Optimizing Visual Element Placement via Visual Attention Analysis

Rawan Alghofaili*
George Mason University

Michael S Solah
University of Massachusetts-Boston

Haikun Huang
University of Massachusetts-Boston

Yasuhito Sawahata
Japan Broadcasting Corporation

Marc Pomplun
University of Massachusetts-Boston

Lap-Fai Yu
George Mason University

Figure 1: Our approach analyzes a virtual environment and automatically places visual elements (e.g., paintings) at locations users are expected to pay attention to. Left: an input 3D scene with its corresponding layout. Right: the optimal placement of visual elements that will attain the target gaze duration. The eyes depict the camera location and angle in taking the screenshots.

Abstract

Eye-tracking enables researchers to conduct complex analysis on human behavior. With the recent introduction of eye-tracking into consumer-grade virtual reality headsets, the barrier of entry to visual attention analysis in virtual environments has been lowered significantly. Whether for arranging artwork in a virtual museum, posting banners for virtual events or placing advertisements in virtual worlds, analyzing visual attention patterns provides a powerful means for guiding visual element placement.

In this work, we propose a novel data-driven optimization approach for automatically analyzing visual attention and placing visual elements in 3D virtual environments. Using an eye-tracking virtual reality headset, we collect eye-tracking data which we use to train a regression model for predicting gaze duration. We then use the predicted gaze duration output of our regressors to optimize the placement of visual elements with respect to certain visual attention and design goals. Through experiments in several virtual environments, we demonstrate the effectiveness of our optimization approach for predicting gaze duration and for placing visual elements in different practical scenarios. Our approach is implemented as a useful plug-in that level designers can use to automatically populate visual elements in 3D virtual environments.

Index Terms: Computing methodologies—Computer graphics—Graphics systems and interfaces—Virtual Reality;

1 Introduction

A key challenge of designing a 3D virtual environment for an immersive experience is to predict the users' visual attention in the environment throughout their navigation. Such predictions can serve as a useful guide for placing visual elements—like artworks and visual hints—to enrich the immersive experience. Moreover, to sustain the growth of virtual reality content and satisfy the monetary needs of content providers, advertising in virtual spaces has become increasingly common. For example, Google is experimenting with virtual reality advertising [46]. In this paper, we explore using visual attention data for guiding the placement of visual elements in virtual environments, in an attempt to satisfy the aforementioned needs of content creators and designers.

In most previous visual attention studies, eye-tracking was performed using either mobile cameras or specialty equipment that track a user's visual attention on 2D images. However, with the introduction of consumer-grade virtual reality devices with eye-tracking capabilities—such as the FOVE virtual reality headset—the barrier of entry to visual attention and behavioral studies in 3D virtual environments shrank substantially [13]. In our work, we are interested in analyzing overt visual attention. The allocation of overt visual attention is often measured in terms of gaze duration on a given scene element. Here, gaze duration is defined as the time interval of viewing an element without shifting one's gaze [31]. While visual attention and eye movements are not identical, they are strongly correlated during natural tasks such as the ones considered here, making gaze duration a useful indicator of visual attention. In this work, we make use of the FOVE eye-tracking virtual reality headset for collecting gaze duration data to devise a data-driven optimization approach for visual elements placement.

Because placing visual elements is not the crux of level designers' work, we aim to provide level designers with a method to evaluate the placement of visual elements in 3D virtual environments. Specifically, we achieve this by training a regression model based on gaze duration data obtained from an eye-tracking virtual reality headset. The trained regressor can then be applied to compute the likelihoods of different locations in a virtual environment to be viewed by users in a navigation session. An optimization approach is subsequently run to select a set of locations for placing visual elements such as advertisements by considering visual attention and other design criteria encoded as cost functions. Figure 1 depicts an example.

Major Contributions:

- We propose a novel data-driven approach to train a regressor to estimate the distribution human visual attention in a 3D virtual environment, based on eye-tracking data obtained from virtual reality experiments.
- Based on our visual attention regressor, we devise an optimization approach for automatically placing visual elements

*e-mail: ralghofa@gmu.edu

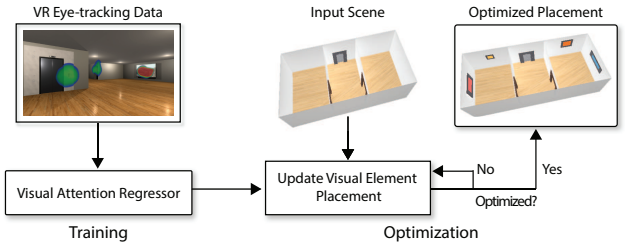


Figure 2: Overview of our approach.

at locations that are expected to receive high visual attention.

- We validate the effectiveness of our approach via a number of within-user experiments conducted in virtual environments. We also demonstrate how our novel approach could be employed to automatically place visual elements in a number of practical scenarios.

2 Related Work

Our problem formulation was inspired by recent works in optimizing interior scene layout [33, 48], lighting design [40], label placements in 3D scenes [18] and predicting gaze fixation [26].

2.1 Visual Attention and Panel Placement

In the domain of human-computer interaction and visualization, researchers have demonstrated how visual attention patterns may be used to guide the placement of different visual elements such as advertisement banners in a website [44], and to guide the generation of visual content such as *film comics* [38] –"a kind of art medium created by editing the frames of a movie into a book in comic style.". Our work was inspired by these studies.

Banner Blindness, a term coined by Benway et al. [5] and a heavily studied phenomenon in usability, occurs when website visitors consciously or subconsciously ignore banner advertisements. Burke et al. [10] studied banner blindness and its negative effects on visual memory and perceived cognitive workload. Franconeri and Simons [16] and Zhang [49] studied various banner animations and how they interfere with attention. Ignored banners have been proven to cost content creators substantial amounts of revenue. As virtual reality content continues to grow, we believe that it is important to explore this phenomenon in 3D virtual environments as well.

In HeatSpace [15], the authors used depth cameras to track users' positions and geometrically project their gaze estimated from head pose in a 3D room. This projected gaze is used to find the best panel placement by maximizing the likelihood that the panel could be seen (visibility). While in our approach, we try to maximize the likelihood that the panel would be seen (viewablity) using eye-tracking data.

The standard way to measure advertisement panel exposure is to count the number of passers-by [45]. This count is eventually adjusted according to some factors related to the panel's visibility. Factors such as how far the panel is from the road, the panel's height and lighting conditions all affect how visible the panel is to its target audience. We drew inspiration from these factors while defining our feature set for predicting gaze duration.

2.2 Eye-Tracking and Visual Attention

Eye-tracking is heavily employed to measure visual attention [43]. For example, Li et al. [30] found a relationship between gaze signals and user interest by using a built-in mobile camera to measure gaze duration while viewing areas of interest on a multi-column web page. Lagun and Agichtein [29] studied users' visual attention via web search behavioral studies. They validate their findings with eye-tracking results. Additionally, Ennis et al. [14] evaluated the distinctiveness and attractiveness of human virtual characters' gaits using gaze duration. Similarly, we use gaze duration to measure the extent of visual attention to visual elements.

Figure 3: Layout of *Museum L1*.

With the introduction of eye-tracking in virtual reality headsets like the FOVE, platforms like CognitiveVR provide tools for developers to simplify behavioral analysis using virtual reality headsets. Furthermore, companies such as Pupil Labs build eye-tracking devices as well as add-ons for the HTC Vive virtual reality headset that extend its capabilities to support eye- tracking.

Numerous computational models have been devised for predicting the allocation of attention based on the distribution of visual features in a scene [7]. Eye-tracking datasets [8, 11] were created to measure the accuracy of visual attention models [27]. The data was recorded by a camera-based eye tracker. Unlike our approach which is trained on gaze data collected from 3D virtual environments, these datasets and their corresponding models were created to predict visual attention on 2D images. Conversely, like our approach several models [24, 35] have been devised to analyze the allocation of attention in 3D scenes.

Design optimizations based on visual attention panels were explored in several previous works. Pang et al. [36] devised a method to optimize web designs by directing visitor's visual gaze. Similarly, Cao et al. [12] optimizes Manga panel placements according to a desired gaze flow pattern. The aforementioned works optimize placements of elements to satisfy a desired visual attention flow, while our approach optimizes element placement in a virtual environment to satisfy a total amount of visual attention (gaze duration).

Hillaire et al. [19] modeled visual attention in a 3D environment. They used data collected from a gaze tracker of subjects viewing a flat screen to verify their model's accuracy. Jensen et al. [22] used a wearable device to track the subject's gaze as he/she moved in a real world 3D space. However, both of these models have not been utilized to optimize visual attention in 3D spaces.

2.3 Advertisement Placement in Virtual Reality

Several efforts have been made to enhance advertisements in virtual reality. Bates et al. [4] optimized advertisements' content based on crowd statistics within a virtual environment. Hyndman et al. [20] catered advertisements according to the environment's context. Kusumoto et al. [28] incentivized players to place advertisements in their own virtual spaces, while Kimsey [25] proposed the strategy of presenting advertisements concealed as player tasks. These efforts have provided early insights about enhancing advertisements in 3D virtual environments. However, the proposed methods rely solely on designer intervention in placing advertisements. Advertisement locations are selected manually by the level designer without any metrics to inform how effective the selected locations could be with regard to attracting visual attention. In contrast, we aim to automate and facilitate visual element placement for level designers by using a quantitative data-driven approach.

3 Overview

Figure 2 shows an overview of our approach. Our visual attention regressor is trained with eye-tracking data obtained from users who navigated in virtual environments during our data collection sessions (Section 4). Essentially, the regressor learns the relationship between the features characterizing locations in a 3D virtual environment and the visual attention received by elements placed at those locations

Subway L1 & L2		
Mission	**Task**	**Start**
1	Find North Station	Platform 1
2	Find Boston College	Platform 2
3, 4	Find the Boston University exit	Platform 1, 2
5, 6	Find the Fenway Park exit	Platform 1, 2
7, 8	Find the bus stop exit	Platform 1, 2
Museum L1 & L2		
Mission	**Task**	**Start**
9, 10	View paintings in 3 minutes; take a quiz	Stairs, elevators, or escalators

Table 1: The 10 missions displayed to participants during data collection. Section 4 discusses details of the mission formats.

(Section 5). Given a new virtual environment as input, the trained regressor can then be applied to automatically predict the amount of visual attention (in terms of gaze duration) that will be received. An optimization approach is run to automatically place visual elements (e.g., pictures) in the 3D environment to match their combined expected visual attention with a specified target (Section 6).

To facilitate our discussion, we use a scene called *Museum L1*, which mimics a level of the Museum of Modern Art in New York, as our running example to illustrate most parts of our approach. Figures 1 and 3 show its screenshots and layout.

We implemented our approach using Python and C#. We created a plug-in for the Unity 5 game engine, which level designers can use to automatically populate a virtual environment with visual elements.

4 Data Collection Experiment

We will describe the participants, apparatus, tasks and scenes utilized for our data collection experiment.

Participants. To collect the gaze duration data, we recruited 23 participants to complete our experiments. The participants were mostly students and university staff whose ages ranged from 19 to 30 years old. Each participant was asked to navigate the environment with a certain mission in mind (e.g., go from here to there). The whole navigation session was recorded which included the participant's position and gaze. The experiment was approved by the Institutional Review Board (IRB) of the university.

Apparatus. To navigate in the environment, each participant wore a FOVE virtual reality headset which gave us the ability to track his/her head orientation and gaze. The FOVE had a built-in infrared eye-tracking system that operates with a frame rate of 120 fps. An Internal Measurement Unit (IMU) was used to track the head orientation and an infrared sensor was used to track the user's gaze. It displayed visuals at a frame rate of 70 fps. It has a tracking accuracy of less than one degree, significantly smaller than the dimensions of our visual elements. The FOVE was calibrated at the beginning of the experiment and each time the participant removed the headset. Additionally, the user controlled his/her locomotion using a game controller akin to [37] and [47].

Scenes. We created 3D virtual environments and placed visual elements at different locations within them. These scenes were realistic and created by referencing real-world architectural layouts like the Kenmore subway station in Boston and the Museum of Modern Art (MOMA) in New York shown in Figure 10. Using these environments, we defined four *scenes* comprising *Subway L1*, *Subway L2*, *Museum L1*, and *Museum L2*. The two-floored *Subway* scenes were taken from two different placements of visual elements in the subway station. *Museum L1* and *L2* were based on two levels of the museum. Candidate locations where visual elements (e.g., advertisements for *Subway*, artworks for *Museum*) would be placed were preliminary determined; the first and second *Subway* scenes contained 24 and 54 visual elements respectively, while *Museum L1* and *L2* contained 41 and 38. The supplementary material shows some examples of visual elements shown to participants during the data collection experiment.

(a) Mission Screen

(b) FOVE VR Headset

Figure 4: Participants of our data collection experiment completed missions while wearing the eye-tracking VR headset. Their positions and gaze were recorded and later used to train our visual attention regressor. The red dot shows this participant's gaze point.

During the data collection experiments, participants were asked to complete missions in these two floored scenes. In *Museum L1* and *L2*, participants were moved from one floor to the other after the mission was complete. In the *Subway* scenes, participants could willingly move from one floor to the other using the stairs, elevators or escalators, as shown in the supplementary material.

Missions. Each participant was asked to navigate the environment with a certain mission in mind (e.g., view paintings within a specific amount of time for the museum scenes, or find a specific exit for the subway scenes). The whole navigation session was recorded which included the participant's position and gaze.

We designed a total of 10 missions that replicate realistic navigation scenarios. Table 1 shows the two categories of missions.

- *Subway L1 & L2:* we cycled through the 8 missions shown in Table 1, giving each participant two consecutive missions from the two subway station scenes.
- *Museum L1 & L2:* we created missions 9 and 10 for *Museum L1* and *Museum L2* respectively. We asked all participants to complete missions 9 and 10. We requested that they looked around and informed them that they would be quizzed on which images they saw in the museum. The task was stopped after 3 minutes.

To ensure that our missions produce realistic paths, the participant was initially dropped at an entry location like the stairs, elevators or escalators in the *Museum L1 & L2* example; or in a random location from a pool of predefined locations on one of the platforms of *Subway L1* to simulate exiting a train. We assigned each mission the starting platform that would ensure participants will traverse the longest distance possible.

For *Museum L1* (Figure 3 shows its layout), each participant was asked to navigate the museum level like visiting an art gallery in the real world. In this case, the visual elements refer to paintings. This task was limited to 3 minutes and participants were incentivized to notice artwork in their navigation by informing them that they will take a short quiz about the artwork after sessions. In reality, no quiz assignments were given to the participants after the experiment as we only needed to and had already collected their eye gaze data for training our regressors.

As for *Subway L1 & L2*, assuming the participant started out with mission 1, he/she would be dropped on Platform 1 and asked to find his/her way to North Station. After completing the task, the participant would enter the second *Subway* scene (has the same layout as the previous, with a different arrangement of advertisements) and would be given mission 2. The participant would enter the station on Platform 2 and would be tasked to find the way to Boston College.

We displayed the missions for participants through a FOVE virtual reality headset (Figure 4). The participants completed the missions by navigating through the scene using an Xbox game controller as we tracked their eye-movements using the headset. It took each

participant on average 11 minutes and 13 seconds to complete the entire experiment.

5 Training of Visual Attention Regressor

Based on the eye gaze data collected from users navigating the virtual environments, we train a visual attention regressor to predict the duration of eye gaze received by a visual element placed at a certain location in a 3D virtual environment. We discuss the training of our regressor in this section.

5.1 Data

We measured the gaze duration of each visual element received from participants while navigating our scenes. A visual element could receive several gaze durations from a participant, as he/she could view them multiple times while completing the mission. In this case, the element was recorded multiple times (with different gaze durations) in our dataset. In other words, visual elements can be visited more than once. Elements not viewed by the participant were assigned a gaze duration of zero.

5.2 Training and Prediction

We use the gaze duration of each visual element obtained in our data collection session as the target of our regression model. In a nutshell, each data sample in our dataset refers to an element, represented as an 12-feature vector as defined in the following section with a gaze duration measured from every user.

We experimented with different regressors (e.g., Decision Trees, Random Forests, Support Vector Machines) for learning the relationship between the environment features and the amount of visual attention. Because the Random Forest regressor gave us the lowest error empirically, we chose to use it in our optimization. We provide more details of the training results in Section 7.1.

Given a new 3D scene with the candidate locations for placing a visual element, our trained regressor predicts the gaze duration of placing the element at each location. This predicted gaze duration is used for optimizing the placement of visual elements.

5.3 Features

For each visual element placed in the environment, we extract a number of features that we use to train our regressor. These features were designed as per a consultation interview with 5 experts—museum curators and interior designers—and referencing spatial design books [6, 34]. Our features allow a visual element placed at a certain location to be assigned a predicted gaze duration by the regressor. We describe the features we use as follows:

- *Element's Dimension:* The element's rectangular width and length each as a separate feature. In general, larger elements receive more attention than smaller elements. It could be aesthetically pleasing to display varying sizes of visual elements. Hence, we provide the designer the flexibility of combining multiple sizes of visual elements.
- *Element's Height From Ground:* The distance between the ground and the element's centroid, i.e., how high the element is placed on the wall. Experts follow specific guidelines pertaining to element visual accessibility [32]. As we can adjust the player's height in virtual reality, we can ignore the visible accessibility aspect of the element's height and just focus on the visual attention it receives.
- *Locations Relative to Places of Interest:* We measure the minimum distance between the element and entrances/exits; the element and stairs; the element and escalators; and the element and elevators. This gives us four feature values with the entrances and exits assumed to be equivalent. Much like the *element height* feature we ignore accessibility (e.g., blocking emergency exits) and focus on visual attention.
- *Location Relative to Room Center:* The location of the element's centroid relative to the room center as the origin. Visual elements that are placed in spacious locations generally receive more visual attention than others. Because it is computationally expensive to measure the amount of empty space in front of large numbers of elements placed in sizable virtual spaces, we use this feature—along with the *occlusion metric*—as an approximation.
- *Occlusion Metric:* We uniformly project 100 rays from the element's centroid onto a hemisphere with a 5-meter radius, and count the number of collisions due to occlusions. This feature gives us an indicator on how well the element can be viewed from different angles.
- *Lighting Metrics:* We measure the effect of environment lighting on visual elements. First, we determine the minimum distance between our element and light sources (point or spot lights with constant intensity and range) placed in the environment. To find the distance to each light source, we cast a ray from the visual element to each light source. We ignore any light source that is occluded by another object as it has little influence on the element's illumination. We use the distance to the nearest light source as the first lighting metric. In addition, we record the number of unblocked light sources that are within 15 meters of the visual element as the second lighting metric. The first and second metrics are included as two separate features. Experts use lighting to highlight pieces that they wish to receive attention. Therefore, we give the designer the flexibility of defining their lighting locations.
- *Distance from Nearest Path:* We specify a set of paths (shown in the supplementary material) that users are likely to traverse and calculate the distance from the closest path as a feature value. It is common practice for experts to blueprint visitor-flow [42] prior to determining placement of elements. These visitor-flow paths reflect the experts' estimation of how the space should be traversed by visitors. In designing visitor-flow, experts may consider heavy traffic when they are concerned with the safety, security and accessibility of the environment (e.g., blocking emergency exits). However, they do not consider crowds in blueprinting visitor-flow. Therefore we assume that our virtual environments are not populated by crowds.

6 Optimizing Visual Elements Placement

Given the trained visual attention regressor, we devise an optimization framework for automatically placing visual elements in a 3D virtual environment.

6.1 Cost Function

Suppose the virtual environment has a set $L = \{l_i\}$ of candidate locations (e.g., panels) for placing visual elements. The candidate locations were selected using a procedural layout technique akin to [48] and manually refined by the designer. Using the trained visual attention regressor, we can predict the gaze duration $T(l_i)$ that a visual element placed at location l_i receives. For formulation simplicity, we assume that the visual element takes up the dimension of the panel and is centered at the selected location.

Total Cost Function. Let $L_S = \{l_i\}$ be a placement solution, where each l_i refers to a selected location for placing a visual element and $L_S \subset L$. We define the total cost function for evaluating the placement solution as:

$$C_{\text{Total}}(L_S) = w_G C_G(L_S) + w_R C_R(L_S) + w_P C_P(L_S), \quad (1)$$

where C_G is a goal-specific cost, C_R is a regularization cost and C_P is a prior cost; and w_G, w_R and w_P are their respective weights. We describe details of the cost terms in the following.

Goal-Specific Cost. The goal-specific cost is defined to encode the major goal that the designer wants the visual elements placement

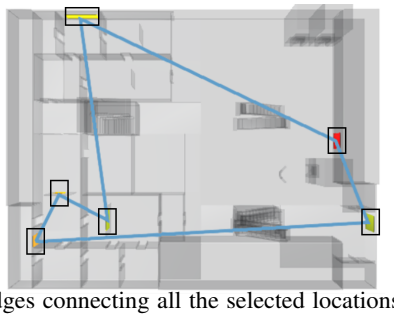


Figure 5: Edges connecting all the selected locations found by a nearest neighbor search starting from the leftmost location.

to achieve. One common goal is that the visual elements should altogether attract a target gaze duration:

$$C_{\mathrm{G}}(L_{\mathrm{S}}) = \frac{1}{T'}|\sum_{l_i} T(l_i) - T'|, \tag{2}$$

where $T(l_i)$ returns the gaze duration predicted by the trained regressor for a visual element placed at location l_i; T' is the target gaze duration. Note that the goal-specific cost could be redefined to fit with the designer's goal if necessary. We show some examples in our experiments in Section 8.

Regularization Cost. Generally the designer may prefer the selected visual elements locations to be distributed evenly in the scene. We define a regularization cost to allow such consideration:

$$C_{\mathrm{R}}(L_{\mathrm{S}}) = \frac{1}{|E|D}\sum_{e_i \in E}(e_i - \bar{e})^2, \tag{3}$$

where $E = \{e_i\}$ is a set of edges that form a path going through each of the selected locations $l_i \in L_{\mathrm{S}}$; $\bar{e}$ is the average length of the edges in E; D is the normalization constant and is set as the squared diagonal of the bounding rectangle of the environment's floor plan. The path is found by a nearest neighbor search which approximates the shortest path for the traveling salesman problem by always choosing the nearest unvisited node (i.e., nearest location in our case) as the next move. We set the search to start from the leftmost selected location, and an extra edge is added to connect the last visited location to the starting location. See Figure 5 for an illustration.

Prior Cost. We also define a prior cost for constraining the number of visual elements used:

$$C_{\mathrm{P}}(L_{\mathrm{S}}) = \exp(-\frac{1}{2\sigma^2}(|L_{\mathrm{S}}| - n)^2), \tag{4}$$

where n is the prior number of visual elements used and σ, which controls the spread of the Gaussian penalty function, is empirically set as 1.0.

6.2 Optimization

To extensively explore the solution space, we solve the optimization by using a Markov chain Monte Carlo technique, namely, simulated annealing with Metropolis-Hastings state searching steps. As a variable number of locations L_{S} can be selected from the set of all possible locations L, the optimization needs to be performed in a trans-dimensional solution space. To this end, we apply the reversible-jump Markov chain Monte Carlo technique [17], which can handle changing dimensionality. First, we define a Boltzmann-like objective function:

$$f(L_{\mathrm{S}}) = \exp(-\frac{1}{t}C_{\mathrm{Total}}(L_{\mathrm{S}})), \tag{5}$$

where t is the temperature parameter for simulated annealing. The optimization proceeds iteratively. At each iteration, a move is proposed to alter the current placement solution L_{S} to a proposed placement solution L'_{S}. There are three types of moves:

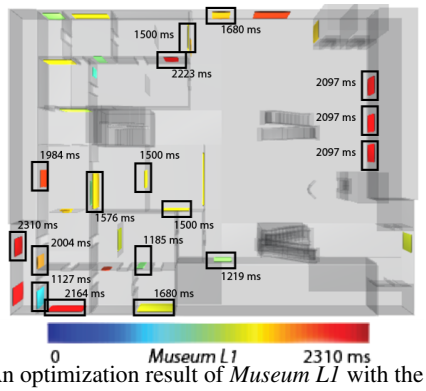


Figure 6: An optimization result of *Museum L1* with the target gaze duration set as 30,000 ms. The colors correspond to the gaze duration predicted by the regressor. The selected locations for placing visual elements, whose predicted gaze duration sums to 29,941 ms, are enclosed.

- *Add*: a random location $l_i \in L - L_{\mathrm{S}}$ is added to the set of selected locations L_{S}, such that $L'_{\mathrm{S}} = L_{\mathrm{S}} \cup \{l_i\}$;
- *Remove*: a random location $l_i \in L_{\mathrm{S}}$ is removed from the set of selected locations L_{S}, such that $L'_{\mathrm{S}} = L_{\mathrm{S}} - \{l_i\}$;
- *Modify*: a random location $l_i \in L_{\mathrm{S}}$ is removed, and a random location $l_j \in L - L_{\mathrm{S}}$ is added to the set of selected locations L_{S}, such that $L'_{\mathrm{S}} = (L_{\mathrm{S}} - \{l_i\}) \cup \{l_j\}$ and $l_i \neq l_j$.

The *Add*, *Remove* and *Modify* moves are respectively selected with probabilities p_{a}, p_{r} and p_{m}. In our implementation, we set $p_{\mathrm{a}} = 0.4$, $p_{\mathrm{r}} = 0.2$ and $p_{\mathrm{m}} = 0.4$ to slightly favor adding and modifying locations.

The total cost $C_{\mathrm{Total}}(L'_{\mathrm{S}})$ of the proposed placement solution L'_{S} is compared with the total cost $C_{\mathrm{Total}}(L_{\mathrm{S}})$ of the current placement solution L_{S}. The proposed solution L'_{S} is accepted with the following acceptance probability $\alpha(L'_{\mathrm{S}}|L_{\mathrm{S}})$ set according to the Metropolis criterion to maintain the detailed balance condition:

For an *Add* move,

$$\alpha(L'_{\mathrm{S}}|L_{\mathrm{S}}) = \min\left(1, \frac{p_{\mathrm{r}}}{p_{\mathrm{a}}}\frac{|L - L_{\mathrm{S}}|}{|L'_{\mathrm{S}}|}\frac{f(L'_{\mathrm{S}})}{f(L_{\mathrm{S}})}\right), \tag{6}$$

For a *Remove* move,

$$\alpha(L'_{\mathrm{S}}|L_{\mathrm{S}}) = \min\left(1, \frac{p_{\mathrm{a}}}{p_{\mathrm{r}}}\frac{|L_{\mathrm{S}}|}{|L - L'_{\mathrm{S}}|}\frac{f(L'_{\mathrm{S}})}{f(L_{\mathrm{S}})}\right), \tag{7}$$

For a *Modify* move,

$$\alpha(L'_{\mathrm{S}}|L_{\mathrm{S}}) = \min\left(1, \frac{f(L'_{\mathrm{S}})}{f(L_{\mathrm{S}})}\right), \tag{8}$$

By default, we set the temperature t as 1.0 at the beginning of the optimization, which decreases by 0.2% every iteration until it reaches a small value of 0.005 after which it remains the same. Essentially, this allows the optimizer to explore the solution space aggressively at the beginning, while refining the solution in a more greedy fashion in the later stage. The optimization terminates when the absolute change in total cost is less than 5% over 50 iterations. We set the weights as $w_{\mathrm{G}} = 1.0$, $w_{\mathrm{R}} = 0.1$ and $w_{\mathrm{P}} = 0.1$ in our optimization unless specified otherwise. In practice these weights can be controlled by the designer to emphasize different cost terms based on design needs. Figure 6 shows an optimization result of *Museum L1* and Figure 7 shows a plot of the change in total cost during a run of the optimization. We show optimization results for additional scenes in Section 8.

For the running example (*Museum L1*), it takes on average 180 iterations to finish the optimization with a target gaze duration of 30 seconds. The optimization takes about 0.2 seconds to finish using our current implementation with a workstation equipped with 3.6GHz Intel Core i7 processor and 16GB of RAM.

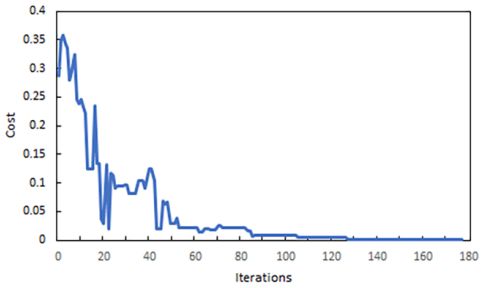


Figure 7: Change in cost over an optimization run of *Museum L1*.

	RMSE(%)	RMSE(ms)
Support Vector Regressor	7.54	2,002
Decision Tree	6.77	1,797
Random Forest	6.70	1,780

Table 2: Prediction error of our *test set* of 762 examples using different types of regressors. The ms error was computed by scaling up the $[0,1]$ output of our regressors by the maximum value (2,655 ms)

7 Regressor Results

We discuss the results of training our visual attention regressor. An analysis of our regressor's performance is provided in the supplementary material.

7.1 Training Results

At each participant's session, a visual element was assigned a gaze duration record for each uninterrupted viewing period. This may result in multiple gaze durations for each visual element, all of which are included in our dataset. Any visual element not viewed during a session was assigned a zero gaze duration. The above assignments resulted in a total of 3,045 data samples.

Prior to training our regressors we performed *L2*-normalization on our dataset. We then randomly sampled a 762 test set (about 25% of the entire dataset) prior to training our regressor. The remaining samples were used for training. We trained the regressors with scene-specific data samples but did not achieve a lower RMSE.

We experimented with training different types of regressors. For support vector machine, we use an Epsilon-Support Vector Regressor (ε-SVR) [39] with an ε of 0.01 and an error term penalty parameter C of 1,000. For decision tree, we set the maximum depth to 5. For random forest [9], we set the maximum depth to 5 for all 5 trees in the forest. We used these hyper-parameters as they produced the highest accuracy using grid-search with 10-fold cross-validation.

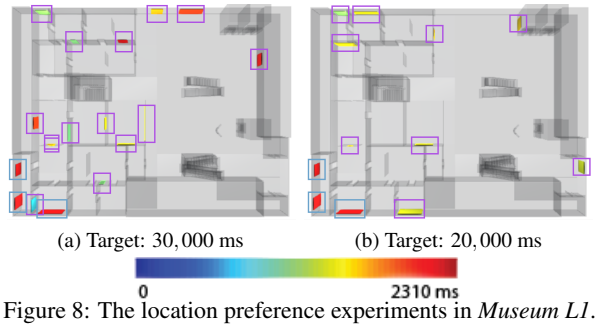
In a 10-fold cross-validation done on the 3,045 sample *training set*, we obtained a root mean squared error of 2,226 ms, 2,026 ms and 2,003 ms for the support vector machine, decision tree and random forest respectively. Table 2 shows that the random forest regressor attains the smallest root mean square error on our isolated randomly sampled *test set*. We chose to use the random forest regressor in our optimization because it attained the lowest prediction error as well as the highest correlation coefficient (0.64).

We tried training regressors per scene, but did not attain a lower RMSE. This and other results comparing the ground-truth gaze duration with the predicted gaze duration for *Museum L1*, *Museum L2* and *Subway L1 & L2*, are provided in the supplementary material.

8 Optimization Results

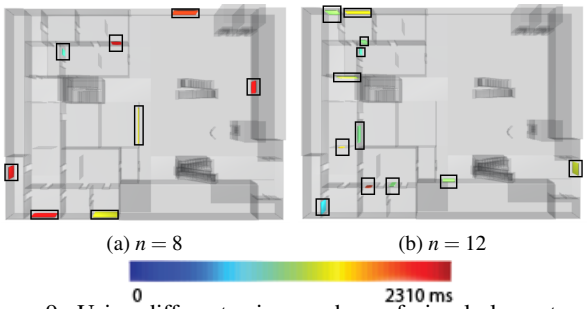
Our optimization framework for choosing visual element locations provides ample flexibility for the designer to generate solutions that fit with different goals or constraints. We demonstrate its use to tackle some practical scenarios.

Target Duration of Visual Attention. Our approach allows the designer to specify an accumulated target gaze duration T' that he/she wants our optimizer to achieve by automatically selecting a set of visual elements whose predicted gaze duration meets the target. This functionality could be useful for advertisers who want to place advertisements in a shopping mall or a subway station to attract a certain amount of visual attention for marketing purposes. Figure 11 shows optimization results on three different scenes (refer to Figure 10 for their 3D views), each with a different target accumulated gaze duration. Please refer to the figure captions for the experiment settings and results. In all four examples, the target total gaze durations are successfully met by the selected set of visual elements, with an error of about 0.1 second or less.



(a) Target: 30,000 ms (b) Target: 20,000 ms

Figure 8: The location preference experiments in *Museum L1*.

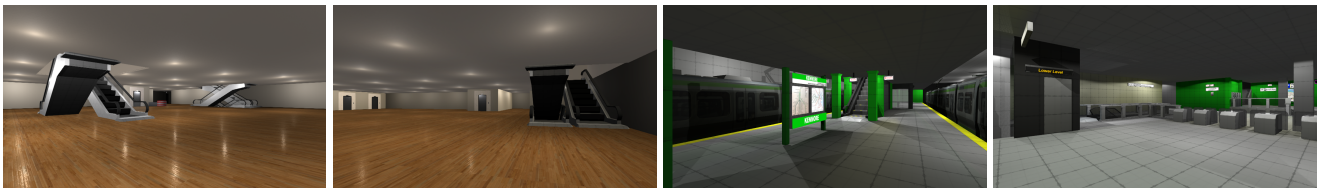


(a) $n = 8$ (b) $n = 12$

Figure 9: Using different prior numbers of visual elements n in *Museum L1*.

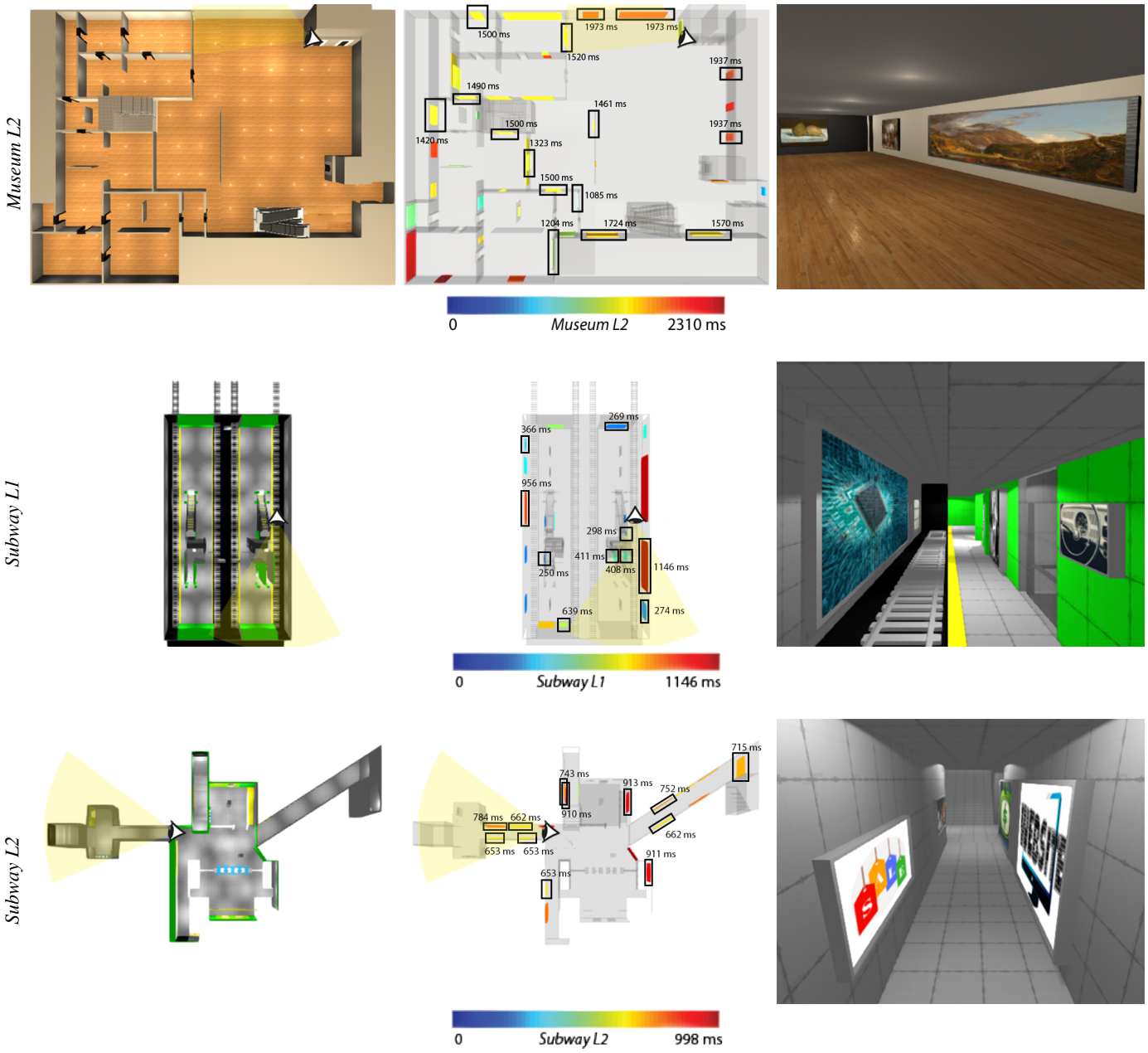
Varying Number of Visual Elements. Our approach also allows the designer to specify a prior number of visual elements n to be used in the placement solution, which is encoded as a soft constraint by Equation 4. This functionality could find practical applications. For example, in a virtual subway station, an advertiser may choose to place a lot of advertisements at various locations, or a small set of advertisements at a few eye-catching locations, to attain a certain target total duration of visual attention. Our approach allows choosing either strategy by changing n. Two optimization results generated with different prior numbers of visual elements in *Museum L1* are shown in Figure 9. Both results were optimized with the total target gaze duration set as 15,000 ms. The total predicted gaze duration of the 8 selected visual elements in (a) is 14,957 ms, while that of the 12 selected visual elements in (b) is 15,030 ms, both close to the total target duration of 15,000 ms.

Location Preference. In some scenarios, the designer may want the solution to include visual elements placed at certain locations. For example, an advertiser may want at least one advertisement to be placed near the entrance of a subway station regardless of where the rest of the advertisements are placed. Our approach can easily achieve this feature by hard constraints: the visual element locations that must be included are added to the initial solution set; these visual element locations are kept unchanged (i.e., not modified by any moves) while the optimizer modifies the rest of the visual element locations. Figure 8 shows two examples. In each example, three visual element locations (blue) are fixed by the designer, and the optimizer is asked to choose the rest of the visual element locations (purple) to attain the target accumulated gaze duration (30,000 ms and 20,000 ms). The total predicted gaze durations of the selected locations are 29,992 ms and 20,026 ms



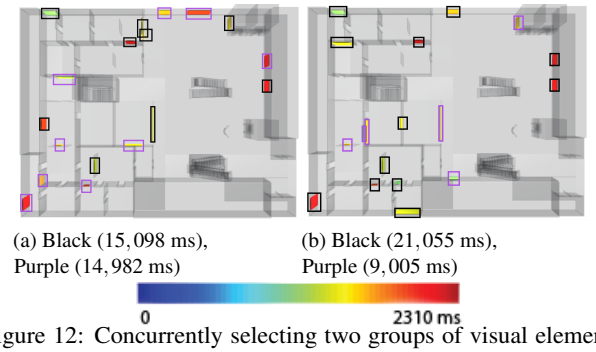
Museum L1 *Museum L2* *Subway L1* *Subway L2*

Figure 10: 3D views of the input scenes.



(a) Floor Plan (Input) (b) Floor Plan (Result) (c) 3D View (Result)

Figure 11: Selecting visual elements to attain a total gaze duration target. (a) Input scenes with no visual elements placed. (b) Candidate locations that the optimization will consider while placing visual elements—red indicates high predicted gaze duration while dark indicates low predicted gaze duration. The selected locations by the optimizer are enclosed. The target accumulated gaze duration for *Museum L2*, *Subway L1* and *Subway L2* are set as 25,000 ms, 5,000 ms, and 9,000 ms, while the total predicted gaze durations of the locations selected by the optimizer are 25,117 ms, 4,987 ms, and 9,011 ms respectively, which are close to the targets. (c) 3D views of the scenes with the visual elements placed according to the optimization results. The screenshots in (c) are captured from the cameras in (b).



(a) Black (15,098 ms), Purple (14,982 ms)

(b) Black (21,055 ms), Purple (9,005 ms)

Figure 12: Concurrently selecting two groups of visual elements with two target gaze durations within a single optimization in *Museum L1*.

respectively in (a) and (b), which are close to the targets.

Groups of Visual Elements. Our approach can also deal with the scenario where several groups of visual elements (e.g., from several companies) need to be placed in the same space. To achieve this, the designer specifies the target total gaze duration that should be received by each group. Our formulation can be easily extended to handle the scenario by including multiple targets in Equation 2 instead of one target. Figure 12 shows two examples based on *Museum L1*. In (a), the optimizer is asked to select two visual element groups with each group receiving roughly the same amount of visual attention (15,000 ms and 15,000 ms). It selected visual elements (black) whose predicted gaze duration sums to 15,098 ms and another group (purple) whose predicted gaze duration sums to 14,982 ms. In (b), the optimizer is asked to select two groups with each group receiving a different amount of visual attention. The targets are set as 21,000 ms and 9,000 ms. The optimizer selects a group (black) whose predicted gaze duration sums to 21,055 ms and another group (purple) whose predicted gaze duration sums to 9,005 ms.

9 Discussion

Limitations. Although using eye-tracking VR headsets for visual attention experiments seems like a promising endeavor, it is still a long way from completely replacing specialty eye-trackers. The FOVE headset still has room for improvement in terms of precision; currently it has too high of a latency to be utilized in elaborate behavioral studies. Moreover, virtual reality headsets are more restrictive and less comfortable than specialty eye-trackers, and some cannot be worn with glasses. These factors could skew experiment results, especially ones that require a high level of detail. Our experiments were conducted for a relatively short time due to these reasons.

There is room for improvement on our feature set as well. For example, our lighting feature could be improved to include light intensity, shading and reflection to account for the element's visibility. Similarly, we have not integrated complex features like textures into our model. The distance from the nearest path metric captures the relation between the participants' paths and the element's visibility. We specified these paths based on our knowledge of the predefined tasks and desired visitor-flow. In practice, these paths might be estimated based on visitor navigation statistics recorded from the real world [1, 42] or determined by experienced wayfinding designers.

We collected data from only four scenes and 23 subjects. Collecting data from more scenes will expand our approach's ability to generalize. Moreover, we tested our approach on scenes built manually based on floor plans. Provided with more detailed architectural layouts, or by scanning and reconstructing 3D scenes from the real world, we will be able to test our approach in a more realistic setting and validate its ability for placing visual elements effectively in real-world scenes. Furthermore, in validating our regressor we held out random data samples. With more subjects, samples can be held out per subject to determine how well the regressor can generalize across participants it has not encountered.

Future Work. With simple regressors and a reasonably-sized dataset, we were able to achieve a relatively low RMSE. The physical set-up of our data collection sessions posed challenges for us to collect gaze data from a lot of users. In future work, with more widespread support of WebVR and availability of eye-tracking VR headsets, we could possibly scale up the virtual reality-based eye-tracking experiments via crowdsourcing. While we demonstrate the idea of using eye-tracking data to optimize visual element placement in virtual environments through our framework, with a large-scale eye gaze dataset, utilizing a deep neural network could improve our framework's performance.

While our simple features identified through interviews are commonly utilized by curators and spacial designers in deciding visual element placements, we could extend our optimization-based design framework by incorporating other perceptual factors and functionalities that may be important in the real-world spacial design process. With such extension, our framework might be able to facilitate the design of real-world architectural spaces. For example, an architect could be presented with the visual elements' placement suggestions that could yield the highest financial gain before making any physical changes to or even constructing an architectural space.

Moreover, art curators often utilize salon-style (grid-like) hanging of artwork. This style of hanging artwork creates competition among artists and the curator must decide which artist will receive the best "eye level" or "on the line" location for their paintings [3]. With some modifications to our feature set we could provide curators with a tool to automatically optimize element placement according to exhibition accessibility design guidelines [32]. Our approach could provide curators with a way to evaluate painting placements quantitatively. Similarly, marketing firms could use our model to generate effective advertisement placements for their campaigns.

Another avenue for future work is to use memorability [21], importance [41] or visual saliency [23] of the visual element's content as additional features for predicting visual attention, which could enhance our framework's applicability.

We designed our method to predict and optimize 2D visual elements like paintings or advertisements in a 3D virtual environment void of any stimuli (e.g. crowds, traffic, visual aids). Some visual stimuli [2] were shown to decrease the overall gaze points the scene receives. More experiments need to be done to determine if this phenomenon extends to visual elements placed in the scene. We have not considered 3D visual elements like statues in our implementation. While we implemented our approach to optimize the placement of static objects, in future work it would be interesting to extend our approach to consider moving elements as well.

10 Summary

In this work, we captured users' gaze data via an eye-tracking VR headsets. We used the collected eye gaze data to train a visual attention regressor which is capable of predicting gaze duration for candidate locations based on features such as height from the floor, occlusion, lighting and distance from navigation paths.

Based on the regressor, we devised a novel data-driven optimization approach for automatically placing visual elements in a virtual environment by considering a number of design criteria. Experiments showed that the regressor can reasonably predict gaze duration towards elements placed at various locations, and that our approach can be used for effectively placing elements for various practical scenarios.

Acknowledgments

This research is supported by the National Science Foundation under award number 1565978. We thank the anonymous reviewers for their constructive comments.

REFERENCES

[1] A. Alahi, K. Goel, V. Ramanathan, A. Robicquet, L. Fei-Fei, and S. Savarese. Social lstm: Human trajectory prediction in crowded spaces. In *CVPR*, 2016.

[2] R. Alghofaili, Y. Sawahata, H. Huang, H.-C. Wang, T. Shiratori, and L.-F. Yu. Lost in style: Gaze-driven adaptive aid for vr navigation. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19. ACM, 2019.

[3] H. Allen. Art that stacks up. *the Washington Post*, 2004.

[4] C. Bates, J. Chen, Z. Garbow, and G. Young. Advertising in virtual environments based on crowd statistics, Dec. 30 2014. US Patent 8,924,250.

[5] J. P. Benway and D. M. Lane. Banner blindness: Web searchers often miss obvious links. *Itg Newsletter*, 1(3):1–22, 1998.

[6] E. Bogle. *Museum Exhibition Planning and Design*. EBSCOhost ebooks online. AltaMira Press, 2013.

[7] A. Borji and L. Itti. State-of-the-art in visual attention modeling. *IEEE transactions on pattern analysis and machine intelligence*, 35(1):185–207, 2013.

[8] A. Borji and L. Itti. Cat2000: A large scale fixation dataset for boosting saliency research. *CVPR 2015 workshop on "Future of Datasets"*, 2015. arXiv preprint arXiv:1505.03581.

[9] L. Breiman. Random forests. *Machine Learning*, 45(1):5–32, 2001.

[10] M. Burke, A. Hornof, E. Nilsen, and N. Gorman. High-cost banner blindness: Ads increase perceived workload, hinder visual search, and are forgotten. *ACM Trans. Comput.-Hum. Interact.*, 12(4):423–445, Dec. 2005.

[11] Z. Bylinskii, T. Judd, A. Borji, L. Itti, F. Durand, A. Oliva, and A. Torralba. Mit saliency benchmark. http://saliency.mit.edu/.

[12] Y. Cao, R. Lau, and A. B. Chan. Look over here: Attention-directing composition of manga elements. *ACM Transactions on Graphics (Proc. of SIGGRAPH 2014)*, 33, 2014.

[13] A. T. Duchowski. Eye tracking methodology. *Theory and practice*, 328, 2007.

[14] C. Ennis, L. Hoyet, and C. O'Sullivan. Eye-tracktive: Measuring attention to body parts when judging human emotions. *Eurographics 2015 (Short papers)*, pp. 37–40, 2015.

[15] A. Fender, D. Lindlbauer, P. Herholz, M. Alexa, and J. Müller. Heatspace: Automatic placement of displays by empirical analysis of user behavior. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, UIST '17, pp. 611–621. ACM, New York, NY, USA, 2017.

[16] S. L. Franconeri and D. J. Simons. Moving and looming stimuli capture attention. *Perception & Psychophysics*, 65(7):999–1010, 2003.

[17] P. J. Green. Reversible jump markov chain monte carlo computation and bayesian model determination. *Biometrika*, 82(4):711–732, 1995.

[18] K. Hartmann, T. Götzelmann, K. Ali, and T. Strothotte. Metrics for functional and aesthetic label layouts. In *Proceedings of the 5th International Conference on Smart Graphics*, SG'05, pp. 115–126. Springer-Verlag, Berlin, Heidelberg, 2005.

[19] S. Hillaire, A. Lecuyer, T. Regia-Corte, R. Cozot, J. Royan, and G. Breton. Design and application of real-time visual attention model for the exploration of 3d virtual environments. *IEEE Transactions on Visualization and Computer Graphics*, 18(3):356–368, March 2012.

[20] A. Hyndman, N. Sauriol, and G. WALLS. Advertising in a virtual environment, Oct. 11 2012. US Patent App. 13/081,070.

[21] P. Isola, J. Xiao, A. Torralba, and A. Oliva. What makes an image memorable? In *CVPR*, 2011.

[22] R. R. Jensen, J. D. Stets, S. Suurmets, J. Clement, and H. Aanæs. Wearable gaze trackers: Mapping visual attention in 3d. In *Scandinavian Conference on Image Analysis*, pp. 66–76. Springer, 2017.

[23] S. Jetley, N. Murray, and E. Vig. End-to-end saliency mapping via probability distribution prediction. In *CVPR*, 2016.

[24] B. John, P. Raiturkar, O. L. Meur, and E. Jain. A benchmark of four methods for generating 360 saliency maps from eye tracking data. In *2018 IEEE International Conference on Artificial Intelligence and Virtual Reality (AIVR)*, pp. 136–139, Dec 2018.

[25] R. Kimsey. Method of active advertising and promotion in an online environment, Mar. 6 2008. US Patent App. 11/837,510.

[26] G. A. Koulieris, G. Drettakis, D. Cunningham, and K. Mania. Gaze prediction using machine learning for dynamic stereo manipulation in games. In *Virtual Reality (VR), 2016 IEEE*, pp. 113–120. IEEE, 2016.

[27] M. Kümmerer, T. S. A. Wallis, and M. Bethge. Deepgaze II: reading fixations from deep features trained on object recognition. *CoRR*, abs/1610.01563, 2016.

[28] L. Kusumoto, D. Sacerdoti, L. Sigler, and S. Sigler. System and method for consumer-selected advertising and branding in interactive media, Mar. 26 2013. US Patent 8,407,086.

[29] D. Lagun and E. Agichtein. Viewser: Enabling large-scale remote user studies of web search examination and interaction. In *Proceedings of the 34th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '11. ACM, 2011.

[30] Y. Li, P. Xu, D. Lagun, and V. Navalpakkam. Towards measuring and inferring user interest from gaze. In *Proceedings of the 26th International Conference on World Wide Web Companion*, pp. 525–533. International World Wide Web Conferences Steering Committee, 2017.

[31] S. P. Liversedge, K. B. Paterson, and M. J. Pickering. Eye movements and measures of reading time. *Eye guidance in reading and scene perception*, 1998.

[32] J. Majewski. Smithsonian guidelines for accessible exhibition design. *Smithsonian Accessibility Program*, 1996.

[33] P. Merrell, E. Schkufza, Z. Li, M. Agrawala, and V. Koltun. Interactive furniture layout using interior design guidelines. In *ACM SIGGRAPH 2011 Papers*, SIGGRAPH '11, pp. 87:1–87:10. ACM, New York, NY, USA, 2011.

[34] L. O'Shea, C. Grimley, and M. Love. *The Interior Design Reference & Specification Book: Everything Interior Designers Need to Know Every Day*. Rockport Publishers, 2013.

[35] C. Ozcinar and A. Smolic. Visual attention in omnidirectional video for virtual reality applications. In *2018 Tenth International Conference on Quality of Multimedia Experience (QoMEX)*, pp. 1–6, May 2018.

[36] X. Pang, Y. Cao, R. W. H. Lau, and A. B. Chan. Directing user attention via visual flow on web designs. *ACM Trans. Graph.*, 35(6):240:1–240:11, Nov. 2016.

[37] S. P. Sargunam, K. R. Moghadam, M. Suhail, and E. D. Ragan. Guided head rotation and amplified head rotation: Evaluating semi-natural travel and viewing techniques in virtual reality. In *Virtual Reality (VR), 2017 IEEE*, pp. 19–28. IEEE, 2017.

[38] T. Sawada, M. Toyoura, and X. Mao. Film comic generation with eye tracking. In *International Conference on Multimedia Modeling*, pp. 467–478. Springer, 2013.

[39] B. Schölkopf, A. J. Smola, R. C. Williamson, and P. L. Bartlett. New support vector algorithms. *Neural computation*, 12(5):1207–1245, 2000.

[40] M. Schwarz and P. Wonka. Procedural design of exterior lighting for buildings with complex constraints. *ACM Trans. Graph.*, 33(5):166:1–166:16, Sept. 2014.

[41] M. Spain and P. Perona. Some objects are more equal than others: Measuring and predicting importance. *ECCV*, 2008.

[42] R. Strohmaier, G. Sprung, A. Nischelwitzer, and S. Schadenbauer. Using visitor-flow visualization to improve visitor experience in museums and exhibitions. In *The Annual Conference of Museums and the Web*, 2015.

[43] Y. Sugano, Y. Matsushita, and Y. Sato. Learning-by-synthesis for appearance-based 3d gaze estimation. In *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pp. 1821–1828. IEEE, 2014.

[44] A. Sutcliffe and A. Namoun. Predicting user attention in complex web pages. *Behaviour & Information Technology*, 31(7):679–695, 2012.

[45] M. van Hamersveld and C. de Bont. *Market research handbook*. Wiley, 2007.

[46] J. Vanian. Google is experimenting with virtual reality advertising. `http://fortune.com/2017/06/28/google-virtual-reality-advertising/`, 2017.

[47] A. Verhulst, J.-M. Normand, C. Lombard, and G. Moreau. A study on the use of an immersive virtual reality store to investigate consumer perceptions and purchase behavior toward non-standard fruits and vegetables. In *Virtual Reality (VR), 2017 IEEE*, pp. 55–63. IEEE, 2017.

[48] L.-F. Yu, S.-K. Yeung, C.-K. Tang, D. Terzopoulos, T. F. Chan, and S. J. Osher. Make it home: Automatic optimization of furniture arrangement. In *ACM SIGGRAPH 2011 Papers*, SIGGRAPH '11, pp. 86:1–86:12. ACM, New York, NY, USA, 2011.

[49] P. Zhang. The effects of animation on information seeking performance on the world wide web: Securing attention or interfering with primary tasks? *Journal of the AIS*, 1(1es):1, 2000.