisfied with the lines, they can adjust the position of points on the line or continue to draw using vector techniques by placing points on the figure that automatically connect into lines.

297  
298  
299  
300  
301

It's straightforward for users to stay updated with their current settings because the top of the UI banner displays the drawing mode and brush size. The user also has the option to switch from freehand drawing to Tapline mode if they would like to modify the drawing mode. Their work is automatically saved and exported after the drawing session is finished.

302

### 3.3 Freehand Drawing: Quick sketch on the surface of 3D scanned models

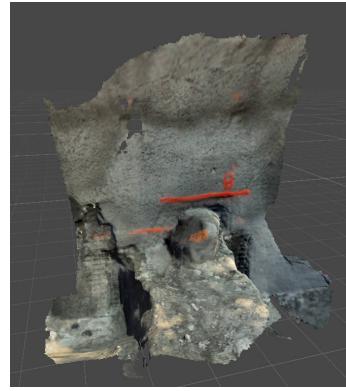
303  
304  
305  
306  
307  
308  
309

Freehand Drawing is specifically designed for rapid sketching directly on 3D models. This function is extremely useful in a collaborative scenario like a virtual meeting. In such scenario, the host can show to other participants easily where they are going to modify. This tool can also be used as an annotation tool, which allows engineers to point out where needs to be, as shown in this photo3, excavated.

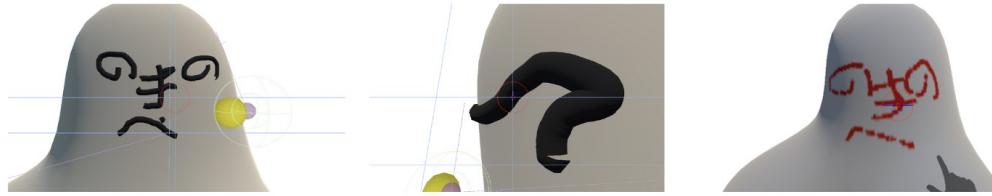
310  
311  
312

**Mimicry method:** In this research project, we will use a context-based drawing approach called *mimicry*[4], this method is called *mimicry* because the user tends to **mimic** the shape of their intended curve on surface. Such

313  
314  
315  
316  
317  
318  
319  
320  
321  
322  
323  
324  
325  
326



327 Fig. 3. Making annotation  
328  
329  
330



340 Fig. 4. Comparison between Mimicry - 3D mesh builder(left) and Texture Painting (right)  
341  
342

343 drawing algorithms are used to project 3D curves, in our research, the stroke on 3D surface. Traditional projection  
344 algorithms *Spraycan* method tend to have the limitation that can't capture the model curve, especially when models have  
345 complicated, layered surfaces. So here we will introduce the technical description of this method. The core principle  
346 of the mimicry method is to accurately map the trajectory and curvature of freehand strokes made in mid-air onto a  
347 virtual object's surface. The mathematical framework involves the projection of stroke points onto the 3D model and  
348 optimization of the resulting curve for smoothness and continuity.  
349  
350

- 351 • **Projection of Stroke Points:** For subsequent points  $p_i$  (where  $i > 0$ ) of the stroke in mid-air, each point is  
352 projected onto a corresponding point  $q_i$  on the surface  $S$  of the 3D model. In our research, we utilize both the  
353 previous stroke data  $p_{i-1}$  and the previous projection  $q_{i-1}$  to enhance the projection accuracy. This process is  
354 mathematically defined by the following equations:  
355

$$356 \quad r_i = q_{i-1} + \Delta p_i,$$

357 where  
358

$$359 \quad \Delta p_i = p_i - p_{i-1}$$

$$360 \quad q_i = \arg \min_{x \in S} \|r_i - x\|$$

365  
366  
367  
368  
369  
370  
371  
372  
373  
374  
375  
376  
377  
378  
379  
380  
381  
382



Fig. 5. Texture Painting method  
[38]

383 where  $\|r_i - x\|$  is the Euclidean distance between  $r_i$  and  $x$ .  
384

385  
386 **Texture Painting Method:** During the implementation of our project, we encountered specific challenges with the initial  
387 implementation of the mimicry method for freehand drawing on 3D models. Originally, the mimicry approach relied  
388 on a 3D mesh builder for drawing, which attaches a volumetric block to the surface (Figure 4), is unsuitable for our  
389 project's needs. To address this, we have shifted to using Texture Painting [38] (Figure 5).

390 Texture Painting is a technique we have implemented to enhance the drawing capabilities within our project. This  
391 method involves creating separate 2D image files for each layer, which are then bound to the object's existing texture  
392 and material (Mtl) files. This integration allows for the textures to be updated in real-time and adhered directly to  
393 the 3D surfaces, significantly enhancing both the flexibility and performance of the drawing tool. Texture Painting  
394 leverages shaders to dynamically apply colors or textures to a mesh based on real-time user interactions [38], such as  
395 clicking or dragging over the mesh surface. The shader used in this process is designed to project a texture onto a mesh  
396 while modifying the texture's pixels directly on the mesh. This allows for immediate visual feedback and artistic control,  
397 enabling users to see the effects of their modifications instantly. This capability is crucial for applications requiring  
398 high levels of interaction, as in our project where responsive freehand drawing on 3D models is essential.

399  
400 **Handling Discontinuous Lines During Rapid Drawing:** To address the issue of discontinuous lines when  
401 drawing quickly, we implemented an improvement in our freehand drawing method. If the distance between points  
402 from consecutive frames is greater than the brush size, the line will appear discontinuous. To solve this, if the distance  
403 between the points is greater than the brush size, we connect these points by projecting the line segment onto the  
404 3D object's surface texture, ensuring continuous lines. This enhancement ensures that rapid strokes are accurately  
405 represented as continuous lines, maintaining the integrity of the drawing experience.

406  
407 **Summary:** We initially implemented the Projection of Stroke Points technique from the mimicry method, which  
408 utilizes a 3D mesh builder. This approach effectively attaches a volumetric block to the surface, making it less suitable  
409 for engineering tasks such as drawing, annotating, and coloring. Therefore, we transitioned to using texture painting  
410 for rendering our drawings. This adaptation allows for a more appropriate application of markings and colors directly  
411 onto the 3D model's surface, enhancing the usability of our tool in engineering contexts.

### 417 3.4 Vectorization Method Evolution

418 Initially, the project used the *VTracer* method [36] to transform hand-drawn movements into vector strokes. Originally  
419 designed for 2D graphics, *VTracer* rapidly converts strokes to vectorized data, ensuring scalability without loss of  
420 information.

421 The vector tracing process begins by applying Hierarchical Clustering[41] to the input image, which segments it into  
422 distinct clusters. Each cluster is then converted into vector format through a three-step algorithm: First, the pixels within  
423 each cluster are transformed into a continuous path. Next, this path is simplified into a polygon by minimizing the  
424 number of vertices to maintain the shape's integrity. Finally, the polygon is smoothed and refined using a curve-fitting  
425 technique to enhance the visual quality and accuracy of the resulting vector image.

426 We implemented this vectorization technique specifically targeting the texture images of 3D models. The process  
427 begins by mapping sketches directly onto the separate texture layers of these models. Once mapped, these texture  
428 images go through a detailed vectorization process. This involves converting the 2D representations of the textures  
429 into vector graphics. The converted vector graphics are then re-applied to the 3D models and subsequently evaluated  
430 within an augmented reality (AR) environment. This two-phase approach—first vectorizing the textures in 2D and  
431 then integrating them back onto the 3D models—ensures that the vectors retain true to the model's detailed structure  
432 when viewed in AR. This method allows us to maintain precision in how textures map onto the complex contours and  
433 features of the 3D models, ensuring that the final visualizations in AR are both accurate and visually coherent.

434 However, after implementing this method, we identified significant limitations.

- 435 • Primarily, the technique outlines images rather than capturing them as continuous, dynamic vector lines.  
436 Consequently, while the graphics accurately trace the external boundaries of shapes, they do not reflect the  
437 true nature of hand-drawn strokes.
- 438 • Moreover, the current vectorization technique does not support modifications in an AR environment, restricting  
439 its interactive potential, which is crucial for real-time applications and field adjustments.

440 **Refinement of the 3D Vectorization Process:** The initial shortcomings of the *VTracer* method led to a comprehensive  
441 revision of our vectorization approach. To better preserve the intrinsic qualities of hand-drawn strokes, we  
442 developed an advanced vectorization technique that incorporated real-time tracking of the drawing's 3D coordinates.  
443 This enhancement allowed for a more nuanced capture of each stroke's dynamic properties, ensuring that the vectorized  
444 outputs retained the fluidity and expressiveness of the original sketches.

- 445 (1) **Tracking 3D Coordinates:** During live drawing sessions, we begin our enhanced vectorization process by  
446 properly recording the 3D coordinates of points as they are projected onto a curve. This tracking occurs in real  
447 time and captures every detail of the user's drawing motion. By recording these locations at regular times, often  
448 every few milliseconds, we assure a detailed description of the drawing trajectory, which is used to generate  
449 exact 3D vectors<sup>6</sup>(Figure 6).
- 450 (2) **Joining Points to Form Discrete Vectors:** Once the points along the curve have been traced, the next step is  
451 to combine them to form discrete three-dimensional vectors<sup>7</sup>(Figure 7). This procedure converts the sequence  
452 of points into a collection of vectors that follow the route of the original drawing in 3D space. Although this  
453 results in a segmented line, it is an important initial step in developing a continuous vector representation.
- 454 (3) **Path Walking:** To improve the continuity between these discrete vectors, we use a technique called Path  
455 Walking<sup>8</sup>(Figure 8). This method focuses on merging vectors that are all in the same direction. By combining  
456 these aligned vectors, Path Walking eliminates abrupt directional changes and contributes to a more unified and  
457 smooth final vector representation.

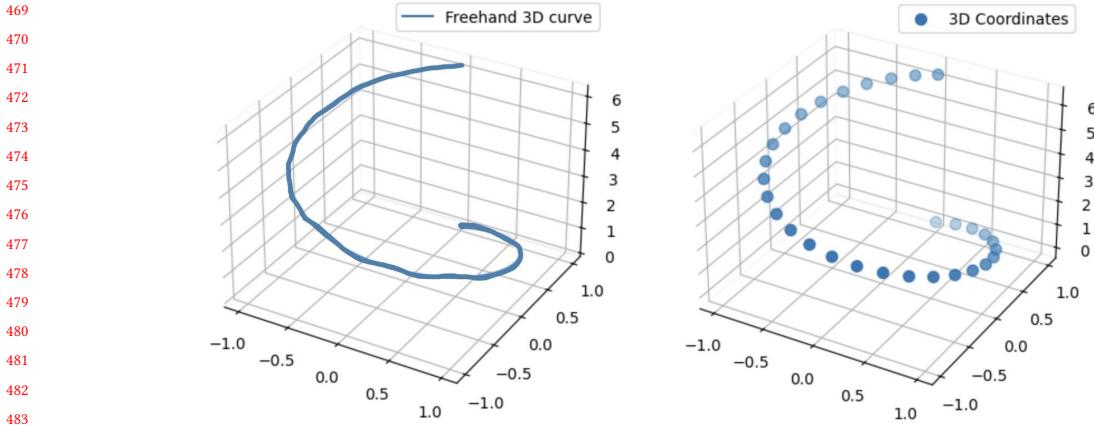


Fig. 6. Tracking 3D Coordinates: Left - Continuous Freehand Drawing; Right - 3D Points Capturing the Stroke Path

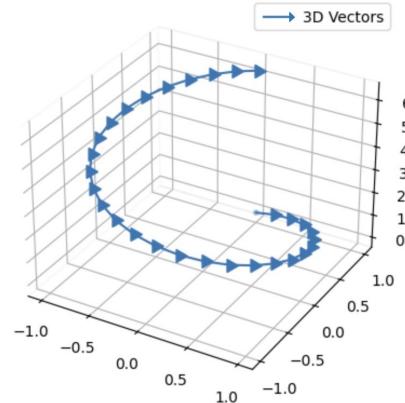


Fig. 7. Joining Points to Form Discrete Vectors

flowing vector path. This approach is essential for preserving the original drawing's integrity and smoothness in vectorized form.

- (4) **Path Smoothing:** The final phase in our method is path smoothing, depicted in Figure 99, which refines the vector path to produce a smoother, more continuous line. Initially, we employed Chaikin's Corner Cutting Algorithm[9]. However, we observed that repeated iterations of this algorithm resulted in significant deviations. Consequently, we shifted to using Catmull-Rom Interpolation Algorithm[8]. This method ensures that the curve smoothly passes through the control points, thus maintaining the fidelity of the path. It is described by the following equation:

$$P(t) = 0.5 \left( (2p_1) + (-p_0 + p_2)t + (2p_0 - 5p_1 + 4p_2 - p_3)t^2 + (-p_0 + 3p_1 - 3p_2 + p_3)t^3 \right)$$

The parameter  $t$  represents the interpolation factor between points, ranging from 0 to 1, with  $t = 0$  at  $p_1$  and  $t = 1$  at  $p_2$ . This parameterization allows for precise control over the interpolation process, enabling adjustments

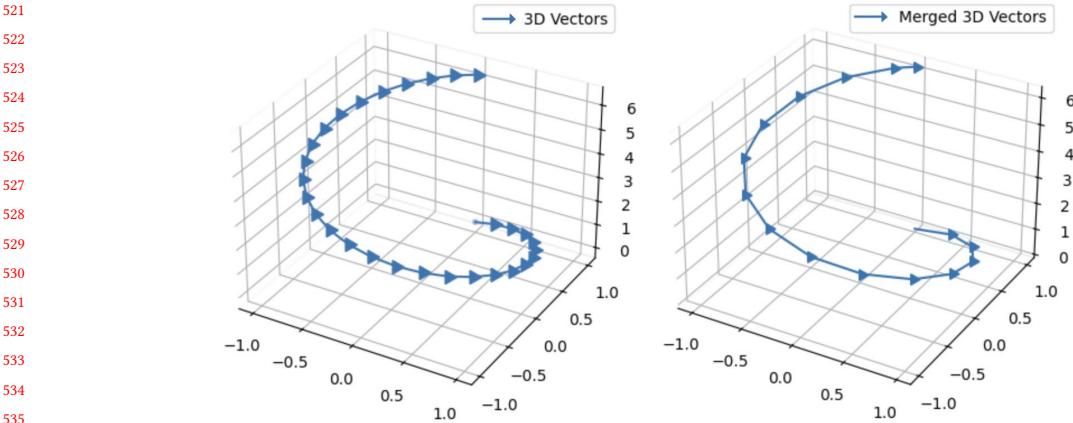


Fig. 8. Path Walking: Left - 3D Vectors; Right - Merged 3D Vectors

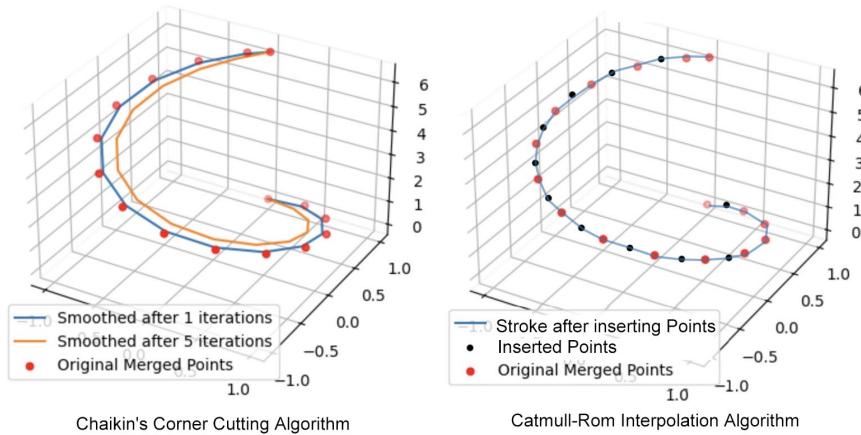


Fig. 9. Left: Significant deviation resulting from Chaikin's Corner Cutting Right: Shift to using Catmull-Rom Interpolation Algorithm

for greater accuracy or visual appeal in the resultant vector path. In our application, given the scale of our drawings, we currently default to inserting only one point between each pair of points. However, this parameter can be adjusted to achieve greater precision if needed.

- (5) **Path Projection:** When constructing a continuous line on the surface between two points, we can utilize the following procedure to ensure accuracy and continuity:
- **Plane Construction:** Given two points and the Raze direction (the ray pointing from the finger to the model), we construct a plane. The raze direction, along with the two points, helps define this plane.
  - **Intersection Calculation:** Using this plane, we calculate the intersection line. The raze direction and the two points form a triangle. The intersection of this triangle with the 3D model surface determines the path we need to project. By identifying the intersecting segment of this triangle with the model's surface, we can accurately color this segment, ensuring a continuous and seamless line on the 3D object.

### 573      3.5 Implement details

574  
 575 Our development is based on the official MRTK(Mixed Reality toolkit) template[32], and our techniques are implemented  
 576 in C# and Shader, utilizing the Unity Engine for interaction, rendering, and VR support, including integration with  
 577 Hololens2. During the freehand drawing phase, we combine the mimicry method[4] and texture painting techniques[38].  
 578 In the vectorization stage, we employ the key idea of the VTracer method[36], such as path walking, and we utilize  
 579 Catmull-Rom Interpolation Algorithm[8] to achieve path smoothing. This comprehensive approach enhances the  
 580 usability of 3D drawing, significantly improving collaboration and design processes in mixed reality environments.  
 581

## 582      4 USER STUDY

583 We conducted a user study to evaluate and compare the performance of data from the freehand drawing phase and  
 584 post-vectorization, as well as user experience with the system. The study involved developers and engineers, aiming to  
 585 enable participants to perform drawing tasks on predefined 3D models, observe their workflows, and store the results.  
 586

### 587      4.1 Participants

588 We recruited 10 participants (5 female, 5 male) with a mean age of 38.75 (SD = 13.34). The participants were a mix of  
 589 engineers, who are the actual users of the system, and software developers, who provided valuable feedback on the  
 590 system's usability and functionality. Six of these participants reported prior experience with the HoloLens 2, which was  
 591 used for the experiments.

### 592      4.2 Procedures

593 Participants began the experiment with a briefing that introduced them to the AR headset and explained the drawing  
 594 techniques. They then completed a familiarization task to ensure comfort and proficiency with the system. Subsequently,  
 595 they engaged in a series of drawing tasks designed to evaluate the practical application and effectiveness of the techniques.  
 596 After the freehand drawing phase, the drawing data were stored. After that, participants used a vectorization function  
 597 to convert bitmap strokes into vectors. If participants were unsatisfied with the vectorized lines, they could then adjust  
 598 the points of the lines. Data was collected again. Participants provided feedback through semi-structured interviews  
 599 and questionnaires, assessing user satisfaction, ease of use, and system responsiveness, among other factors. Following  
 600 the experiment, we compared bitmap and vectorized lines in terms of line smoothness, accuracy, data size and so on.  
 601 This method enabled a full evaluation of both technical performance and user experience, which enabled an extensive  
 602 examination of the system's effectiveness.

### 603      4.3 Training and familiarisation task

604 Participants were given a thorough training that introduced the AR headset and shown how to use the platform and its  
 605 drawing skills. Before starting the main tasks, participants completed a gesture familiarization exercise involving a  
 606 rock model dotted with ten points. Participants were instructed to connect these dots in a continuous line, practicing  
 607 multiple times until they could complete the task within 20 seconds.

### 608      4.4 TASK1: Draw a Simple Rock Fault Line

609 Participants will be presented with a 3D rock model displaying a prominently marked geological fault line[11], originally  
 610 delineated by engineers in a digital model. This thick line represents a fault and serves as a visual guide to clarify the  
 611 Manuscript submitted to ACM

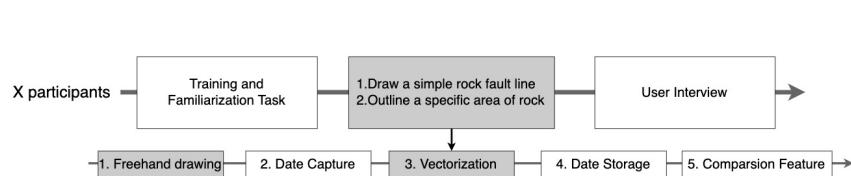


Fig. 10. The illustration of the protocol

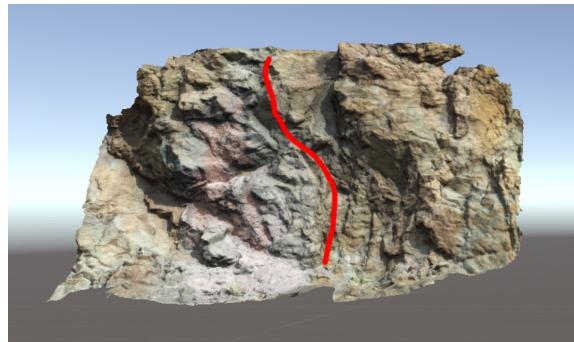


Fig. 11. Draw a Simple Rock Fault Line

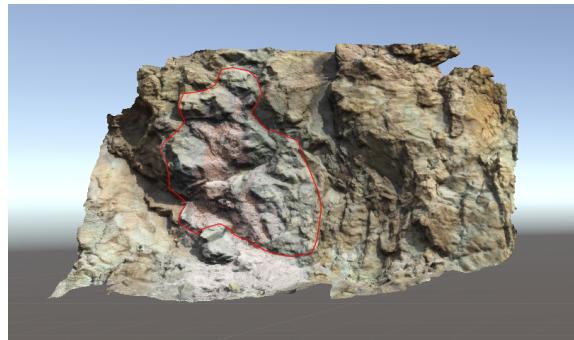


Fig. 12. Outline a Specific Area of Rock

task's objective. The surface of the rock model designated for drawing is relatively flat, and the entire fault line is visible within the user's field of view, making the task straightforward. Participants are tasked with replicating this fault line as efficiently as possible on an identical rock model positioned adjacent to the original, without the need for adjusting their viewpoint significantly. They can make necessary adjustments during the drawing process, the task is considered complete once the participant is satisfied with their replication. This task allows participants to quickly and easily complete the drawing, reflecting the simplicity and directness of the task.

#### **677 4.5 TASK2: Outline a Specific Area of Rock**

678 In this task, participants will observe a rock model where a specific section of a geological layer is outlined with a clear  
 679 closed line<sup>12</sup>, originally marked by engineers in a computer model. Their objective is to replicate this enclosed line on a  
 680 similar rock model located next to the original. The outlined area is relatively large, requiring participants to adjust  
 681 their viewpoint or adjust their viewpoint to complete the drawing accurately. This task tests the participants' ability  
 682 to draw and delineate an enclosed area on a 3D rock surface, emphasizing accuracy and the ability to manipulate the  
 683 model for better visibility.  
 684

#### **685 4.6 Vectorization of Drawn Lines**

686 Upon completing each line drawing, users can vectorize the lines through the UI panel. Selecting this option initiates  
 687 the system's automatic conversion of the originally sketched line into a vectorized curve. This vectorized curve is  
 688 smooth, aesthetically pleasing, and editable. In vectorized mode, users can select and modify points along the line to  
 689 alter its shape, achieving their desired outcome.  
 690

#### **691 4.7 Questionnaire and the Semi-structured User Interview**

692 After completing the tasks, participants fill out a questionnaire. This questionnaire collects immediate feedback on  
 693 their experience, focusing perceived responsiveness, editable satisfaction, and scalability satisfaction. Following the  
 694 completion of the questionnaire, participants undergo a semi-structured interview<sup>[5]</sup>. This interview allows for a  
 695 more in-depth exploration of their experiences, opinions, and suggestions for improvement, building on the responses  
 696 given in the questionnaires. These interviews provided insightful input for upcoming improvements by attempting to  
 697 comprehend the complex viewpoints and prospective enhancements from the user's point of view. The interview was  
 698 organized using a series of pre-planned open-ended questions that allowed participants to share their opinions and  
 699 experiences while also serving as a guide to ensure that all relevant subjects were covered. The interviews explored  
 700 several key areas:  
 701

- 702 • **Perceived Responsiveness:** "How would you rate the responsiveness of the AR system during the drawing  
 703 tasks? Did the system react quickly to your inputs? "
- 704 • **Editable Satisfaction:** "After vectorizing the lines, how satisfied were you with the ability to edit the vectorized  
 705 curves? Did the editing features meet your expectations for flexibility and precision?"
- 706 • **Scalability Satisfaction:** "After scaling the drawings, how would you assess the clarity and detail retention?  
 707 Were the details preserved to your satisfaction?"
- 708 • **Functionality and Tools:** "Which functions or tools within the system did you find most useful or lacking?  
 709 How did these impact your drawing process?"
- 710 • **System Usability:** "How did you find the usability of the AR system for drawing tasks? Were there any aspects  
 711 that felt particularly intuitive or challenging?"
- 712 • **Overall Experience:** "Can you describe your overall experience using the AR drawing system? What were the  
 713 highlights and lowlights?"
- 714 • **Suggestions for Improvement:** "What improvements would you suggest for enhancing the drawing experience  
 715 in this AR system?"
- 716 • **Future Usage:** "Based on your experience, how likely are you to use this type of technology in your regular  
 717 work? What changes would make it more appealing or useful for you?"

With the participant's permission, audio recordings of the around 10-15 minute interviews were made in order to ensure accuracy in the gathering and processing of data. The interviewer followed the guide but also took the opportunity to go deeper based on the participant's responses, ensuring a comprehensive understanding of their experience.

## 5 DATA ANALYSIS

The data analysis phase is crucial for interpreting the results gathered from the user study and drawing meaningful conclusions. This chapter outlines the procedures and techniques used to analyze the data collected from the questionnaires, tasks, and semi-structured interviews. The analysis is divided into quantitative and qualitative sections to provide a comprehensive understanding of user experiences and system performance.

### 5.1 Quantitative Analysis

Quantitative data were collected from questionnaires and task performance metrics. The following quantitative metrics were analyzed:

**Quantitative Metrics:** These metrics are directly measured from the data collected during the tasks and questionnaires:

- **Impact of Vectorization on Data Size:** Evaluates and compares the file sizes of the lines from freehand stage and vectorization stage. This metric helps understand the impact of vectorization on data efficiency and storage requirements.
- **Impact of Vectorization on Curve Smoothness:** This metric assessed the fluidity and visual quality of lines from the freehand stage and vectorization stage, indicating how natural and smooth the lines appeared on the 3D models. This metric can be quantitatively assessed using angle difference:

**Angle Difference:** To quantitatively analyze the smoothness, the angle changes between adjacent segments on two stages of the curve are examined. The coordinate of each vertex is recorded in a CSV file before and after vectorization. This involves computing the angle between tangent vectors at successive points, which can be determined using the dot product formula:

$$\cos(\theta) = \frac{T_1 \cdot T_2}{|T_1||T_2|}$$

where  $T_1$  and  $T_2$  are the tangent vectors at consecutive points along the curve,  $\cdot$  denotes the dot product, and  $|T_1|$  and  $|T_2|$  are the magnitudes of these vectors. This calculation provides the cosine of the angle  $\theta$ , indicating the directional change of the curve at each point.

- **Impact of Vectorization on Curve Accuracy:** This metric evaluates the similarity between lines from the freehand stage and the vectorization stage by measuring the deviations of the vectorized lines from the freehand lines. This analysis helps identify which stage retains more detail and precision in drawings on 3D models. To measure accuracy, the vectorized curves can be superimposed on the freehand curves, and deviations can be quantified by calculating the Euclidean distance between corresponding points on the freehand and vectorized curves.
- **Perceived Responsiveness:** During drawing tasks, participants rate the perceived responsiveness of the AR system on a scale of 1 (very unsatisfactory) to 5 (extremely satisfactory). This input facilitates the assessment of the system's functionality in real-time applications.

- 781 • **Scalability and Detail Retention:** Following the scaling of drawings, participants rate the effectiveness on a  
 782 scale of 1 (very poor) to 5 (excellent), depending on how well details are retained. This measure assesses the  
 783 system's adaptability to changes in scale in terms of quality and detail.
- 784 • **Editable Satisfaction:** Participants rated their satisfaction with the ability to edit vectorized lines on a scale  
 785 of 1 (very unsatisfied) to 5 (very satisfied). This measure assessed how well the editing features met their  
 786 expectations for flexibility and precision.
- 787
- 788

## 789 5.2 Qualitative Analysis

790 Qualitative data were collected from semi-structured interviews conducted after the tasks. These interviews provided  
 791 deeper insights into user perceptions and experiences. The following qualitative metrics were analyzed:

792 **Qualitative Metrics:** These metrics are derived from participants' subjective evaluations, providing insights based  
 793 on user perception and experiences:

- 794 • **Total User Experience:** Participants highlight both the greatest and least rewarding parts of the system as they  
 795 discuss their overall experience with it. This measure pinpoints particular high and low times in the system's  
 796 use and assesses user satisfaction.
- 797 • **System Usability and Functionality:** Users evaluate the AR system's general usability for drawing tasks and  
 798 talk about the usefulness of certain tools and functions. This includes comments on any easy-to-understand or  
 799 difficult-to-understand elements they come across that affect their workflow and drawing process.
- 800 • **Suggestions for System Improvement:** Participants offer direct feedback on how to improve the drawing  
 801 experience in the augmented reality environment, along with possible system changes.
- 802 • **Future Usage and Appeal:** Participants consider how likely they are to utilize this technology on a daily basis  
 803 at work and offer suggestions for improvements that could make the system more user-friendly and appealing.  
 804 This measure sheds light on the technology's flexibility and long-term viability in work environments.
- 805
- 806
- 807
- 808
- 809
- 810

## 811 5.3 Integration of Quantitative and Qualitative Data

812 The results from the quantitative and qualitative analyses were integrated to provide a holistic view of the user  
 813 experiences with the AR drawing system. This integration helped to validate findings and offered deeper insights into  
 814 user satisfaction and areas for improvement. By combining the strengths of both quantitative and qualitative data, this  
 815 analysis provides a robust foundation for understanding user interactions with the AR drawing system and informs  
 816 future development and enhancements.

## 817 6 RESULT

818 This section presents the outcomes of our experimental evaluation involving ten participants, encompassing a diverse  
 819 group of engineers and software developers. Their insights were crucial in assessing the usability and functionality of  
 820 our 3D Drawing tool. Using two specific tasks designed to test the tool's capabilities, we evaluated its precision, ease of  
 821 use, and the effectiveness of vectorization in a 3D annotation environment. Task 1 involved drawing a simple rock fault  
 822 line, while Task 2 required outlining a complex area on a rock surface. Both tasks aimed to test the tool's ability to  
 823 handle varying complexities in drawing tasks within a virtual environment.

- 824 (1) **Data Size Reduction through Vectorization:** The study assessed the impact of vectorization on data storage,  
 825 revealing a significant reduction in data size across tasks. For Task 1, data size decreased by an average of 86.8%.



Fig. 13. A participant using the 3D drawing tools

while Task 2 showed a reduction of approximately 85.9%. These reductions suggest that vectorization not only conserves storage space but also potentially enhances data management efficiency in real-time applications.

- (2) **Impact of Vectorization on Curve Accuracy Analysis:** Our analysis aimed to measure the deviation between curves before and after vectorization to demonstrate the process's ability to maintain precision and minimize distortion. We focused on two tasks, each involving different types of drawings within a 3D annotation environment. The data results are shown in Figure 1414.

#### Task1:

- **Average Distance:** The reported average distances, such as 0.03, 0.07, and 0.1, show good adherence to the original drawings, indicating minimal loss of detail. The overall consistency suggests effective vectorization.
- **Maximum Distance:** Maximum deviations like 0.21 highlight areas where vectorization may struggle with absolute fidelity, particularly in more complex drawing segments. However, these remain within a reasonable range.
- **Standard Deviation:** Low standard deviations across participants indicate a reliable vectorization process, with most data showing reasonable consistency.

#### Task2:

- **Average Distance:** Despite the complexity of the tasks, average distances like 0.19 and 0.13 suggest that the vectorization maintains good detail.
- **Maximum Distance:** Higher deviations, such as 0.44, indicate challenges in maintaining precision in detailed areas but are still within acceptable limits.
- **Standard Deviation:** The variability in standard deviations shows consistency in the vectorization process, although some improvements could be beneficial.

Feedback from participants using the system supported the quantitative data. Users noted that the post-vectorization curves generally resembled the original inputs closely, which is consistent with the measured data showing that vectorization consistently maintains the overall integrity of the curve and important features.

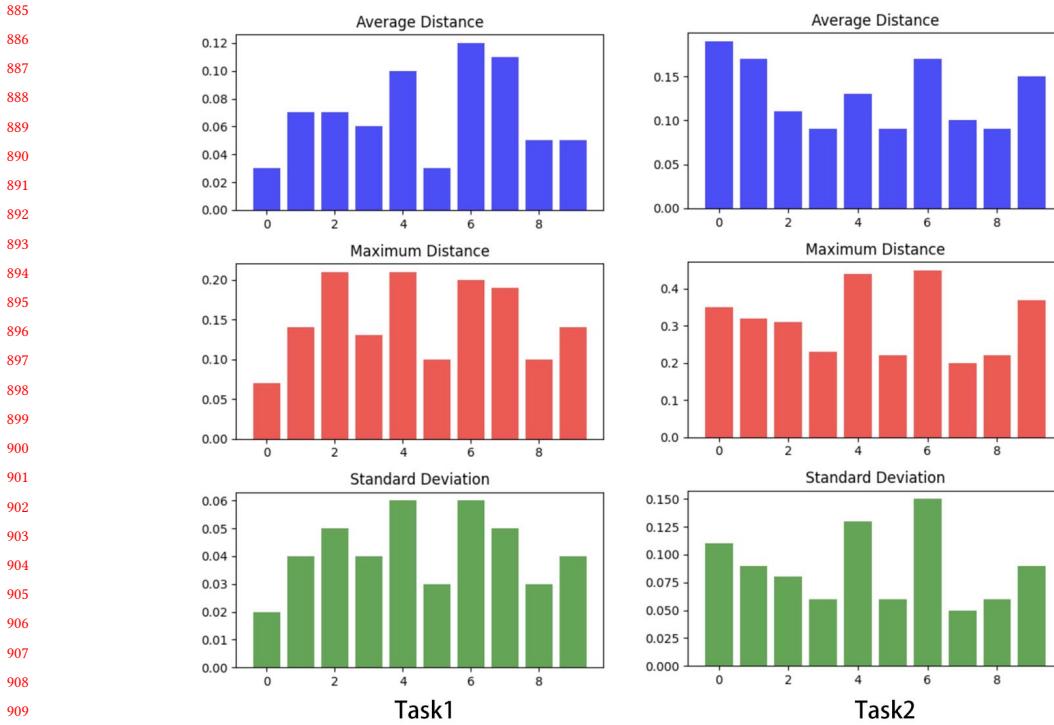


Fig. 14. Impact of Vectorization on Curve Precision Analysis

Considering the model's dimensions (2m x 1.16m x 0.45m), we have utilized the diagonal length, approximately 2.36 meters, as a metric to evaluate deviations. This approach offers an extensive overview on the spatial impact of these deviations within the model. An observed average deviation of about 0.1 meters represents approximately 4.24% of the diagonal length. This percentage provides a clear indication of the deviations' relative importance throughout the model. While a 4.24% deviation might be acceptable for general applications, it becomes significant for tasks that require high precision. This realization emphasizes the need for the vectorization technique to be further improved in order to minimize deviations, particularly in applications that call for a high degree of accuracy. Future improvements should focus on enhancing the tool's precision, particularly in managing complex shapes and detailed features.

### (3) Curve Smoothness Enhancement:

The evaluation of curve smoothness was substantiated by the analysis of average angles between vectors and the standard deviation of these angles, with the aim of illustrating the impact of vectorization on the smoothness of curve transitions. The data derived from both tasks are visualized in Figure 15<sup>12</sup>, showing notable differences in smoothness before and after vectorization.

#### Task1:

- **Average Angle Between Vectors:** The data indicates a clear reduction in the average angles between vectors post-vectorization, with figures showing a decrease across almost all data points. This suggests

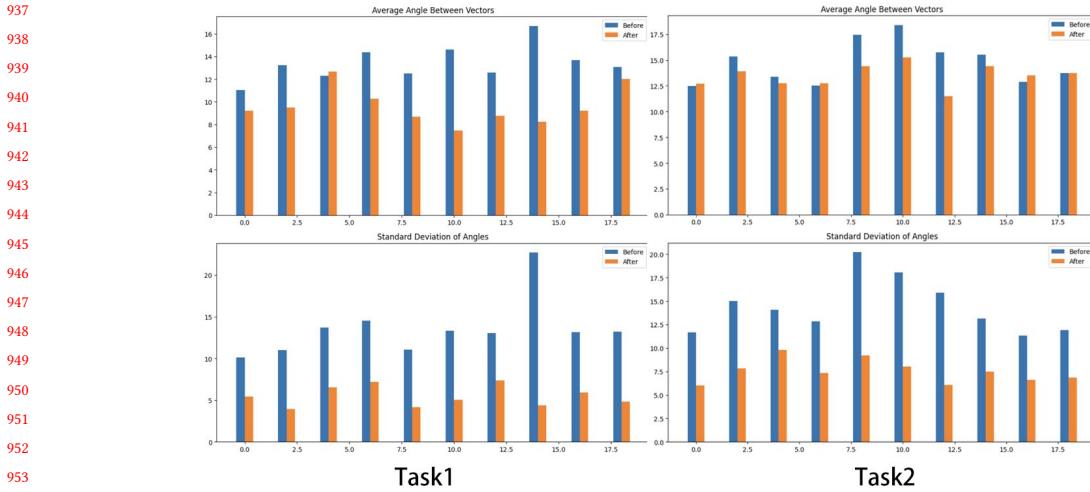


Fig. 15. Curve Smoothness Enhancement

that vectorization helps in aligning the curve segments more smoothly, resulting in a more cohesive and continuous line.

- **Standard Deviation of Angles:** There is a significant reduction in the variability of angles, as evidenced by the lower standard deviations after vectorization. This reduction highlights an enhanced consistency in curve smoothness, with fewer abrupt changes in direction, which is crucial for the precision required in fault line drawings.

#### Task2:

- **Average Angle Between Vectors:** The changes in average angles post-vectorization are less pronounced compared to Task 1, yet still present a general improvement. This indicates that while the vectorization process enhances smoothness, the complexity of the shapes in Task 2 poses challenges in achieving the same level of smoothness as seen in Task 1.
- **Standard Deviation of Angles:** Similar to Task 1, there is a decrease in the standard deviation of angles post-vectorization, although not as substantial. This improvement shows a better consistency in the way curves are formed post-vectorization, albeit with room for further enhancement to match the smoothness seen in simpler tasks.

The appended graphs clearly depict these changes, with the 'Before' bars generally higher than the 'After' bars, especially in the context of standard deviations, corroborating the textual data with visual evidence. The visible enhancements in curve smoothness following vectorization indicate how well the vectorization process works to improve the smoothness of drawn curves.

These results highlight the positive impact of vectorization on curve smoothness across different types of tasks. Task 1 demonstrates a more significant improvement due to its less complex nature, while Task 2, despite showing progress, suggests a need for tailored vectorization strategies to handle complex geometries more effectively. Future work will focus on optimizing vectorization techniques to enhance curve smoothness further, particularly for intricate and detailed designs.

- 989 (4) **Questionnaire Results:** Questionnaire responses rated the system highly in terms of responsiveness, editing  
990 features, and detail preservation, with average scores above 4 out of 5. Participants were asked to rate their  
991 experience using the tool on three aspects on a scale from 1 (lowest) to 5 (highest): responsiveness of the tool,  
992 quality of the editing features, and how well the details were preserved in their tasks.  
993

994 • **Responsiveness:** The tool showed excellent responsiveness, mainly receiving scores between 4 and 5.  
995 With most users reporting a responsive interface that easily satisfied their needs, the system's ability to  
996 enable rapid and efficient involvement is reflected in its high ranking.  
997

998 • **Editing Features:** Ratings for editing features were consistently high, ranging from 3 to 5, suggesting  
999 that the tool's functionality met or exceeded the expectations of most users. Although the essential  
1000 functionalities are powerful, these ratings indicate that there may be space for interface improvement and  
1001 the addition of features like enhanced pressure sensitivity and undo/redo capabilities.  
1002

1003 • **Details Preserved:** The detail preservation aspect received strong positive feedback, with average score  
1004 higher than 4, showing that the tool effectively maintained the essential details of the original designs.  
1005 The uniformly high ratings across all aspects reflect a successful user interface and functionality of the tool,  
1006 with particular strengths noted in responsiveness and detail preservation. The slightly lower score in editing  
1007 features from one participant might indicate areas for further improvement or additional feature integration.  
1008

1009 (5) **Semi-Structured Interview Insights:** The semi-structured interviews provided valuable qualitative data,  
1010 offering insights into the user experience and potential areas for enhancement in the AR drawing system.  
1011 Participants were asked about their overall experience, challenges faced, the intuitiveness of the system, and  
1012 suggestions for improvements.  
1013

1014 **Key Feedback and Suggestions:** Ease of Use and Intuitiveness: Participants noted that the system was  
1015 generally intuitive, with features like "key points to reshape" being highlighted as particularly useful. However,  
1016 some mentioned that aspects such as "the feel of the cursor on the screen" could be improved for a more natural  
1017 drawing experience.  
1018

1019 **System Improvements:** UI Enhancements: Users suggested improvements in the user interface, such as  
1020 "colors need to be more " and "improved UI elements like buttons to be easier to understand". This feedback  
1021 points to a need for more visual feedback and clearer interaction cues. Feature Enhancements: Several users  
1022 expressed the need for "better feedback on touch," indicating a desire for more responsive tactile feedback.  
1023 Suggestions included "very likely needs a redo/undo feature" and "more responsive to pressure," which would  
1024 allow for greater control over the drawing process. Functionality: Enhancements such as "add points of the  
1025 line to reshape" and "more ways to manipulate the drawing directly" were suggested to enhance the editing  
1026 capabilities and flexibility of the system.  
1027

## 7 DISCUSSION

This study underscores the effectiveness of the tool in enabling rapid, editable, and precise 3D drawing within a mixed reality environment, as demonstrated through tasks that assessed the tool's ability to draw and annotate complex geological features.

**Effectiveness of Vectorization:** Our findings indicate that vectorization significantly reduces data size, with reductions of 86.8% and 85.9% for Tasks 1 and 2, respectively. This substantial decrease in data size not only enhances storage efficiency but also proves crucial for the processing capabilities of lightweight mobile devices. Furthermore, the vectorization process maintains a high degree of accuracy in the curves, significantly improving their smoothness and

aesthetic appeal. It also increases the editability of the data, which has been highly appreciated by users, as it allows for more flexible and precise modifications.

**Challenges in Handling Complex Shapes:** Despite the noted successes, the result in Task 2 revealed some limitations when dealing with complex shapes. Compared to Task 1, the deviations from the original curves were greater in Task 2, even if there was still some increase in the smoothness of the curves. This increased deviation highlights the need for more advanced algorithms capable of effectively managing complex details and irregular shapes. Such capabilities are particularly critical in fields like geological surveying and architecture, where precision in representing complex forms can significantly impact the outcome of practical applications. The greater deviations observed in Task 2 underscore the challenges and emphasize the necessity for algorithmic improvements to ensure high fidelity in the vectorization of complex geometries.

**User Experience and System Usability:** Participant feedback highlighted the system's high responsiveness and editability, yet it also pointed out several areas needing improvement. Current editing features are good but limited; users expressed a desire for functionalities such as undo and redo options, an eraser tool, and the ability to add points between existing segments during the vector drawing stage. Additionally, issues were noted with button responsiveness, where touches sometimes result in delays. Enhancing these aspects would greatly improve user interaction, making the system not only more intuitive but also more accommodating to detailed user operations.

## 8 FUTURE WORK

In this project, our focus was primarily on the implementation of drawing and vectorization features. Consequently, some essential editing tools are currently not incorporated. Specifically, the addition of features such as "Undo" and "Eraser" would significantly enhance usability. However, implementing an "Undo" function requires tracking each action, which could lead to substantial performance challenges. An "Eraser" tool also presents complexities, particularly in managing data for interconnected strokes, necessitating sophisticated data handling strategies. Future enhancements should also aim to include advanced annotation capabilities and color-filling features to expand the tool's functionality.

Feedback from survey participants has highlighted areas needing improvement, such as pressure sensitivity and gesture recognition. Notably, difficulties were reported with the Microsoft Hololens 2 in detecting pinching gestures and left-hand usage. Since this technology has been evolving since 2018, these limitations might be addressed more effectively with newer devices, like the Apple Vision Pro, suggesting a direction for future hardware integration to better support our system's requirements.

## REFERENCES

- [1] Adobe. 2021. Medium by Adobe. <https://www.adobe.com/products/medium.html> Accessed: 2023-06-03.
- [2] Rahul Arora, Rubaiat Habib Kazi, Tovi Grossman, George Fitzmaurice, and Karan Singh. 2018. SymbiosisSketch: Combining 2D & 3D Sketching for Designing Detailed 3D Objects in Situ. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems* (Montreal QC, Canada) (*CHI '18*). Association for Computing Machinery, New York, NY, USA, 1–15. <https://doi.org/10.1145/3173574.3173759>
- [3] Rahul Arora, Rubaiat Habib Kazi, Fraser Anderson, Tovi Grossman, Karan Singh, and George Fitzmaurice. 2017. Experimental Evaluation of Sketching on Surfaces in VR. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems* (Denver, Colorado, USA) (*CHI '17*). Association for Computing Machinery, New York, NY, USA, 5643–5654. <https://doi.org/10.1145/3025453.3025474>
- [4] Rahul Arora and Karan Singh. 2021. Mid-Air Drawing of Curves on 3D Surfaces in Virtual Reality. *ACM Trans. Graph.* 40, 3, Article 33 (jul 2021), 17 pages. <https://doi.org/10.1145/3459090>
- [5] Jennifer Baumbusch. 2010. Semi-structured interviewing in practice-close research. *Journal for specialists in pediatric nursing* 15, 3 (2010), 255.
- [6] Mikhail Bessmeltsev and Justin Solomon. 2019. Vectorization of Line Drawings via Polyvector Fields. *ACM Trans. Graph.* 38, 1, Article 9 (jan 2019), 12 pages. <https://doi.org/10.1145/3202661>

- [7] Pengbo Bo, Gongning Luo, and Kuanquan Wang. 2015. A graph-based method for fitting planar B-spline curves with intersections. *Journal of Computational Design and Engineering* 3, 1 (06 2015), 14–23. <https://doi.org/10.1093/jcde.2015.05.001> arXiv:<https://academic.oup.com/jcde/article-pdf/3/1/14/33133648/jcde.2015.05.001.pdf>
- [8] Edwin Catmull and Raphael Rom. 1974. A CLASS OF LOCAL INTERPOLATING SPLINES. In *Computer Aided Geometric Design*, ROBERT E. BARNHILL and RICHARD F. RIESENFELD (Eds.). Academic Press, 317–326. <https://doi.org/10.1016/B978-0-12-079050-0.50020-5>
- [9] George Merrill Chaikin. 1974. An algorithm for high-speed curve generation. *Computer Graphics and Image Processing* 3, 4 (1974), 346–349. [https://doi.org/10.1016/0146-664X\(74\)90028-8](https://doi.org/10.1016/0146-664X(74)90028-8)
- [10] Elizabeth K Cridland, Sandra C Jones, Peter Caputi, and Christopher A Magee. 2015. Qualitative research with families living with autism spectrum disorder: Recommendations for conducting semistructured interviews. *Journal of Intellectual and Developmental Disability* 40, 1 (2015), 78–91.
- [11] Chris De Paoli and Karan Singh. 2015. SecondSkin: sketch-based construction of layered 3D models. *ACM Trans. Graph.* 34, 4, Article 126 (jul 2015), 10 pages. <https://doi.org/10.1145/2766948>
- [12] Michael F. Deering. 1995. HoloSketch: a virtual reality sketching/animation tool. *ACM Trans. Comput.-Hum. Interact.* 2, 3 (sep 1995), 220–238. <https://doi.org/10.1145/210079.210087>
- [13] Luca Donati, Simone Cesano, and Andrea Prati. 2017. An Accurate System for Fashion Hand-Drawn Sketches Vectorization. In *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*. 2280–2286. <https://doi.org/10.1109/ICCVW.2017.268>
- [14] Luca Donati, Simone Cesano, and Andrea Prati. 2018. A complete hand-drawn sketch vectorization framework. *Multimedia Tools and Applications* 78 (2018), 19083 – 19113. <https://api.semanticscholar.org/CorpusID:3379308>
- [15] Suyang Dong, A. Behzadan, Chen Feng, and V. Kamat. 2013. Collaborative visualization of engineering processes using tabletop augmented reality. *Adv. Eng. Softw.* 55 (2013), 45–55. <https://doi.org/10.1016/j.advengsoft.2012.09.001>
- [16] Tobias Drey, Jan Gugenheimer, Julian Karlbauer, Maximilian Milo, and Enrico Rukzio. 2020. VRSketchIn: Exploring the Design Space of Pen and Tablet Interaction for 3D Sketching in Virtual Reality. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems* (Honolulu, HI, USA) (CHI ’20). Association for Computing Machinery, New York, NY, USA, 1–14. <https://doi.org/10.1145/3313831.3376628>
- [17] John J. Dudley, Hendrik Schuff, and Per Ola Kristensson. 2018. Bare-Handed 3D Drawing in Augmented Reality. In *Proceedings of the 2018 Designing Interactive Systems Conference* (Hong Kong, China) (DIS ’18). Association for Computing Machinery, New York, NY, USA, 241–252. <https://doi.org/10.1145/3196709.3196737>
- [18] Danilo Gasques, Janet G. Johnson, Tommy Sharkey, and Nadir Weibel. 2019. PintAR: Sketching Spatial Experiences in Augmented Reality. In *Companion Publication of the 2019 on Designing Interactive Systems Conference 2019 Companion* (San Diego, CA, USA) (DIS ’19 Companion). Association for Computing Machinery, New York, NY, USA, 17–20. <https://doi.org/10.1145/3301019.3325158>
- [19] Danilo Gasques, Janet G. Johnson, Tommy Sharkey, and Nadir Weibel. 2019. What You Sketch Is What You Get: Quick and Easy Augmented Reality Prototyping with PintAR. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) (CHI EA ’19). Association for Computing Machinery, New York, NY, USA, 1–6. <https://doi.org/10.1145/3290607.3312847>
- [20] M. Gomarasca. 2009. Elements of Photogrammetry. (2009), 79–121. [https://doi.org/10.1007/978-1-4419-0914-1\\_3](https://doi.org/10.1007/978-1-4419-0914-1_3)
- [21] Google. 2020. Tilt Brush by Google. <https://www.tiltbrush.com/> Accessed: 2023-06-03.
- [22] Gravity Sketch. 2020. Gravity Sketch. <https://www.gravitysketch.com/> Accessed: 2023-06-03.
- [23] Johann Israel, Laurence Mauderli, and Laurent Greslin. 2013. Mastering Digital Materiality in Immersive Modelling. *Proceedings - Sketch-Based Interfaces and Modeling, SBIM 2013 - Part of Expressive 2013*, 15–22. <https://doi.org/10.1145/2487381.2487388>
- [24] J.H. Israel, E. Wiese, M. Mateescu, C. Zöllner, and R. Stark. 2009. Investigating three-dimensional sketching for early conceptual design—Results from expert discussions and user studies. *Computers* 33, 4 (2009), 462–473. <https://doi.org/10.1016/j.cag.2009.05.005>
- [25] Bret Jackson and Daniel F. Keefe. 2016. Lift-Off: Using Reference Imagery and Freehand Sketching to Create 3D Models in VR. *IEEE Transactions on Visualization and Computer Graphics* 22, 4 (2016), 1442–1451. <https://doi.org/10.1109/TVCG.2016.2518099>
- [26] Sho Kamuro, Kouta Minamizawa, and Susumu Tachi. 2011. 3D Haptic modeling system using ungrounded pen-shaped kinesthetic display. In *2011 IEEE Virtual Reality Conference*. 217–218. <https://doi.org/10.1109/VR.2011.5759476>
- [27] Daniel Keefe, Robert Zeleznik, and David Laidlaw. 2007. Drawing on Air: Input Techniques for Controlled 3D Line Illustration. *IEEE Transactions on Visualization and Computer Graphics* 13, 5 (2007), 1067–1081. <https://doi.org/10.1109/TVCG.2007.1060>
- [28] Steven Eric Krauss, Azimi Hamzah, Zoharah Omar, Turiman Suandi, Ismi Arif Ismail, Mohd Zaidan Zahari, and Zanariah Mohd Nor. 2009. Preliminary investigation and interview guide development for studying how Malaysian farmers’ form their mental models of farming. *The Qualitative Report* 14, 2 (2009), 245.
- [29] Vojtěch Krs, Ersin Yumer, Nathan Carr, Bedrich Benes, and Radomír Měch. 2017. Skippy: single view 3D curve interactive modeling. *ACM Trans. Graph.* 36, 4, Article 128 (jul 2017), 12 pages. <https://doi.org/10.1145/3072959.3073603>
- [30] Mayra D. Barrera Machuca, Paul Asente, Wolfgang Stuerzlinger, Jingwan Lu, and Byungmoon Kim. 2018. Multiplanes: Assisted Freehand VR Sketching. In *Proceedings of the 2018 ACM Symposium on Spatial User Interaction* (Berlin, Germany) (SUI ’18). Association for Computing Machinery, New York, NY, USA, 36–47. <https://doi.org/10.1145/3267782.3267786>
- [31] Microsoft. [n. d.]. Microsoft HoloLens. <https://www.microsoft.com/en-us/hololens>. Accessed: 2024-06-04.
- [32] Microsoft. n.d.. Mixed Reality Toolkit - Unity. <https://github.com/MixedRealityToolkit/MixedRealityToolkit-Unity>. Accessed: July 15, 2024.
- [33] Hirokazu Muraki, Shigenori Tanaka, H. Furuta, Etsushi Kitagawa, and Hajime Noda. 2001. A Study for Producing 3D Model Space from 2D Digital Picture using Photogrammetry Technology. *Journal of Civil Engineering Information Processing System* 10 (2001), 277–286. <https://doi.org/10.11532/>

- 1145 JOURNALAC1992.10.277  
1146 [34] Patryk Najgebauer and Rafal Scherer. 2019. Inertia-based Fast Vectorization of Line Drawings. *Computer Graphics Forum* (2019). <https://doi.org/10.1111/cgf.13829>  
1147 [35] Gioacchino Noris, Alexander Hornung, Robert W. Sumner, Maryann Simmons, and Markus Gross. 2013. Topology-driven vectorization of clean line drawings. *ACM Trans. Graph.* 32, 1, Article 4 (feb 2013), 11 pages. <https://doi.org/10.1145/2421636.2421640>  
1148 [36] Sanford Pun. 2023. VTracer Documentation. <https://www.visioncortex.org/vtracer-docs>  
1149 [37] Steven Schkolne, Michael Pruitt, and Peter Schröder. 2001. Surface drawing: creating organic 3D shapes with the hand and tangible tools. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (Seattle, Washington, USA) (*CHI '01*). Association for Computing Machinery, New York, NY, USA, 261–268. <https://doi.org/10.1145/365024.365114>  
1150 [38] Shahriyar Shahrabi. 2021. Mesh Texture Painting in Unity Using Shaders. <https://shahriyarshahrabi.medium.com/mesh-texture-painting-in-unity-using-shaders-8eb7fc31221c>  
1151 [39] Hemant Bhaskar Surale, Aakar Gupta, Mark Hancock, and Daniel Vogel. 2019. TabletInVR: Exploring the Design Space for Using a Multi-Touch Tablet in Virtual Reality. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems* (Glasgow, Scotland Uk) (*CHI '19*). Association for Computing Machinery, New York, NY, USA, 1–13. <https://doi.org/10.1145/3290605.3300243>  
1152 [40] Janice Tan, Lee Kannis-Dymand, and Christian Jones. 2022. Examining the potential of VR program Tilt Brush in reducing anxiety. *Virtual Reality* 27 (2022), 1–13. <https://doi.org/10.1007/s10055-022-00711-w> ©The Author(s) 2022.  
1153 [41] Chris Tsang. 2020. Impression Documentation. <https://www.visioncortex.org/impression-docs>. Accessed: 2024-06-18.  
1154 [42] Unity. 2024. Unity Game Engine. <https://unity.com/>. Accessed: 2024-06-11.  
1155 [43] Gerold Wesche and Hans-Peter Seidel. 2001. FreeDrawer: a free-form sketching system on the responsive workbench. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology* (Banff, Alberta, Canada) (*VRST '01*). Association for Computing Machinery, New York, NY, USA, 167–174. <https://doi.org/10.1145/505008.505041>  
1156 [44] Lisa S Whiting. 2008. Semi-structured interviews: guidance for novice researchers. *Nursing Standard (through 2013)* 22, 23 (2008), 35.  
1157 [45] Emilie Yu, Rahul Arora, Tibor Stanko, J. Andreas Bærentzen, Karan Singh, and Adrien Bousseau. 2021. CASSIE: Curve and Surface Sketching in Immersive Environments. In *Proceedings of the 2021 CHI Conference on Human Factors in Computing Systems* (Yokohama, Japan) (*CHI '21*). Association for Computing Machinery, New York, NY, USA, Article 190, 14 pages. <https://doi.org/10.1145/3411764.3445158>  
1158 [46] Xue Yu, Stephen DiVerdi, Akshay Sharma, and Yotam Gingold. 2021. ScaffoldSketch: Accurate Industrial Design Drawing in VR. In *The 34th Annual ACM Symposium on User Interface Software and Technology* (Virtual Event, USA) (*UIST '21*). Association for Computing Machinery, New York, NY, USA, 372–384. <https://doi.org/10.1145/3472749.3474756>  
1159 [47] Päivi Åstedt Kurki and Riitta-Liisa Heikkilä. 1994. Two approaches to the study of experiences of health and old age: the thematic interview and the narrative method. *Journal of Advanced Nursing* 20, 3 (1994), 418–421. <https://doi.org/10.1111/j.1365-2648.1994.tb02375.x>  
1160 arXiv:<https://onlinelibrary.wiley.com/doi/pdf/10.1111/j.1365-2648.1994.tb02375.x>  
1161  
1162  
1163  
1164  
1165  
1166  
1167  
1168  
1169  
1170  
1171  
1172  
1173  
1174  
1175  
1176  
1177  
1178  
1179  
1180  
1181  
1182  
1183  
1184  
1185  
1186  
1187  
1188  
1189  
1190  
1191  
1192  
1193  
1194  
1195  
1196