Evaluating Gesture-Based Augmented Reality Annotation

Yun Suk Chang,[*] Benjamin Nuernberger,[*] Bo Luan,[*] Tobias Höllerer[*]
University of California, Santa Barbara

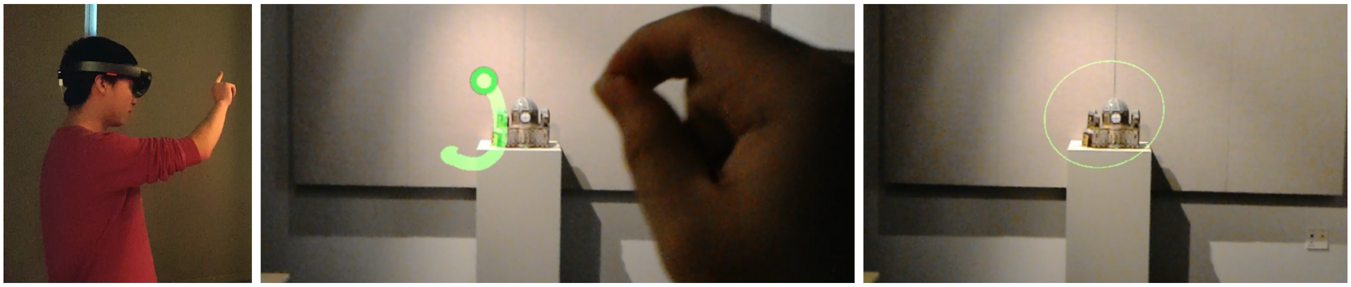


Figure 1: Overview of our annotation user study. Left: user wearing HoloLens, about to draw an annotation. Middle: “mixed-reality” capture view through the HoloLens while the user is drawing an annotation with the Air-Drawing method (user verging on the finger would actually see background as double image). Right: final beautified result of annotation.

Abstract

Drawing annotations with 3D hand gestures in augmented reality are useful for creating visual and spatial references in the real world, especially when these gestures can be issued from a distance. Different techniques exist for highlighting physical objects with hand-drawn circle and arrow annotations from a distance, assuming an approximate 3D scene model (*e.g.*, as provided by the Microsoft HoloLens). However, little is known about user preference and performance of such methods for annotating real-world 3D environments. In this paper, we compare different annotation methods using the HoloLens augmented reality development platform: Surface-Drawing and Air-Drawing, with either raw but smoothed or interpreted and beautified gesture input. For the Surface-Drawing method, users control a cursor that is projected onto the world model, allowing gesture input to occur directly on the surfaces of real-world objects. For the Air-Drawing method, gesture drawing occurs at the user’s fingertip and is projected onto the world model on release. The methods have different characteristics regarding necessitated vergence switches and afforded cursor control. We performed an experiment in which users draw on two different real-world objects at different distances using the different methods. Results indicate that Surface-Drawing is more accurate than Air-Drawing and Beautified annotations are drawn faster than Non-Beautified; participants also preferred Surface-Drawing and Beautified.

Keywords: Augmented Reality, Annotations, Spatial Referencing, HoloLens, User Study

Index Terms: H.5.1 [Information Interfaces and Presentation]: Multimedia Information Systems—Artificial, augmented, and virtual realities

1 Introduction

Annotating physical objects is a central 3D user interaction task in Augmented Reality (AR) applications [8]. Annotation at a distance is of particular importance, as a main advantage of AR is that users can browse and interact with objects in their field of view without having to move up to those objects. Current AR development platforms, such as the Microsoft HoloLens, facilitate drawing gesture interaction by providing stable tracking of head pose and approximate tracking of several finger poses in the user’s view. When issuing drawing gestures referring to real-world objects, the question arises as to how the drawing should be done: the paint can be dropped in mid-air, at the user’s fingertips, or it can be applied to surfaces in the real world (if those were previously modeled by or otherwise known to the system).

Several pros and cons are associated with each of these approaches. Drawing in mid-air at one’s fingertip is a very general technique that does not require a world surface model at drawing time. Once a gesture has been completed, it can be projected onto a possible scene model or onto strategically chosen (virtual) world planes. A drawback of this technique is the need for vergence switches when aiming at world objects (in Figure 1, middle, the user would see two out-of-focus versions of the pedestal and cathedral model when verging on the fingertip; this is the major reason why the annotation appears misaligned in the image). Drawing on world-surfaces has its drawbacks, too. For example, if annotation occurs at significant distances, small inaccuracies in fingertip tracking can have a large effect on paint placement in the physical world.

In this work, we conduct a user study, comparing several methods for issuing arrow and circle annotations of physical objects and landmarks. Arrow annotations are issued via a simple single paint stroke, with the direction of the stroke indicating the placement of an arrow-head (where the stroke ends). Circle annotations are circular or elliptical strokes that end up in the neighborhood of the stroke starting point. We also studied the effects of beautification of the final drawing. Results show that Surface-Drawing is more accurate than Air-Drawing and Beautified annotations have a slight speed advantage. In terms of subjective satisfaction, participants preferred Surface-Drawing over Air-Drawing and Beautified over Non-Beautified.

2 Related Work

Our work is related both to 2D drawing annotations for 3D AR and 3D free-hand drawings for both AR and VR.

2D drawing annotations. There has recently been much work in the area of using 2D drawing annotations for AR remote collaboration where a remote user draws 2D annotations that are sent over a network to a local user in AR [2, 5]. A major challenge with using such 2D annotations is the ambiguity of how to display them in

[*] e-mail: {ychang, bnuernberger, bo_luan, holl}@cs.ucsb.edu

2017 IEEE Symposium on 3D User Interfaces (3DUI)
March 18-19, 2017, Los Angeles, CA, USA
978-1-5090-6715-2/17/$31.00 © 2017 IEEE

3D AR. Nuernberger *et al.* [5] focused on circle and arrow gesture annotations and developed specific anchoring methods to interpret and render each in 3D AR. We follow their approach in focusing on circle and arrow gesture annotations; however, we utilize free-hand gesture drawings in 3D as input, whereas their approach is mainly concerned with 2D drawings as input. In addition, we further look into the effects of beautification of the resulting gesture annotation.

3D free-hand drawing annnotations. While much work has been done in using 3D hand gestures for a variety of tasks in VR and AR [1, 6], we focus specifically on those related to authoring AR content. Most prior work has concentrated on free-hand gestures for directly modeling 3D geometry for visualization [3], and recently there have been several consumer apps taking this approach (*e.g.*, Microsoft HoloLens Skype). In AR, however, the desire to annotate physical objects is an important use case that does not immediately occur in VR. More recently, Miksik *et al.* [4] introduced the Semantic Paintbrush for labeling real-world scenes via ray-casting plus semantic segmentation. Our Surface-Drawing method can be considered an image-based pointing or ray-casting technique, similar to the ray-casting technique used for the Head Crusher method in Pierce *et al.* [7]. Our Air-Drawing method bears more similarities with typical 3D free-hand drawings [3].

3 Methods

We designed two different drawing methods on HoloLens: Surface-Drawing (SD) and Air-Drawing (AD). We also provided different rendering settings for the post-processing of the user's drawing after user finishes drawing the annotation: a beautified output (B) and a non-beautified output (NB).

The drawings are performed via pinch-and-drag gestures and are completed by releasing the pinch gestures. We provide a 3D cursor (shown in Figure 1) to show the user where the lines will be drawn. The cursor is placed at the fingertip for the AD method and at the detected surface for the SD method. The drawn lines have fixed width of 1 cm. To reduce noise in the gesture input, we sample the user's drawing positions at 30 Hz and the finished annotation's path points at 1 point per 1 mm. The finished and sampled drawings are then recognized as arrows or circles by the $1 gesture recognizer [10]. To simplify the drawing process for the users, we defined an arrow annotation gesture as a single-stroke straight line with the first point representing the arrow tail and last point representing the arrow head.

3.1 Surface-Drawing (SD) and Air-Drawing (AD)

Our methods create annotations in 3D space in two steps. First, SD draws directly on the detected real-world surface data, while AD draws directly at the user's fingertip. Second, the completed drawings by both methods are projected in an appropriate place depending on the annotation type.

For SD, we define the drawing position as the intersection between the detected surface mesh and a ray-cast from user's head through the fingertip position. Consequently, as the user is drawing annotations, the user can easily verge on the object of interest since the annotation is displayed at the detected surface. AD also uses the same algorithm to project its points to the surface when the drawing is completed so that the points can be processed by next step.

As a user completes the drawings, they are placed in the following manner, determined to be effective through pilot studies: For arrow annotations, their heads are anchored at the projected surface and their tails are projected so that the arrow annotation is orthogonal to user's view direction. For circle annotations, first the average depth of each point in the projected drawing path from user's view plane at the time of completing the gesture is calculated. Next the annotations are back-projected to the plane that is orthogonal to the viewing direction and displaced at the calculated average depth.

Note that SD and AD only differ during the drawing phase. Both methods produce the same output given the same hand gestures.

3.2 Beautified (B) and Non-Beautified (NB) Annotation

After the annotation is completed and projected to the appropriate place, it has the option of going through a process of "beautification." If the Beautified mode is on, arrow and circle annotations will be transformed. Arrow annotations are replaced with parametrized straightened standard arrows with corresponding orientation and position. Circle annotations are replaced with ellipses with the same major-axis and minor-axis lengths; axis lengths are estimated by using the smallest bounding box that can fit the circle annotation (see Figure 2a).

If the Non-Beautified mode is on, the annotations will not be transformed after projection, with the exception of single-line arrow annotations that always receive arrow heads. Otherwise, there will be no further processing of the user's annotation (see Figure 2b).

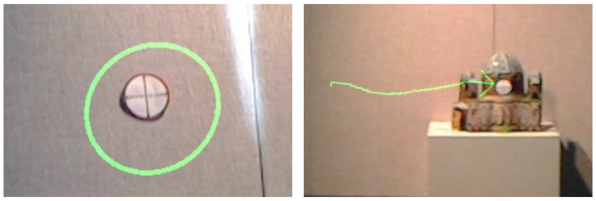

(a) Beautified Circle (b) Non-Beautified Arrow

Figure 2: Beautified and Non-Beautified annotations on objects at different distances.

4 Experiment Design

We designed our evaluation to investigate user preferences and performance on simple referencing tasks, using arrow and circle drawing gestures. Based on preliminary testing, we hypothesized that SD would be more accurate since users can verge their eyes on both the target object and the drawing simultaneously (at closer distances there still is the accommodation-vergence conflict to contend with), whereas they would have to verge at different distances for AD. On the other hand, verging solely on the drawing may allow AD to have an advantage over SD in terms of obtaining a "nice," aesthetically pleasing drawing. Based on this consideration, we included beautification mode as a condition in the experiment.

Participants used the Microsoft HoloLens and were placed in a room with controlled illumination. Because only the hand center position is provided by the HoloLens API, we estimated a generic displacement vector to allow users to draw at the fingertip[1].

4.1 Conditions

We used a 2x2 within-subjects design, where the blocked conditions were method (SD or AD) and beautification (B or NB). Two types of target objects were used: a planar crosshair target at about 5 feet away and a 3D building model at about 15 feet away, also with crosshairs (see Figure 2). Participants could draw two types of gestures: arrows and circles. Objects and gestures were balanced via randomized ordering.

4.2 Procedure

Participants completed a pre-study questionnaire to gather demographics information, followed by HoloLens calibration. After a short training phase, each participants completed the tasks for each condition twice, which were composed of drawing arrows and circles at the two target objects. Participants were told to draw the gestures as quickly and well as possible after a beep sound, which starts the timer; for arrows, they were told to try to hit the crosshair

[1] We did implement a manual fingertip calibration method, but we found that for most users, generic estimated fingertip positions, one for right- and left-handed users each, sufficed and helped to streamline the experiments.

and for circles, to encompass the object. They were also allowed to repeat each task as much as they liked. Questionnaires throughout the study gathered qualitative user responses (each condition used a generic name to avoid possible bias effects in phrasing). The entire procedure took just under one hour, and participants were compensated for their time with $10 USD.

5 Results

There were 16 participants total (ages 18 to 35, average 20.88); 5 male and 11 female. 4 said they had never used drawing tools/software; 10 said almost never; and 2 said several days a week. All participants had never used the HoloLens before (12 had never heard of it). 5 were left-eye dominant, while 11 right-eye dominant as determined by an eye-dominance test. Interpupillary distance (IPD) measurements were taken using the HoloLens calibration app; the average IPD was 62.52mm (stddev. 2.832). All but one participant used the right hand for drawing.

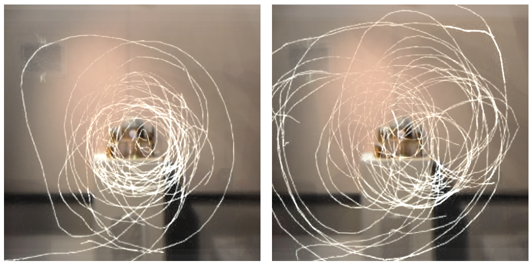
5.1 Task Results

Timings. We measured two different timings per each tasks. "Gesture time" measures time from the start gesture (finger-pinch) to the end gesture (finger-release). "Total time" measures time from the beep sound to the end gesture, to factor in time for aiming.

Participants took on average 5.37s total time (5.69s SD, 5.05s AD; 5.03s B, 5.70s NB) and 2.07s gesture time (2.21s SD, 1.93s AD; 1.94s B, 2.21s NB). To appropriately analyze the data, we first applied a natural logarithm transform to the timings since they were skewed towards zero. A two-way repeated measures ANOVA (factors 'method' and 'beautification') found no main effects on total time by factor method ($F_{1,15} = 2.59$, $p = 0.13$) and factor beautification ($F_{1,15} = 1.40$, $p = 0.26$). For gesture time, there was no significance for factor method ($F_{1,15} = 3.89$, $p = 0.067$), but there was a main effect for factor beautification ($F_{1,15} = 5.20$, $p < 0.05$). There were no interaction effects for either total time or gesture time. It is interesting that the gesture time for beautified drawings was faster since the actual drawing procedure did not differ.

Accuracy. Arrow drawing accuracy was measured as the Euclidean distance between the final arrow head position and the crosshair target's 3D position (determined via a static model of the environment along with HoloLens' spatial anchors).

For circle annotations, we give a qualitative composite of all non-beautified circle drawings in Figure 3. As can be seen from the figure, SD circles tend to directly encompass the target object, whereas the AD circles exhibit left/right shifting with respect to the target object due to the aforementioned verging conflict between target and finger distance.



(a) Surface-Drawing condition (b) Air-Drawing condition

Figure 3: Composite of circle drawings on to the 3D target object.

Arrow drawings ended on average 0.16m away from the target (0.06m SD, 0.25m AD). Using the same data transformation approach, a two-way repeated measures ANOVA showed a statistically significant difference in accuracy between SD and AD ($F_{1,15} = 57.35$, $p < 0.001$). Beautification did not have a statistically significant effect on accuracy ($p = 0.13$).

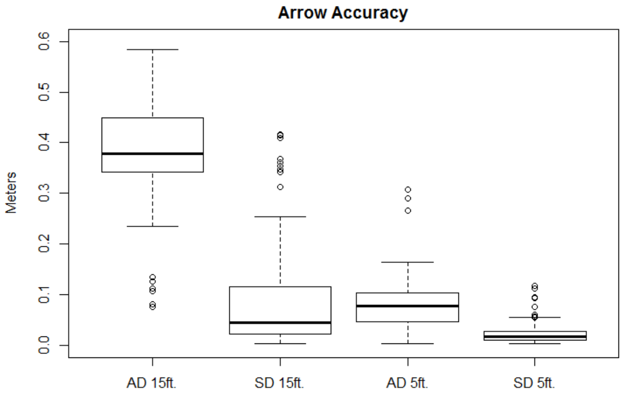


Figure 4: Box plot of arrow accuracy results based on different objects. The 3D building object was 15 feet away, whereas the planar object was 5 feet away.

Accuracy also decreased based on distance (the 3D building object was 15 feet away, whereas the planar object was 5 feet away); see Figure 4. A two-way repeated measures ANOVA (factors 'method' and 'object') indicated a statistically significant difference in accuracy between object types ($F_{1,15} = 156.6$, $p < 0.001$). There was also an interaction effect ($F_{1,15} = 9.10$, $p < 0.01$).

Repeats. We grouped repeats into a binary choice of did or did not repeat. For SD, users did not repeat 222 times and repeated 34 times. For AD, users did not repeat 181 times and repeated 75 times. Factorial logistic regression (factors 'method' and 'beautification') indicated a statistically significant difference for method ($p < 0.001$) and an interaction effect between method and beautification ($p = 0.01$). Without beautification, the amount users chose to repeat the task increased from 9 to 25 for SD but decreased from 39 to 36 for AD.

Training Effects. Users performed every condition twice. Using a two-way repeated measures ANOVA (factors 'method' and 'task number'), the second time around they were faster in terms of total time (5.71s then 5.03s; $F_{1,15} = 12.09$, $p < 0.005$); no effect was found in terms of pure gesture time (2.15s then 1.99s; $F_{1,15} = 4.15$, $p = 0.06$). There was no training effect on accuracy ($p = 0.69$).

Discussion. In general, SD is more accurate than AD, which fits with our hypothesis. However, we perceived a potential trend that SD may involve slower gesturing (method on gesture time, $F_{1,15} = 3.89$, $p= 0.067$), but follow-up studies are necessary to determine this. This potential trend may be due to the use of raycasting for precise annotating at distances. Although the algorithmic precisions for SD and AD are the same, we hypothesize that the user may experience more apparent precision for SD because small hand movements affect the drawing more visibly, encouraging users to draw more slowly in the hope for more accuracy. In addition, the beautified drawings had faster gesture time than the non-beautified. This may be due to users trusting that the system will improve the annotation eventually, leading to a more rapid drawing process.

5.2 Questionnaire Results

Q1: Individual method usability. After each condition, we asked participants, using a Likert-style questionnaire, how much they agreed or disagreed that the annotations were easy to draw, easy to aim, easy to hit the crosshair target (or circle the overall object), that the drawing input was aesthetically pleasing, and that the drawing output (result) was aesthetically pleasing (see Figure 5); we asked these questions separately for arrows and circles. In general, participants agreed more with those statements for the SD method than for the AD method. Many commented that the AD method would shift their drawing to the right or the left, thus making it less accurate than the SD method (this effect is due to the vergence mismatch

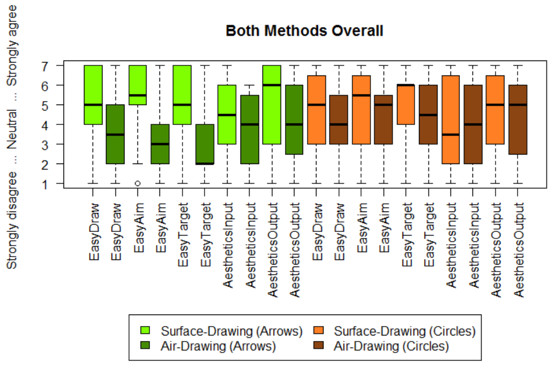


Figure 5: Box plot results for questionnaire Q1 for individual method usability (7 = strongly agree; 4 = neutral; 1 = strongly disagree).

Table 1: ART ANOVA results for questionnaire Q1. In all main effects, SD > AD and B > NB. Significance levels are: *** = 0.001, ** = 0.01, * = 0.05, . = 0.1. There were no interaction effects.

Question	Method (SD vs. AD)				Beautification (B vs. NB)			
	Arrow		Circle		Arrow		Circle	
	$F_{1,45}$	p	$F_{1,45}$	p	$F_{1,45}$	p	$F_{1,45}$	p
EasyDraw	18.24	***	2.65	0.11	8.00	**	6.76	*
EasyAim	31.23	***	6.36	*	3.29	.	7.00	*
EasyTarget	28.03	***	5.39	*	0.55	0.46	6.02	*
Aes. Input	5.85	*	0.10	0.76	32.24	***	28.15	***
Aes. Output	3.74	.	0.0	0.95	32.89	***	55.93	***

problem discussed above). Results from Aligned Rank Transform two-way repeated measures ANOVA tests [9] are shown in Table 1.

Q2: Method comparison. A second questionnaire asked participants to compare between SD and AD (Figure 6). One-sample median tests confirm that overall for all five Likert-scale statements, SD is preferred with statistical significance (all $p < 0.001$, except $p < 0.01$ for AestheticsInput). For beautified drawings, SD is still preferred with statistical significance for all five statements, but for non-beautified drawings, the final two statements on aesthetics did not yield a preferred method with statistical significance (p = 0.10, 0.11).

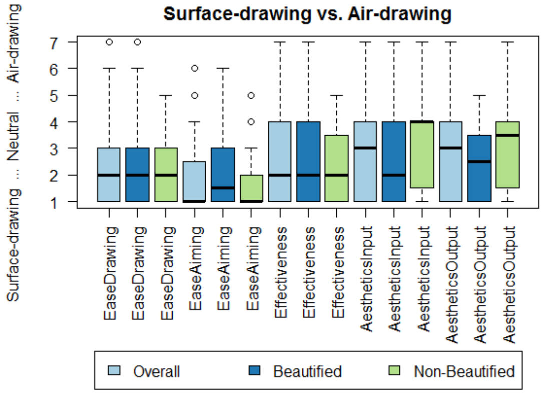


Figure 6: Box plot results for questionnaire Q2 (method comparison).

Q3: Overall preference. Finally, overall preference for the methods is shown in Figure 7. The entries labeled All_X refer to answers that state that all methods X were equally preferred. As can be seen from the results, participants chose SD with Beautification as their preferred method more than all other options. SD fared well in general and Beautification was generally preferred also.

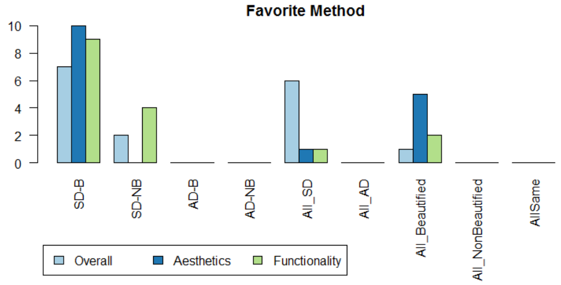


Figure 7: Histogram of responses for questionnaire Q3.

6 Conclusion

We presented an evaluation of two 3D drawing gesture annotation methods—Surface-Drawing and Air-Drawing—for spatial referencing of real-world objects in augmented reality. Surface-Drawing directly draws onto real-world surfaces, while Air-Drawing draws at the user's fingertip and is projected into the world upon release. Experimental results indicate that Surface-Drawing is more accurate than Air-Drawing and Beautified annotations are drawn faster than Non-Beautified; user participants also preferred Surface-Drawing over Air-Drawing and generally appreciated Beautification. Note that our findings generalize beyond HoloLens to any AR and VR devices that can detect hand gestures and have an environment model. Future work will investigate different gestures, target objects, and additional distances for drawing 3D annotations in AR. Future work will also explore ways to handle the vergence problem for the Air-Drawing method (*e.g.*, using and projecting from only the dominant eye for issuing annotations [7]). While drawing is a thoroughly explored concept for traditional user interfaces, 3D gesture drawing for AR annotation needs further exploration, and this work presents results in this direction.

Acknowledgements

This work was supported by NSF grant IIS-1423676 and ONR grants N00014-14-1-0133 and N00014-16-1-3002. We thank Donghao Ren and Anna Luo for help with the statistical analysis.

References

[1] D. A. Bowman, E. Kruijff, J. J. LaViola, and I. Poupyrev. *3D User Interfaces: Theory and Practice*. Addison Wesley Longman Publishing Co., Inc., Redwood City, CA, USA, 2004.

[2] O. Fakourfar, K. Ta, R. Tang, S. Bateman, and A. Tang. Stabilized annotations for mobile remote assistance. In *CHI*. ACM, 2016.

[3] D. Keefe, R. Zeleznik, and D. Laidlaw. Drawing on air: Input techniques for controlled 3d line illustration. *TVCG*, 13(5), 2007.

[4] O. Miksik, V. Vineet, M. Lidegaard, R. Prasaath, M. Nießner, S. Golodetz, S. L. Hicks, P. Perez, S. Izadi, and P. H. S. Torr. The semantic paintbrush: Interactive 3d mapping and recognition in large outdoor spaces. In *CHI*. ACM, 2015.

[5] B. Nuernberger, K.-C. Lien, T. Höllerer, and M. Turk. Interpreting 2d gesture annotations in 3d augmented reality. In *3DUI*. IEEE, 2016.

[6] O. Oda and S. Feiner. 3d referencing techniques for physical objects in shared augmented reality. In *ISMAR*. IEEE, 2012.

[7] J. S. Pierce, A. S. Forsberg, M. J. Conway, S. Hong, R. C. Zeleznik, and M. R. Mine. Image plane interaction techniques in 3d immersive environments. In *I3D*, New York, NY, USA, 1997. ACM.

[8] J. Wither, S. DiVerdi, and T. Höllerer. Annotation in outdoor augmented reality. *Computers & Graphics*, 33(6), 2009.

[9] J. O. Wobbrock, L. Findlater, D. Gergle, and J. J. Higgins. The aligned rank transform for nonparametric factorial analyses using only anova procedures. In *CHI*, New York, NY, USA, 2011. ACM.

[10] J. O. Wobbrock, A. D. Wilson, and Y. Li. Gestures without libraries, toolkits or training: A $1 recognizer for user interface prototypes. In *UIST*, New York, NY, USA, 2007. ACM.