

Dragon's Path: Synthesizing User-Centered Flying Creature Animation Paths for Outdoor Augmented Reality Experiences (Supplementary Material)

MINYOUNG KIM, George Mason University, USA
RAWAN ALGHOFAILI, University of Texas at Dallas, USA
CHANGYANG LI, George Mason University, USA
LAP-FAI YU, George Mason University, USA

CCS Concepts: • Computing methodologies → Graphics systems and interfaces.

ACM Reference Format:

Minyoung Kim, Rawan Alghofaili, Changyang Li, and Lap-Fai Yu. 2024. Dragon's Path: Synthesizing User-Centered Flying Creature Animation Paths for Outdoor Augmented Reality Experiences (Supplementary Material). In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Conference Papers '24 (SIGGRAPH Conference Papers '24), July 27-August 1, 2024, Denver, CO, USA*. ACM, New York, NY, USA, 12 pages. <https://doi.org/10.1145/3641519.3657397>

1 DATASET

To train our learning-based method, we created a dataset of location images annotated with potential action(s) compatible with the location. First, we rendered 500 RGB images of 100 candidate locations sampled from an input 3D scene. Figure 2 shows some examples from our dataset. These images contain the scene's texture showing its terrain (e.g., grass, buildings, water) and shading.

Then, we recruited 10 annotators using Amazon Mechanical Turk to label each location image with a set of compatible character actions from the action set A that could plausibly occur at that location. Figure 1 shows a screenshot of the annotation interface. Each location is associated with five images captured by a camera above the location and rotated about the location with a regular orientation interval. We allowed annotators to select multiple labels for each location image. In other words, each location image can be associated with more than one class of storyline action (e.g., an annotator can label a location image showing a lake with both the *drink* and *swim* actions). During data collection, each annotator annotated 100 locations with five images each. In other words, each annotator annotated 500 location images. A total of 5,000 images were annotated. After excluding images that were not given any character actions by annotators (i.e. images labeled with “None of the above”), our final dataset comprised 4,882 annotated images. For training, we split our dataset into 4,036 images for the training set and 846 images for the test set.

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

SIGGRAPH Conference Papers '24, July 27-August 1, 2024, Denver, CO, USA

© 2024 Copyright held by the owner/author(s).

ACM ISBN 979-8-4007-0525-0/24/07.

<https://doi.org/10.1145/3641519.3657397>



Fig. 1. Amazon Mechanical Turk example task. Annotators were asked to select action(s) that matched the red cross location in the scene.



Fig. 2. Example images of our dataset. Each row refers to the images of a location rendered from five different perspectives.

2 IMPLEMENTATION DETAILS

We conducted experiments with our learning-based location compatibility prediction method and the pathfinding approach implemented on a workstation equipped with Intel Core i9-11900F CPU, 64 GB of RAM, and NVIDIA GeForce RTX 3090 GPU.

2.1 Model Training

For training our model, we used images rendered by Unity out of the input 3D scene as our dataset. During the training process, the input RGB images with a 223×223 resolution are converted to 2048-dimensional latent vectors, which comprise the environmental features. This latent vector is used as the input of our fully-connected predictor and finally converted to the location compatibility value by a sigmoid function. In the fine-tuning step, we used 100 epochs with a batch size of 1. For optimization, we employed the Adam optimizer with 0.0005 learning rate and used Binary Cross Entropy Loss as the loss function. We evaluated our model using a separate test set and calculated the mean square error, which was 0.1810.

Our model was implemented using the PyTorch [Paszke et al. 2019] library.

2.2 Augmented Reality Deployment

Our approach on the server side was implemented based on the Unity 3D game engine and C# on the workstation. We trained our prediction model and ran the optimization on the server. In our user study, we used a laptop computer to load pre-sampled results. As a client, Microsoft HoloLens 2 visualizes the computation and rendering result from the server to the user in augmented reality. The Mixed Reality Toolkit 2 (MRTK2) framework handled deployment and communication between the server and client, tracking the position and rotation of the head-mounted device.

To consider a realistic surrounding area, we placed real-world 3D buildings and terrain geometry obtained from OpenMap API and Google Earth in the scene of Unity 3D. These 3D objects needed to be aligned with a real-world environment during the experiment. We accomplished the alignment by using manual transformation adjustment and by setting the virtual position of the user as the center of virtual space in Unity 3D. In experiments, the manual alignment took less than 5 minutes by adjusting the y-axis rotation and did not require specific expertise. This process could be automated by recent industry APIs (e.g., Google’s Geospatial Creator API) under Android or iOS mobile platforms; though it has not yet supported HoloLens AR app development. Note that, in our experiment setting, we assumed that the user was at a given position with a specific head orientation when she started the AR experience. After the approach started at the server side, the tracked pose driver component of Unity kept synchronized with the movement of the body and head of the user wearing the HoloLens 2.

3 EXPERIMENTS

3.1 Visibility Results

Figure 3 shows our virtual environment setting and result of visibility check process.

3.2 Unicorn Results in Hawaii

We demonstrate our approach using another flying character, a unicorn in a park near Magic Island Lagoon in Hawaii. Figure 4 shows the results. Like the dragon example, we use two stories, Story A (*Sleep-Sit-Drink*) and Story B (*Drink-Swim-Eat*), and three locations (Locations 1, 2, & 3). To generate the location compatibility maps, we sample the location images and use them as input to our trained action predictor. In general, despite a new environment, our predictor generates location compatibility values reasonably and associates the specific environmental features with storyline actions adequately. As can be seen in the top-right pictures, our approach chooses waypoints with a high location compatibility. For example, the unicorn chooses to drink in the near of the seashore (Figure 4(c)). The generated paths are also generally smooth and visible from the user’s viewpoint, as shown in (Figure 4(c) and (d)), even though half of the user’s viewspace was obscured by a tree. Figure 6 shows the location compatibility maps of the region around Magic Island Lagoon in Hawaii generated by the location compatibility predictors.

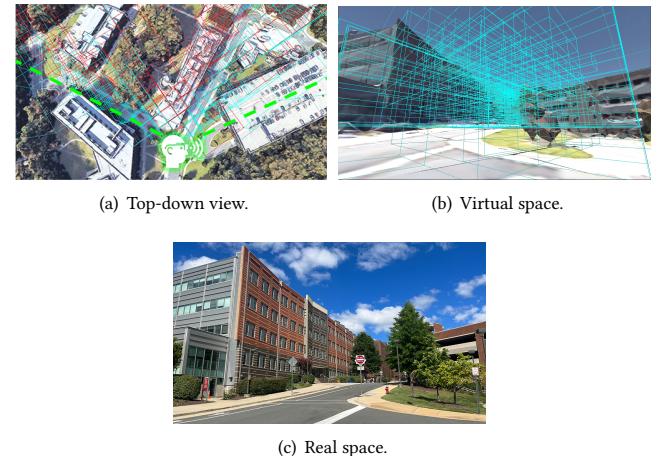


Fig. 3. Visibility check example. The top-down view (a) and the user’s view (b) of the scene are shown in the virtual space. The region fully visible to the user, including the air, is depicted by cyan voxels. The occluded region is depicted by red voxels. The real space (c) corresponding to the virtual space is also shown.

3.3 Unicorn Results in Sydney

Additionally, we produced results for a neighborhood around the Sydney ICC building as shown in Figure 5. Similar to the dragon example, we use two stories, Story A (*Eat-Sit-Drink*) and Story B (*Drink-Swim-Sleep*), and three locations (Locations 1, 2, & 3). We generated new location compatibility map for the new environment by using our trained action predictor (Figure 7). As can be seen in the top-right pictures, our approach chose waypoints with a high location compatibility. For example, the unicorn chose to drink in near the lakeshore (Figure 5(e)). The generated paths are also smooth in general and visible from the user’s viewpoints. Figure 7 shows the location compatibility maps of the urban area around ICC building in Sydney generated by the location compatibility predictors.

4 USER STUDY

We conducted a user study to evaluate our approach to deliver an outdoor AR experience based on the attribute on Table 1. We collected participants’ feedback on paths optimized by our approach and other baseline methods to investigate their efficacy in addressing the human-centered factors (e.g., visibility, engagement, effortlessness) we measured.

Participants. In this study, we recruited 25 participants. Eighteen identified as female and seven as male with ages ranging from 19 to 35 years old. 72 % of participants reported that they had no outdoor AR experience while 68 % reported that they experienced AR at least once. Participants gave written consent to participate in this IRB-approved study.

Designer Demographics Information. To obtain human-created designs for the locations in our user study, we recruited five designers who are identified as female and range in age from 26 to 34. All designers had experience with AR filters. Three are familiar with 3D game engines like Unity, but did not have any previous experience related to outdoor AR animation.

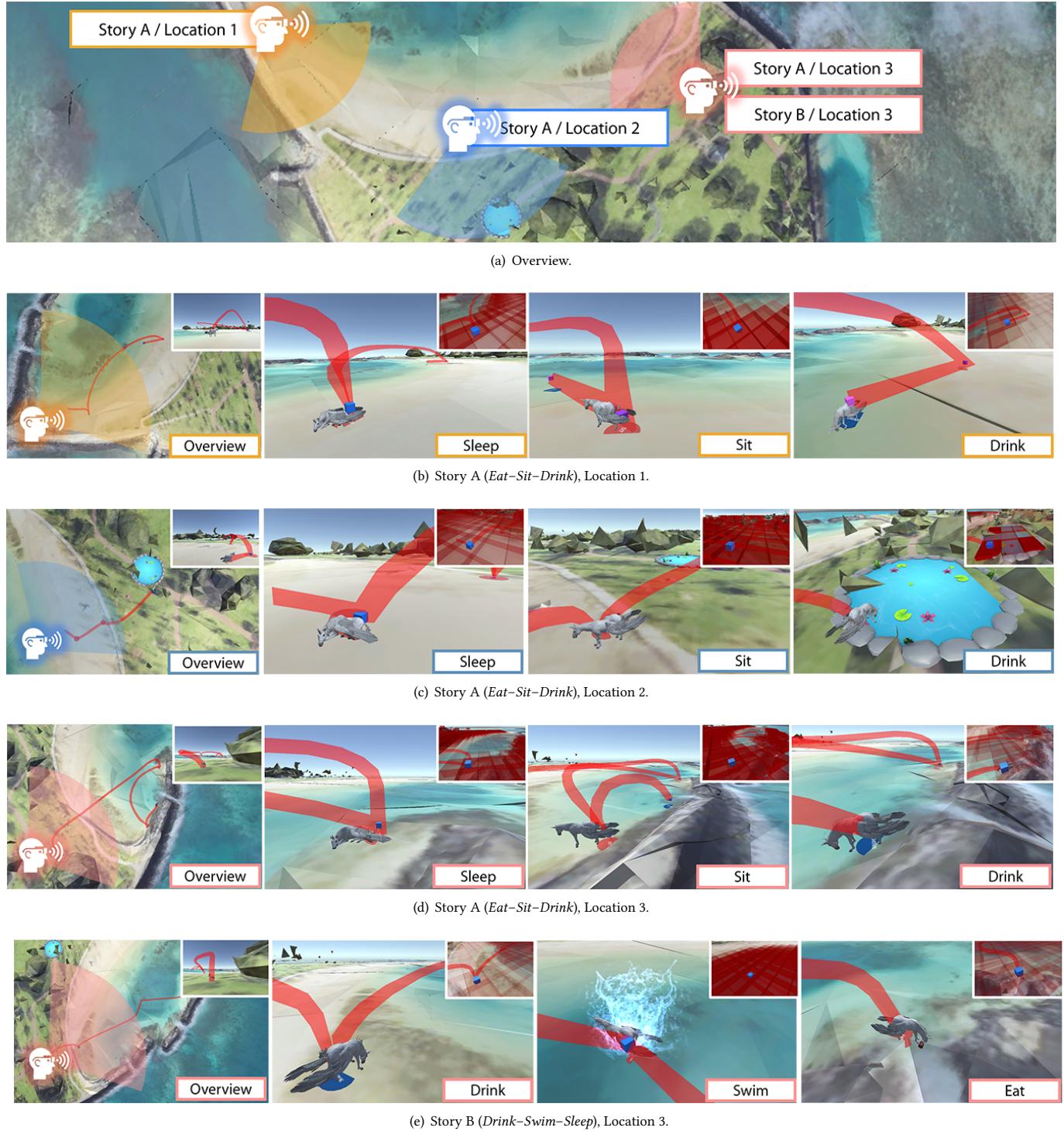


Fig. 4. Waypoints and paths synthesized for a unicorn around the Magic Island Lagoon in Hawaii with respect to different story-location combinations. The face icon represents the user's position. The circular sector denotes the user's view. For each storyline action, the image at the upper-right corner shows the location compatibility map, where a high redness refers to a high compatibility with the action shown. The blue dot on the map shows the flying creature's waypoint.



Fig. 5. Waypoints and paths synthesized for a unicorn around the Sydney ICC building with respect to different story-location combinations. The face icon represents the user's position. The circular sector denotes the user's view. For each storyline action, the image at the upper-right corner shows the location compatibility map. The blue dot on the map shows the flying creature's waypoint.

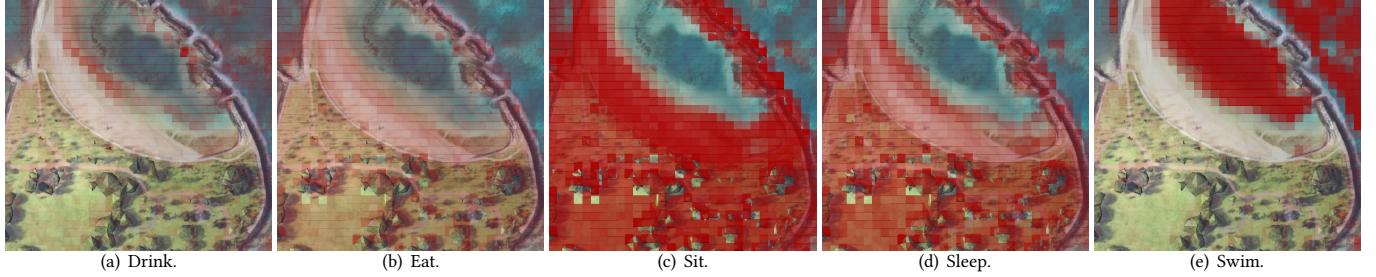


Fig. 6. Location compatibility maps of the region around the Magic Island Lagoon in Hawaii. Each map depicts the location compatibility values with respect to a storyline action class. The red intensity increases with the location compatibility value.

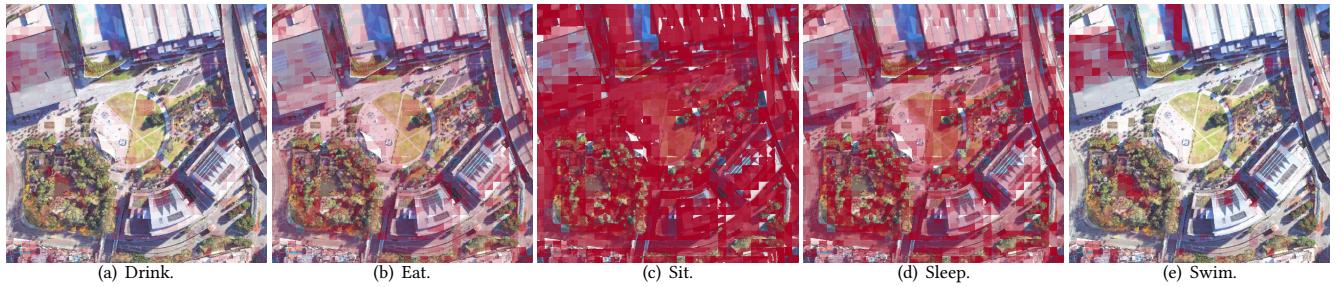


Fig. 7. Location compatibility maps of the region around the Sydney ICC building. Each map depicts the location compatibility values with respect to a storyline action class. The red intensity increases with the location compatibility value.

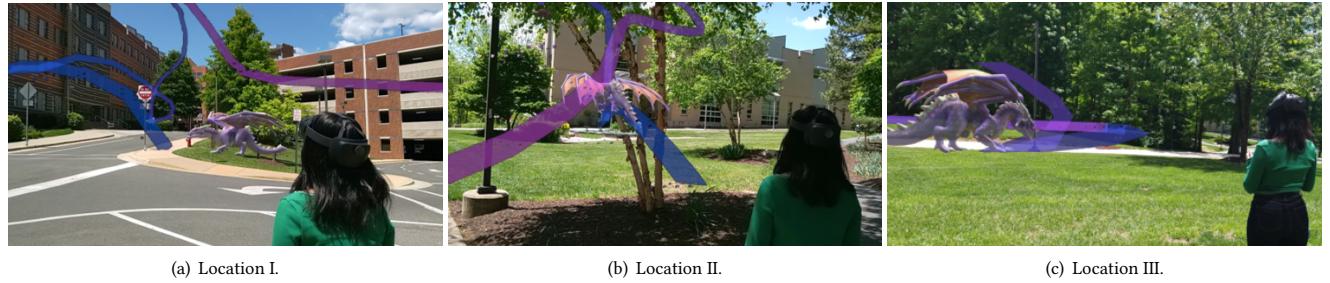


Fig. 8. Locations of the user study. During the experiment, the participants saw the augmented reality animations created with the four conditions at each location. Location I has storyline actions "Eat-Sit-Sleep". Location II has storyline actions "Drink-Eat-Sit". Location III has storyline actions "Sit-Swim-Sleep". The visualizations in this figure show the paths generated by the ML-Modified A* (Ours) condition.

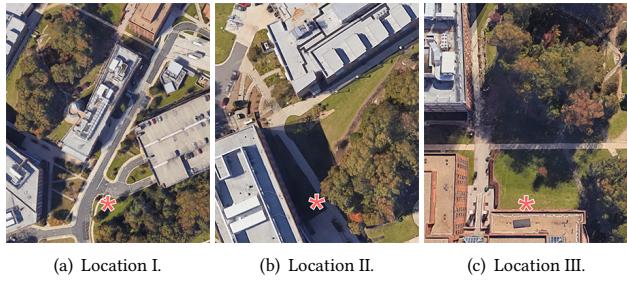


Fig. 9. Top-down View of the user study location. The asterisks indicate the user's location.

Experiment Design. We chose three different locations (Locations I, II, & III) and displayed a set of paths in AR to be evaluated by each participant. Figure 9 shows a top-down view of these locations. Each participant experienced and evaluated four path conditions (Random-Traditional A*, Designer-Traditional A*, ML-Traditional A*, and ML-Modified A*) in every location, with each location assigned a sequence of storyline actions for the path to be generated with. Figure 8 shows examples of paths generated by the ML-Modified A* (Ours) condition at the three user study locations. Note that we assume the user's position and orientation are static at each location to present a consistent path for all participants.

To evaluate our approach, we used four path conditions: Random-Traditional A* (random waypoints, with traditional A*), Designer-Traditional A* (waypoints selected by human designers, with traditional A*), ML-Traditional A* (waypoints selected by our prediction model, with traditional A*), and ML-Modified A* (waypoints selected by our prediction model, with our modified A*). The order of the locations and path conditions given to the participants were counterbalanced using Latin Square.

First, for the Random-Traditional A* condition, we randomly located three waypoints corresponding three storyline actions, only constraining the waypoint to be within five meters from the user. Once the sequence of waypoints were determined, the A* algorithm was applied to generate a single path to connect the waypoints. In this case, we used the traditional A* algorithm that considered the shortest path with a heuristic function. While similar in concept to Dreamwalker [Yang et al. 2019], which enables walking experiences using the user's environment, our result provides character action locations and a 3D aerial path based on the user's environment, visibility, and distance rather than redirect simulations.

For the Designer-Traditional A* condition, we used waypoints selected by designers. Before conducting the user study, for each of the Locations I, II and III, we collected five paths connecting waypoints, each created by one of our five designers. To facilitate the design process, we provided a top-down view of the animation site and an image showing the path rendered from the user's perspective. The design process is as follows. First, the designer selected waypoints for the three storyline actions in a given storyline action sequence. After that, the designer was shown the path generated by the A* pathfinding algorithm connecting the selected waypoints. The designer repeated this iterative process and modified the waypoints until satisfaction. During the user study, for each of the Locations I, II and III, a participant was shown one of the five designer-created paths connecting waypoints. As we had twenty-five participants in total, each of the five designer-created path was shown to and rated by five participants. Like the Random-Traditional A* path condition, the Designer-Traditional A* path condition generated final paths by using the traditional A* algorithm. Since traditional A* is the prevalent practice for pathfinding in game¹ and AR² app development, we provide that to designers for comparison.

Our optimization approach is able to select optimal waypoints considering the location compatibility of the given story actions, as well as human-centered factors such as distance and visibility. Our optimization also considers the cost of the generated path, while the ML-Traditional A* condition which uses traditional A* pathfinding only computes the shortest path between locations.

Lastly, the ML-Modified A*(Ours) condition used our optimization approach to select storyline action waypoints and produced a user-centered optimal path via our modified A* algorithm. In other words, the ML-Modified A* condition refers to results synthesized by our full approach considering location compatibility and user-centered interaction for outdoor AR experience. Because of the different spaciousness of experiment sites, we set $\lambda_l = 0.5$ as the weight of

¹Unity's article on its navigation system based on A*: <https://docs.unity3d.com/Packages/com.unity.ai.navigation@1.1/manual/AreasAndCosts.html>

²Developers' discussion on A* pathfinding in augmented reality: <https://forum.arongranberg.com/t/a-pathfinding-in-augmented-reality/12855/3>

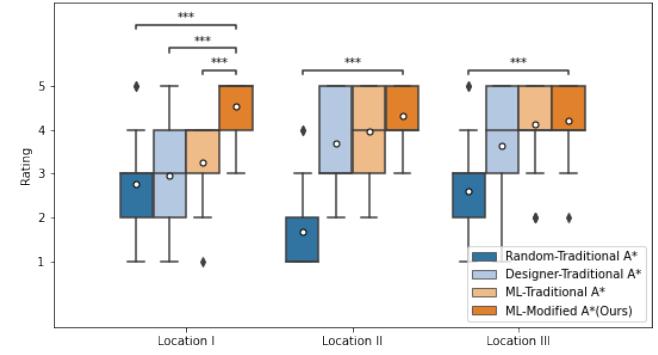


Fig. 10. Box plots of the participants' ratings on each condition at the three locations. At the end of the user study, we asked the participants to rate each condition based on their experience. ML-Modified A* was the highest-rated condition in all three locations. White dots show the means. Circles represent the outliers. Asterisks indicate significant differences.

location compatibility and $\lambda_{vw} = 0.5$ as the weight of visibility. For path term weights, we set $\lambda_{vp} = 0.7$, $\lambda_d = 0.2$, and $\lambda_s = 0.1$.

Procedure. We asked the user study participants to wear a HoloLens 2 device and instructed them to stand in a predetermined location and to look at a certain direction. To allow participants to judge the compatibility of the storyline action waypoints, we informed them about the story's plot before starting the animation. However, to avoid biases, we did not inform the participants about the condition we used to place waypoints and generate a path.

After watching the animations along the generated path for each scenario, participants were asked to complete the survey, which mainly inquired about the generated path, the placement of waypoints, and the efforts required to watch the animations. In the final user study, we collected a total of 300 responses from 25 users about the four path conditions in three locations. Table 1 shows a list of the questions we used to survey participants' experiences. We used a 5-point Likert scale, where a rating of 1 represents "strongly disagree" and a rating of 5 indicates "strongly agree". We also asked participants to assess the overall path shown to them by rating the general AR experience including their effort, enjoyment and impression from 1 to 5 indicating "poor" and "excellent" respectively.

For the attribute 'Effort-head steady', we provided four choices: no head rotation (with a rating of 5), slight head rotation within 45 degrees (with a rating of 4), moderate head rotation with more than 45 degrees and less than 90 degrees (with a rating of 3), much head rotation with more than 90 degrees and less than 180 degrees (with a rating of 2). There was no choice with a rating of 1. We designed the answers of this attribute this way to make it convenient to describe in tandem with the other attributes in the main paper.

4.1 Result

We analyzed participants' ratings of the 11 path attributes shown in Table 1, consisting of three categories: *Path*, *Location*, and *Effort*. Before presenting a detailed analysis of these attributes, we analyzed the overall rating of each path as shown in Figure 10.

Table 2 shows the results of the Wilcoxon signed-ranks test with Bonferroni correction for different attributes used in the survey of

Category	Attribute	Question
Path	realistic	The dragon's path looks realistic.
	dynamic	The animation path looks dynamic and entertaining.
	visible	I could see the entire path without moving the head from the initial position and rotation.
	unoccluded	The path was not occluded or passing through virtual objects e.g. virtual building, virtual tree.
	smooth	The dragon's path looks smooth with no sharp change or angle.
	proximal	The path is properly close for watching the animation from my position.
Location	compatible	The dragon's behavior looks compatible with the surrounding environment.
	reasonable	The selected locations are reasonable for the user to watch the dragon's action sequence.
Effort	intuitive	I was able to see the animation sequence intuitively without the instructor's instruction or advice.
	light	I spend little effort on seeing the entire path of the animation (e.g. searching for the dragon's position or rotating head).
	head steady	My head is steady and does not need to rotate much to see the entire animation path.

Table 1. The list of survey questions used in the user study experiment. We asked participants to answer these questions by rating from "strongly disagree (1)" to "strongly agree (5)" with a 5-point Likert scale.

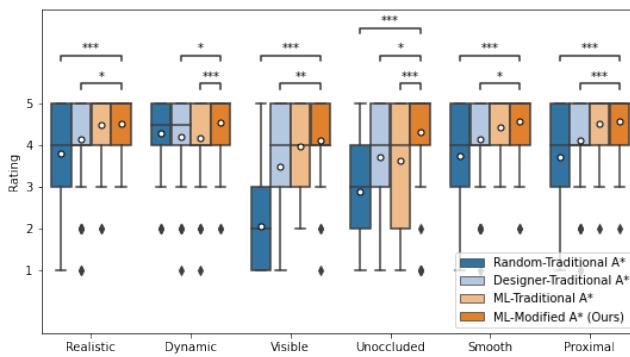


Fig. 11. Box plots of the participants' ratings on the path attributes. White dots show the means. Circles represent the outliers. Asterisks indicate significant differences.

the user study. Table 3 shows the results of the Wilcoxon signed-ranks test with Bonferroni correction for the participant's ratings on each condition at three locations.

The quality of these paths is highly dependent on the user's perspective from a specific location. To assess the differences among these approaches according to the location, we grouped the average ratings of each path condition by location (Figure 10). In all three locations ML-modified A* outperformed Random-Traditional A* ($p < 0.001$), but outperformed Designer-Traditional A* and ML-Traditional A* in only Location I ($p < 0.001$). We believe this is due to the differences in the shape of the visible open space between these locations. For example, Location I has a denser placement of large buildings inside the user's field of view, blocking the vertical air space for moving the creature in mid-air, unless directly in front of the user. Conversely, the other two locations have a vaster open space for placing the dragon in front of the user. Denser locations such as Location I force the designer to consider a more challenging user perspective with many more variables to weigh. Since the ML-Traditional A* utilizes a similar technique to Designer-Traditional A* for generating the path but with a predictive model trained on human-annotated data, it also fails in producing paths for denser areas like Location I.

4.1.1 Path Attribute Comparison. In this evaluation, we analyzed the effectiveness of our modified A* algorithm with respect to the *Path* attributes we examined: realism, dynamism, visibility, occlusion, smoothness, and proximity. Overall, the Friedman result showed a significant difference between ML-Modified A* and the other path conditions in all attributes: realistic ($\chi^2 = 43.6, p < 0.001, df = 3$), dynamic ($\chi^2 = 13.58, p < 0.05, df = 3$), visible ($\chi^2 = 110.96, p < 0.001, df = 3$), unoccluded ($\chi^2 = 45.64, p < 0.001, df = 3$), smooth ($\chi^2 = 44.26, p < 0.001, df = 3$) and proximal ($\chi^2 = 37.17, p < 0.001, df = 3$).

Figure 11 shows participants' ratings of the path attributes. To examine the differences among the four conditions, we conducted a post-hoc test using the Wilcoxon signed-rank test with Bonferroni correction. The test showed that the mean rating of the ML-Modified A* condition was significantly higher than the Random-Traditional A* and Designer-Traditional A*. Specifically, participants rated the paths of ML-Modified A* ($M = 4.52, SD = 0.68$) as more realistic than the paths of Random-Traditional A* ($M = 3.8, SD = 1.20$) ($p < 0.001$) and Designer-Traditional A* ($M = 4.15, SD = 1.056$) ($p < 0.05$). In addition, ML-Modified A* was rated as more dynamic than Designer-Traditional A* ($M = 4.21, SD = 0.97$) ($p < 0.05$) and ML-Traditional A* ($M = 4.18, SD = 0.90$) ($p < 0.001$).

Regarding visibility, participants rated ML-Modified A* ($M = 4.12, SD = 0.91$) paths as significantly more visible than Random-Traditional A* paths ($M = 2.07, SD = 1.17, p < 0.001$), Designer-Traditional A* paths ($M = 3.48, SD = 1.29, p < 0.01$), and significantly less occluded ($M = 4.33, SD = 1.13$) than all other path types (Random-Traditional A*: $M = 2.88, SD = 1.46, p < 0.001$, Designer-Traditional A*: $M = 3.7, SD = 1.47, p < 0.05$, and ML-Traditional A*: $M = 3.60, SD = 1.59, p < 0.001$).

In smoothness and proximity, ML-Modified A* (smoothness: $M = 4.57, SD = 0.72$) (proximity: $M = 4.59, SD = 0.70$) received higher-rates than Random-Traditional A* (smoothness: $M = 3.74, SD = 1.15, p < 0.001$) (proximity: $M = 3.71, SD = 1.30, p < 0.001$), Designer-Traditional A* (smoothness: $M = 4.15, SD = 1.06, p < 0.05$) (proximity: $M = 4.11, SD = 1.11, p < 0.001$).

Overall, our approach, ML-Modified A*, received favorable ratings on path realism, dynamism, visibility, non-occlusion, smoothness, and proximity for the outdoor AR experience. Our method produced

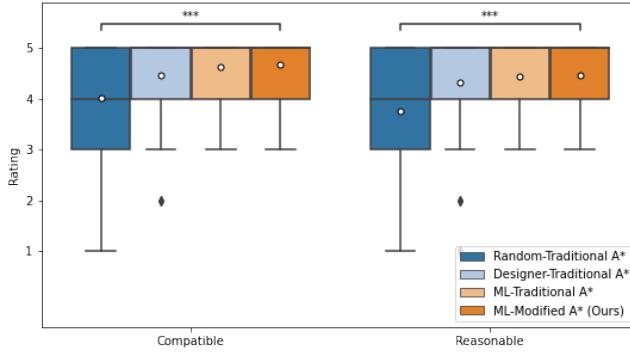


Fig. 12. Box plots of the participant's ratings on location attributes. White dots show the means. Circles represent the outliers. Asterisks indicate significant differences.

more desirable animation paths than the Random-Traditional A* and even the Designer-Traditional A* method in all of these attributes.

4.1.2 Location Compatibility Comparison. Figure 12 shows participants' average ratings of questions related to the *Location* category. With these two questions we investigated whether our approach can produce a set of waypoints compatible with and reasonable for the storyline actions. The Friedman test showed a statistical differences among the four path conditions with respect to location compatibility ($\chi^2 = 26.282, p < 0.001, df = 3$) and location reasonableness ($\chi^2 = 25.22, p < 0.001, df = 3$).

We observed that our location optimization approach, ML-Modified A* ($M = 4.67, SD = 0.55$), received significantly higher ratings than Random-Traditional A* ($M = 4.01, SD = 1.19$) in our post hoc analysis of location compatibility ($p < 0.001$). ML-Modified A*'s average rating ($M = 4.46, SD = 0.71$) was also significantly higher than Random-Traditional A* ($M = 3.75, SD = 1.35$) in terms of location reasonableness ($p < 0.001$), which indicates that our location prediction selects more reasonable and compatible locations for storyline actions than random placement.

Compared to Designer-Traditional A*, the difference between ML-Modified A* and Designer-Traditional A* was not as pronounced as comparing our approach to Random-Traditional A*. Our approach was slightly better than Designer-Traditional A* (compatibility: $M = 4.47, SD = 0.79$, reasonable: $M = 4.31, SD = 0.96$) when evaluating location compatibility and reasonableness. We believe this is because our model was trained with human-annotated data, which resulted in the model predicting similarly compatible results to human designers. In other words, our model was able to learn human intuition for predicting storyline action locations based on their features. Another possible reason is that location compatibility is a highly subjectively perceived attribute due to the individual's imagination and expectation. Furthermore, for basic character animations such as walking, the user accepts any type locations easily because of the generality of the animation which can be happen anywhere.

4.1.3 Effort Comparison. We conclude our analysis by evaluating participants' opinions on the amount of effort they expend in viewing the flying creature animations (Figure 13). The Friedman test

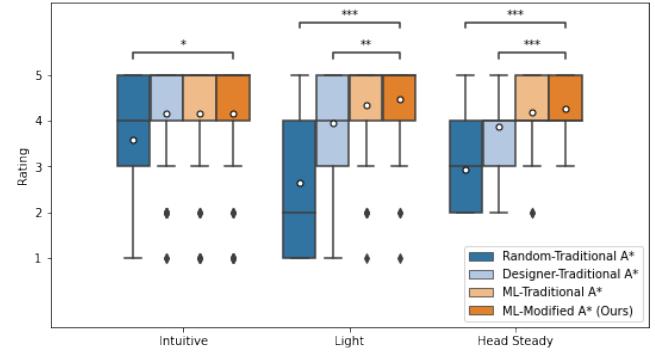


Fig. 13. Box plots of the participant's ratings on effort attributes. White dots show the means. Circles represent the outliers. Asterisks indicate significant differences.

showed a significant differences among the conditions in the *Effort* related attributes: intuitiveness ($\chi^2 = 25.09, p < 0.001, df = 3$), lightness ($\chi^2 = 82.21, p < 0.001, df = 3$), and effort in head rotation ($\chi^2 = 119.38, p < 0.001, df = 3$).

The ML-Modified A* condition (intuitive: $M = 4.15, SD = 1.39$) (light: $M = 3.46, SD = 0.85$) (head steady: $M = 4.27, SD = 0.60$) surpassed the Random-Traditional A* condition (intuitive: $M = 3.59, SD = 1.32, p < 0.05$) (light: $M = 2.63, SD = 1.40, p < 0.001$) (head steady: $M = 2.92, SD = 0.92, p < 0.001$) with respect to the *Effort* category.

Additionally, the ML-Modified A* condition was rated significantly higher than Designer-Traditional A* (light: $M = 3.95, SD = 1.1$) (head steady: $M = 3.88, SD = 0.753$) in the *Effort* attributes ($p < 0.001$), with the exception of intuitiveness (intuitive: $M = 4.14, SD = 1.182$ in Designer-Traditional A*). Meaning, our approach was able to generate paths that outperformed designers' in terms of the efforts in viewing the path with little head rotations. Furthermore, our approach matched designers' ability in producing an experience that needed little instruction or aid to view.

4.2 Per Location Analysis

We conducted a statistical analysis for each location (Figure 14, 15 and 16). In general, compared to the baseline conditions (e.g., Random-Traditional A* and designer-Traditional A*), our approach generated results that received comparable or higher user ratings (e.g., visibility, occlusion avoidance).

The ratings of some attributes (e.g., proximal, location compatibility) vary according to locations. For example, the proximal attribute is the most pronounced for location II, which has enough vertical air space. Based on that location's characteristics, the dragon tended to move vertically than horizontally. Therefore the result of user ratings for the proximal attribute suggests that users are more sensitive to the vertical distance between the dragon and themselves than the horizontal distance. In other words, a user may notice more of a dragon approaching if it flies down from a building to reach the user rather than coming laterally from the user's side.

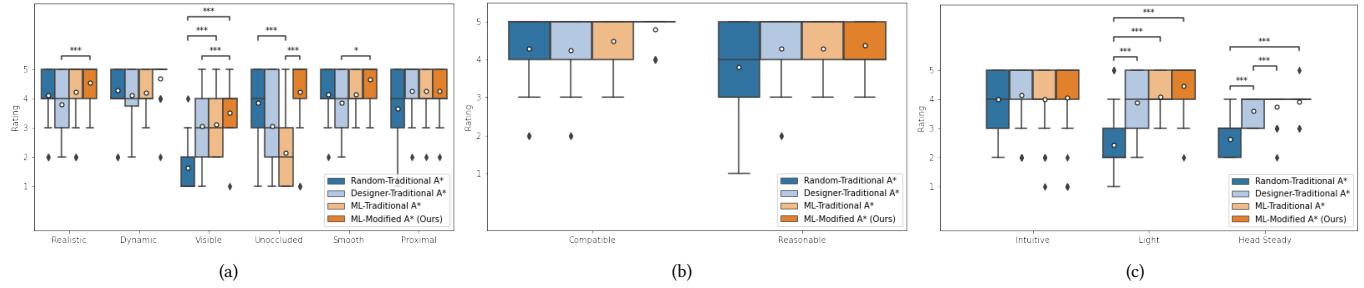


Fig. 14. (a) Box plots of the participant's ratings on path attributes for Location I. (b) Box plots of the participant's ratings on location attributes for Location I. (c) Box plots of the participant's ratings on effort attributes for Location I. Black dots represent the outliers and white circles represents the mean. Asterisks indicate significant differences.

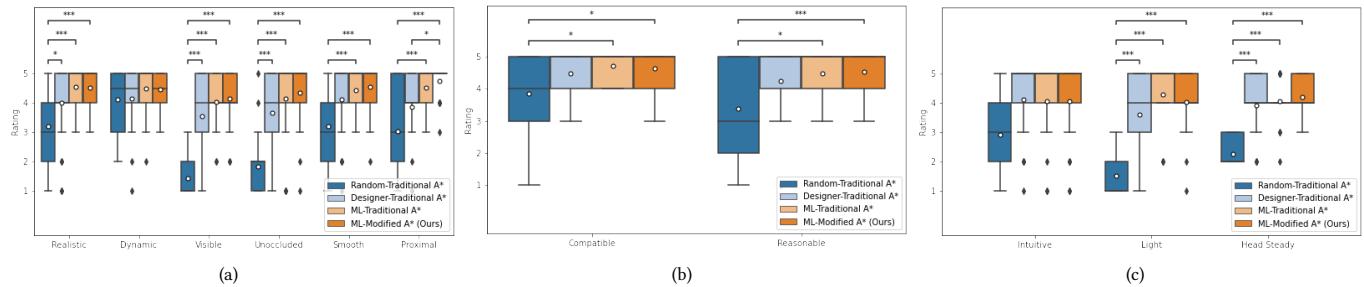


Fig. 15. (a) Box plots of the participant's ratings on path attributes for Location II. (b) Box plots of the participant's ratings on location attributes for Location II. (c) Box plots of the participant's ratings on effort attributes for Location II. Black dots represent the outliers and white circles represents the mean. Asterisks indicate significant differences.

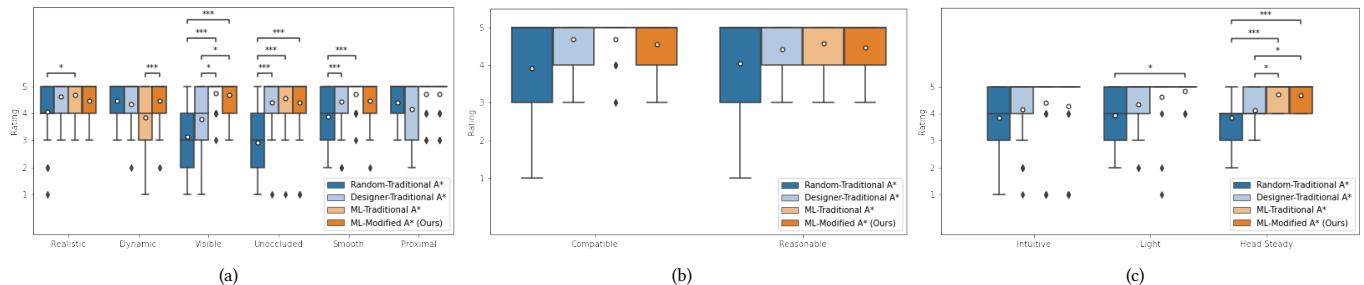


Fig. 16. (a) Box plots of the participant's ratings on path attributes for Location III. (b) Box plots of the participant's ratings on location attributes for Location III. (c) Box plots of the participant's ratings on effort attributes for Location III. Black dots represent the outliers and white circles represents the mean. Asterisks indicate significant differences.

4.3 Qualitative feedback

During the user study, we asked participants to freely comment on the most important aspect for the animation path they saw. Many participants mentioned *proximity* as the most important factor for viewing the animation.

Different participants may feel differently with regard to the distance between the character and them. Some may feel comfortable with the character being close, while some may feel the opposite, as reflected by the comments: "When the Dragon is getting close to me,

it is fun and cool to see." (*Participant 7*) and "Path A (e.g., Random-Traditional A*) was least pleasant because I had to rotate my head almost 90 degrees and the dragon was too close." (*Participant 20*).

Ensuring that the Dragon's path is smooth and collision-free is important for the realistic and natural outlook of the path, which participants mentioned frequently: "The most pleasant path to see did not collide with objects and was not stuck even though the path was complex." (*Participant 3*).

5 DISCUSSION

We further discuss the practical use of our approach. By using different combinations of storylines and viewpoints, we illustrate in Section 3 that our approach could be generalized across different narratives with different characters (e.g., a unicorn) and scenes (Sydney and Hawaii). Nevertheless, we discuss practical issues that could arise in applying different character animations in the following.

First, the designer will need to prepare additional animations to demonstrate other types of characters such as a humanoid. Since we focused on animal characters, the given animal action sets may not apply to characters of other shapes. Moreover, we focused on terrain-oriented interactions between the terrain geometry and the animated character which is an animal in our case. However, for humanoid characters, we anticipate that the character animations are usually associated with detailed objects (e.g., a bench) in the scene. Using appropriate humanoid character animations (e.g., sitting on a bench) would make the result look more realistic.

Second, incorporating additional actions requires fine-tuning the few-shot learning model with a dataset containing additional annotation labels. While our method is not limited to five animal behaviors, the scene compatibility of some specific actions rely on human intuition, so human-labeled annotations may be needed to associate such actions with an environment.

Third, a designer may need to train our predictor with additional training datasets to improve the adaptability of our model so that it can be applied to additional types of scenes (e.g., a desert scene, a metropolitan area).

REFERENCES

- Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, et al. 2019. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems* 32 (2019).
- Jackie Yang, Christian Holz, Eyal Ofek, and Andrew D Wilson. 2019. Dreamwalker: Substituting real-world walking experiences with a virtual reality. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*.

Attributes	Random-Designer		Random-ML Trad A*		Random-ML Modi A*		Designer-ML Trad A*		Designer-ML Modi A*		ML Modi A* - ML Trad A*	
	Z	P-value	Z	P-value	Z	P-value	Z	P-value	Z	P-value	Z	P-value
Path-realistic	-2.234	0.153	-4.433	<0.001	-4.832	<0.001	-3.096	0.012	-3.096	0.012	-0.339	4.406
Path-dynamic	-0.660	3.054	-0.715	2.849	-2.316	0.123	-0.327	4.463	-2.921	0.021	-3.539	0.002
Path-visible	-5.965	<0.001	-6.949	<0.001	-7.073	<0.001	-2.910	0.022	-3.381	0.004	-1.557	0.717
Path-unoccluded	-3.148	0.010	-2.502	0.012	-5.723	<0.001	-0.483	0.629	-3.165	0.002	-3.502	0.003
Path-smooth	-2.591	0.057	-4.268	<0.001	-5.177	<0.001	-2.456	0.084	-2.982	0.017	-1.215	1.345
Path-proximal	-2.199	0.028	-4.527	<0.001	-4.578	<0.001	-2.695	0.042	-3.296	0.006	-1.089	1.657
Location-compatible	-2.928	0.003	-4.016	<0.001	-4.525	<0.001	-1.848	0.065	-1.923	0.055	-0.565	0.572
Location-reasonable	-3.033	0.002	-3.866	<0.001	-4.499	<0.001	-1.122	0.262	-1.177	0.239	-0.163	0.870
Effort-intuitive	-2.947	0.003	-2.751	0.006	-2.660	0.008	-0.213	0.831	-0.141	0.888	-0.114	0.909
Effort-light	-5.225	<0.001	-5.915	<0.001	-6.273	<0.001	-2.325	0.020	-3.210	0.001	-1.006	0.315
Effort-head steady	-5.843	<0.001	-6.857	<0.001	-7.015	<0.001	-2.982	0.003	-3.868	<0.001	-1.279	0.201

Table 2. Wilcoxon signed-ranks test results for each attribute. All p-values are calculated with Bonferroni correction.

Num. Location	Random-Designer		Random-ML Trad A*		Random-ML Modi A*		Designer-ML Trad A*		Designer-ML Modi A*		ML Modi A* - ML Trad A*	
	Z	P-value	Z	P-value	Z	P-value	Z	P-value	Z	P-value	Z	P-value
Location I	-0.338	0.736	-1.472	0.141	-3.934	<0.001	-0.972	0.331	-3.564	<0.001	-4.095	<0.001
Location II	-3.827	<0.001	-4.416	<0.001	-4.409	<0.001	-0.884	0.377	-1.833	<0.001	-1.732	0.498
Location III	-2.587	0.06	-3.538	<0.001	-4.005	<0.001	-1.392	0.984	-1.466	0.858	-0.638	3.138

Table 3. Wilcoxon signed-ranks test results for the rating of each location. All p-values are calculated with Bonferroni correction.

Attribute	Random Traditional A*		Designer Traditional A*		ML Traditional A*		ML Modified A*	
	Mean	Std	Mean	Std	Mean	Std	Mean	Std
Path-realistic	4.12	0.9274	3.8	1.0408	4.24	0.9695	4.56	0.6506
Path-dynamic	4.292	0.8587	4.125	0.9918	4.208	0.658	4.708	0.6903
Path-visible	1.64	0.8602	3.08	1.115	3.12	0.9713	3.52	0.7703
Path-unoccluded	3.88	1.3329	3.08	1.6052	2.16	1.3748	4.24	1.2
Path-smooth	4.16	0.8981	3.88	1.0132	4.16	0.688	4.68	0.5568
Path-proximal	3.68	1.2152	4.28	0.8907	4.28	0.8426	4.28	0.8426
Location-compatible	4.28	1.0214	4.24	1.0116	4.48	0.7141	4.8	0.4082
Location-reasonable	3.81	1.3274	4.286	1.0071	4.286	0.8452	4.381	0.669
Effort-intuitive	4	1.0408	4.16	1.0677	4	1.1902	4.08	1.382
Effort-light	2.429	1.2479	3.905	0.9407	4.095	0.7003	4.476	0.8136
Effort-head steady	2.64	0.7	3.6	0.5	3.76	0.52281	3.92	0.4

Table 4. Mean and standard deviation for Location I. Bold text represents the highest mean among the path type.

	Random Traditional A*		Designer Traditional A*		ML Traditional A*		ML Modified A*	
Attributes	Mean	Std	Mean	Std	Mean	Std	Mean	Std
Path-realistic	3.2	1.354	4	1.291	4.56	0.5831	4.52	0.6532
Path-dynamic	4.125	0.9918	4.167	1.0901	4.5	0.7802	4.458	0.5882
Path-visible	1.44	0.6506	3.56	1.3254	4.04	0.9345	4.16	0.9866
Path-unoccluded	1.84	1.0677	3.68	1.4922	4.16	1.3128	4.36	1.036
Path-smooth	3.2	1.3844	4.12	1.2014	4.44	0.6506	4.56	0.7681
Path-proximal	3.04	1.4855	3.88	1.3329	4.52	0.6532	4.76	0.5228
Location-compatible	3.84	1.179	4.48	0.7141	4.72	0.4583	4.64	0.5686
Location-reasonable	3.381	1.4655	4.238	1.0911	4.476	0.6796	4.524	0.7496
Effort-intuitive	2.92	1.4411	4.12	1.2014	4.08	1.2557	4.08	1.382
Effort-light	1.524	0.6796	3.619	1.244	4.286	1.0556	4.048	1.0713
Effort-head steady	2.28	0.4583	3.92	0.9539	4.08	0.7024	4.2	0.6455

Table 5. Mean and standard deviation for Location II. Bold text represents the highest mean among the path type.

	Random Traditional A*		Designer Traditional A*		ML Traditional A*		ML Modified A*	
Attributes	Mean	Std	Mean	Std	Mean	Std	Mean	Std
Path-realistic	4.08	1.115	4.64	0.5686	4.68	0.5568	4.48	0.7703
Path-dynamic	4.458	0.658	4.333	0.8681	3.833	1.1293	4.458	0.779
Path-visible	3.12	1.1662	3.8	1.3844	4.76	0.5228	4.68	0.5568
Path-unoccluded	2.92	1.2557	4.4	1	4.56	0.9165	4.4	1.1902
Path-smooth	3.88	0.9274	4.44	0.9165	4.72	0.5416	4.48	0.8226
Path-proximal	4.4	0.7638	4.16	1.0279	4.72	0.6137	4.72	0.6137
Location-compatible	3.92	1.382	4.68	0.5568	4.68	0.6272	4.56	0.6506
Location-reasonable	4.048	1.244	4.429	0.8106	4.571	0.6761	4.476	0.7496
Effort-intuitive	3.84	1.2477	4.16	1.3128	4.4	1.3229	4.28	1.4866
Effort-light	3.952	0.9735	4.333	1.0165	4.619	1.0713	4.857	0.3586
Effort-head steady	3.84	0.7461	4.12	0.6658	4.72	0.45826	4.68	0.4761

Table 6. Mean and standard deviation for Location III. Bold text represents the highest mean among the path type.