UEyes: Understanding Visual Saliency across User Interface Types

Yue Jiang
yue.jiang@aalto.fi
Aalto University
Finland

Luis A. Leiva
name.surname@uni.lu
University of Luxembourg
Luxembourg

Paul R. B. Houssel
name.surname@uni.lu
University of Luxembourg
Luxembourg

Hamed R. Tavakoli
hamed.rezazadegan_tavakoli@nokia.com
Nokia Technologies
Finland

Julia Kylmälä
julia.kylmala@aalto.fi
Aalto University
Finland

Antti Oulasvirta
antti.oulasvirta@aalto.fi
Aalto University
Finland

ABSTRACT

While user interfaces (UIs) display elements such as images and text in a grid-based layout, UI types differ significantly in the number of elements and how they are displayed. For example, webpage designs rely heavily on images and text, whereas desktop UIs tend to feature numerous small images. To examine how such differences affect the way users look at UIs, we collected and analyzed a large eye-tracking-based dataset, *UEyes* (62 participants and 1,980 UI screenshots), covering four major UI types: webpage, desktop UI, mobile UI, and poster. We analyze its differences in biases related to such factors as color, location, and gaze direction. We also compare state-of-the-art predictive models and propose improvements for better capturing typical tendencies across UI types. Both the dataset and the models are publicly available.

CCS CONCEPTS

• **Human-centered computing** → **Empirical studies in ubiquitous and mobile computing**; • **Computing methodologies** → *Computer vision*.

KEYWORDS

Human Perception and Cognition; Interaction Design; Computer Vision; Deep Learning; Eye Tracking

ACM Reference Format:
Yue Jiang, Luis A. Leiva, Paul R. B. Houssel, Hamed R. Tavakoli, Julia Kylmälä, and Antti Oulasvirta. 2023. UEyes: Understanding Visual Saliency across User Interface Types. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems (CHI '23), April 23–28, 2023, Hamburg, Germany*. ACM, New York, NY, USA, 21 pages. https://doi.org/10.1145/3544548.3581096

1 INTRODUCTION

What grabs user attention in the setting of looking at user interfaces (UIs) is a long-standing interest in HCI research. Understanding this is essential for designers hoping to guide users' attention, convey critical information, and avoid visual clutter [75, 81]. However, after many years of work on this topic, we still have only a rudimentary sense of how different *types* of UIs differ in visual saliency. For instance, posters often bring together only a few images, while desktop and mobile UIs typically apply more components, structured as widgets. Awareness of how such differences carry over to eye-movement patterns is crucial. The hypothesis underlying the work presented here is that one should expect the users' gaze patterns to reflect the visual features of the UI.

This paper represents a two-pronged approach to advancing the understanding of eye movements that occur with particular UI types. Firstly, we collected and analyzed the *UEyes*, a novel eye-tracking dataset captured by a high-fidelity in-lab eye tracker at a large scale. While previous work used mouse movements or manual annotations as a proxy for eye movements, UEyes offers access to fine-granularity ground-truth data for visual saliency. Our dataset offers multi-duration saliency maps and scanpaths of 62 users who looked at 1,980 different UIs, 495 each from desktop, mobile, webpage, and poster applications. With this paper, we analyze and compare saliency-related tendencies across the UI types, addressing both bottom-up factors related to the visual primitives of the stimulus, such as color bias, and top-down (learned) ones connected with the distribution of features in the dataset, such as location bias and scanpath direction. We present several previously unreported findings illuminating what distinguishes particular UI types.

Secondly, the dataset informed the assessment and improvement of computational models for visual saliency. Given a UI as input, a saliency model can predict saliency maps or scanpaths, simulating how users perceive that UI. These models assist UI designers by predicting where users are likely to fix their gaze within a given design: this enables updating it to emphasize the important areas in the UI better. Such models may help them 'reflow' UI designs and create versons that maintain the desired visual emphases across various screen sizes.

Data-driven approaches require high-quality datasets if they are to employ modern computational models (*e.g.,* based on deep learning) effectively and improve our understanding of visual saliency. There is a plethora of work on saliency modeling, predicting where viewers look [21, 36, 39, 41, 48, 51, 55, 68], and numerous scanpath models, predicting gaze over time [2, 3, 33, 47, 64, 89]. Current approaches all display a limitation, though: they work well only when domain-specific data are available. Yet, datasets thus far have been relatively small (*e.g.,* MASSVIS [11] and iSUN [96]) and often limited to specific types of designs (*e.g.,* only mobile UIs [53]). In contrast,

CHI '23, April 23–28, 2023, Hamburg, Germany

ACM ISBN 978-1-4503-9421-5/23/04.
https://doi.org/10.1145/3544548.3581096

our UEyes dataset is composed of high-quality eye-tracking data for various UI types – webpages, mobile UIs, desktop UIs, and posters – so is more generalizable and valuable for a broader range of applications. In addition, although Leiva et al. [53] proposed analysis for mobile UIs, no prior research that we know of has analyzed biases in saliency maps (*e.g.,* location bias) and in scanpaths (*e.g.,* saccade angle) for comparison across UI types. We aimed to fill this gap by systematically analyzing and comparing eye-tracking data across several UI types.

Furthermore, the UEyes dataset enables dedicated models to predict visual saliency and scanpaths between distinct UI types. A multi-type dataset is important because accuracy decreases significantly when tested on UI types not included in the training data. Designers could use these models to inform a better user experience for interfaces. With visual saliency models, designers can improve their designs by means of well-grounded conclusions about how users are likely to view their UIs [15]. Predictive models for scanpaths are unlike saliency maps in that they retain information about the order of fixations and their temporal dynamics. It is important that the applications keep this information available. For example, these models allow designers to understand visual flows and adjust their designs to encourage users to view the UI elements in the desired order [69].

The prior project most relevant for our work proposed a crowdsouced dataset (Imp1k) and a Unified Model of Saliency and Importance (UMSI) trained on images from various design classes: webpages, movie posters, mobile UIs, infographics, and advertisements [29]. It created a generalizable model for visual importance that performed well for various design types. However, it did not further address differences in how users view those particular types. Our collection and classification of images accomplished that aim by focusing on common UI types and introducing a systematic eye-tracking analysis and comparison across the respective types. Unlike the UMSI researchers, we collected real-time eye-tracking data via an eye tracker. Although crowdsourcing approaches enable amassing large datasets (*e.g.,* Imp1k and SALICON [39]) via proxies for eye-tracking data, such as cursor- or webcam-based methods, they cannot simulate the results collected by actual eye trackers. Webcam-based approaches suffer from low accuracy, while cursor-based methods reflect cognitive processes different from those behind eye movements [83].

In sum, this paper makes three contributions:

(1) We present the first analysis and comparison of eye movements across commonly used UI types. We report differences related to location bias, color bias, saccade angle and amplitude, and visited vs. revisited elements.
(2) We compare the performance of several predictive models for saliency maps and scanpaths across the UI types. In light of our data, we present improvements to existing models, such as changes in loss terms, training strategies, and modeling features (*e.g.,* "inhibition of return").
(3) We release the largest in-lab eye-tracking dataset (from 62 participants and 1,980 UI screenshots), with associated metadata and eye-tracking logs, grouped into webpages, desktop UIs, mobile UIs, and posters.

2 RELATED WORK

Predicting where people look is paradigmatically more ambiguous than such typical tasks related to computer vision as image segmentation [63] and object detection [42]. For a starting point, we hypothesized that significant differences should be observable among UI types for the same reasons that considerable differences have been reported between scenes and between individuals. Differences among individuals and stimulus types can be attributed both to physiologically determined bottom-up factors and to learned top-down features [100]. On one hand, the biological basis for bottom-up saliency is rooted in the parallel processing of retinal input in the visual cortex [85]. Bottom-up features are constituted by a few physiologically determined visual primitives – size, color, shape, orientation, and motion [54, 92]. Objects that in the given context stand out in one or more of these respects tend to attract attention. For instance, larger objects, which also have more stimulus energy, have greater saliency too. Top-down factors, on the other hand, bring in task-linked goals and expectations. Expectations form through repeated exposure to instances of a particular type of stimulus [78].

2.1 Visual Saliency in Natural Scenes

Previous work on visual saliency outside the human-computer interaction (HCI) domain has focused on non-UI stimuli and natural scenes. Consequently, viewing patterns reported for them may not hold for UIs. Research looking at the saliency of natural scenes has found several replicated effects, or (viewing) biases, which we revisit in this paper:

Center bias: Researchers have reported a bias toward looking at the center of the screen when viewing natural scenes [35, 65]. The effect has been replicated with artificial media, especially video [59], text [73], and single objects [65]. Whether it is present for UIs is unclear, since much of their most informative elements lie in the upper half of the display.

Horizontal bias: In observation of natural images that feature objects, fixation paths tend to be distributed more horizontally than vertically [65, 66]. Again, UIs differ from natural scenes in that they arrange the information vertically rather more than horizontally. Therefore, we might expect to see the effect weaken.

Color bias: Color brightness and contrast are among the primary features driving bottom-up saliency [27, 32]. Visual designs such as websites and mobile UIs typically contain colorful icons and images perceived as highly salient. Therefore, we would expect this bias to remain.

2.2 Visual Saliency in UI Designs

The HCI field's research into visual saliency has looked at either eye-movement data limited to a single UI type (*e.g.,* mobile UIs [53]) or proxy constructs that, while correlated with eye movements, are not ideal for saliency modeling. The *visual impression* is the reported visual appeal of a UI's graphical regions or objects as measured via rating scales; results have been reported for both desktop [56] and mobile interfaces [61]. In contrast, visual saliency is a construct related to the control of visual attention, not self-reports on what is felt to be important.

A concept closely related to saliency is that of *visual importance*. Bylinskii et al. [15] extended a pretrained neural network [79] for predicting which regions in a graphic design are felt to be more critical. Their work measured importance by utilizing cursor exploration of a blurred page. However, a "poor man's eye tracker" [19], which involves an element of reflective judgment of importance, is not a good proxy for gauging visual saliency [83]. Finally, research on *visual clutter* is directly motivated by theories of saliency. Work by Rosenholtz [75] showed how one might exploit models of visual saliency to compute indices for how cluttered users perceive a display to be.

2.3 Visual Saliency Datasets

Many existing visual saliency datasets cover only specific types of designs or feature a relatively small number of saliency results. Most of them encompass one specific type of visual design alone, with data collected from a set of participants in a context limited to visualization (*e.g.,* MASSVIS [11]), indoor and outdoor natural images (*e.g.,* iSUN [96], SALICON [39], MIT1003 [43], MIT300 [41], and NUSEF [72]), mobile user interfaces [53]), visual flows in viewing of comics [16], webpages [80], posters [67], etc. While CAT2000 [9] comprises 20 categories, all of them are classes of natural images, with additional augmented natural images (including the non-photorealistic rendering of natural images, such as sketches and cartoons, and noisy natural images, such as low-resolution scenes and Gaussian-noised images). UEyes, the dataset we collected for this work, contains eye-tracking data for four common categories of UIs and extensive variety of images, with focus on visual designs.

Although prior work has explored the power of crowdsourced collection of saliency-related data, (*e.g.,* Imp1k [29] and SALICON [39]), crowdsourcing precludes the use of high-fidelity in-lab eye trackers. As noted above, the proxy sensors, such as cursor movements or webcams, present issues of their own. For instance, accuracy issues with webcam-based methods [96] may arise during facial landmark tracking, eye region extraction, and calibration with the webcam. Cursor-based approaches [4, 39, 44, 45], in turn, reflect slower, more deliberative cognitive processes than eye movements do.

2.4 Computational Visual Saliency Models

Given a stimulus image, a computational model of visual attention predicts a saliency map [7] or a scanpath showing the order in which eye fixations are expected to occur over the image area [53]. Stimulus-driven saliency models are computed via visual primitives [8, 10]. They work well for first-time exposure, for things the user has not seen before [31, 37]. In contrast, task-driven models gauge a user's familiarity [78], which is affected by expectations, location memory, and search strategies. Data-driven modeling makes predictions based on image features, and the architectural assumptions allow it to capture domain-specific viewing tendencies [53] better than other sorts of modeling.

Computational modeling of saliency has attracted computer vision and HCI researchers' interest since the work of Itti and Koch [37]. More recent research on saliency maps has explored emerging types of deep learning architecture. An early approach applied an ensemble of deep networks (eDN) [87], using deep nets as extractors for hierarchical features and combining the outputs with a support vector machine. DeepGaze I [48] followed the same logic, considering a sparsification loss term, center bias, and a smoothing kernel. ML-Net [20] fine-tuned the features for saliency prediction to improve on the previous two models. Then, the Saliency Attentive Model (SAM) [21] added temporal tuning by employing progressive formation of saliency with ConvLSTM blocks to process features.

To consider the evolution of saliency maps over time, Fosco et al. [30] proposed a multi-duration saliency factor, predicting saliency with distinct durations. Generative adversarial networks (GANs) also reached a good approximation of saliency distributions [17, 68]. Some models improved the prediction performance by exploiting contextual information and encoding the similarity between images [46, 58, 71]. SalFBNet [25] is especially noteworthy for employing a recursive feedback architecture feeding later computation blocks back to an earlier stage in the computations; it proved useful in recognition tasks when compared to purely feedforward networks [97]. All these advances in technique notwithstanding, similar results could be achieved by increasing the networks' capacity. For example, EML-NET [38] has been applied for multi-branch prediction at the decoding stage. UniSal [26] unified the prediction of saliency between image and video stimuli. DeepGazeIIE [55] employed a combination of multiple backbones.

Scanpath prediction is a more challenging problem, since information on the order of fixations must be retained. Itti and Koch [37] implemented an inhibition of return (IOR) mechanism to generate a sequence of fixations by means of the computed saliency maps. This work inspired a group of techniques that utilize a saliency map for scanpath generation. For example, Tavakoli et al. [74] proposed a joint sampling mechanism to estimate the saliency and gaze points. Wloka et al. [91] improved on the Itti saccade-generation system by considering the high-level saliency estimated with deep nets and a peripheral conspicuity map obtained via low-level approaches to saliency. In other work, Chen and Sun [18] introduced an advanced architecture to learn the inhibition of return maps from data. Xia et al. [94] estimated joint saliency and fixation location with an auto-encoder in a framework mimicking [74].

Other recent work has developed scanpath models that can generate a sequence of fixation locations. For example, Verma and Sen [86] employed a recurrent architecture to generate a sequence of fixations in a grid-based representation, and PathGAN [3] uses GAN-based training to estimate a fixation sequence with location and duration. Our project considered such prior work by comparing several well-known predictive models that use saliency maps and scanpaths, assessing their ability to model observed differences among UI types.

3 THE DATASET: UEYES

The UEyes dataset is composed of both the 1,980 UI screenshots and the associated metadata and eye-tracking logs from 62 viewers, collected in a laboratory by means of a modern eye tracker. This dataset contains 495 screenshots from each of the following UI types:

Webpage: We collected 494 webpage images from the Alexa 500 dataset [90], 1,507 images from the Visual Complexity and Aesthetics dataset [62], and 200 images from the Imp1k dataset [29]. We extended the breadth of the webpage image set by capturing 103 additional webpage screenshots.

Desktop UI: The desktop UI image set contains the Waltteri Github desktop UI dataset [23], representing 51 desktop UIs, and an additional 303 desktop UI images collected in line with the criteria presented below.

Mobile UI: We extracted a sample of 1,761 images from among the 46,064 mobile UI images from the RICO dataset [24]. We extended the set with 42 further mobile UI images.

Poster: The poster image set contains 200 ads and 198 infographics from the Imp1k dataset [29], along with 103 additional posters we collected.

The images additional to the ones from pre-existing dataset were chosen either for breadth of representation (being substrantially different from the others) or because of their widespead use in day-to-day life. The additional mobile UI images we collected besides the ones in existing datasets are in the categories of school apps, library apps, music apps, and setting pages. This was to ensure a diversity-rich and representative dataset. Also, the addition of more desktop UI images supported a balanced final dataset. Images containing pornography were filtered out, and then all images of each type were pooled together and sampled randomly to create "image blocks" for user assessment (55 blocks in all for the study). Each block included nine images representing each UI type, for 36 images per block.

For the data collection process, the screen angle was adjusted for each participant to mimic the user-specific typical viewing experience. Participants sat approximately 50–65 cm from the screen, and the same visual angle was used for all UI types, even the mobile UIs, to ensure a fair comparison. This allowed for consistent data collection and analysis across the UI types: consistent presentation across the types guarantees that the tracking technology's accuracy limits do not disproportionately affect the mobile UI results.

3.1 Participants

We recruited 66 participants (23 male and 43 female) via mailing lists and social-media-based promotion. The average age was 27.25 (SD = 7.26). Participants had normal vision (43) or, from wearing glasses (18) or contact lenses (5), corrected-to-normal vision. No participant was colorblind. We dropped four users' gaze data for reason of inaccurate eye-tracking calibration. The study took one hour for each user, who received 30 Euros in compensation.

3.2 Experimental Design

From the pool of 55 blocks, our system randomly selected nine blocks for each user (for 36 images in all, as described above). Hence, each block included nine images for each UI type. Within each block, the images were presented in a randomized order.

3.3 Apparatus

The images were shown on a desktop monitor (HP Compaq LA2405wg, 24 inches). The monitor's dimensions were 32.5×52 cm and its resolution was 1920×1200 px. We used a Gazepoint GP3 eye tracker with a sampling rate of 60 Hz to collect high-quality gaze data. The eye tracker was placed under the screen and tilted upward. Its angle was adjusted to suit the individual participant. With the participants seated approximately 50–65 cm from the tracker, the eye-tracking software (Gazepoint Control) indicated a desirable distance.

3.4 Procedure

The procedure began with calibrating the tracker via Gazepoint Control's nine-point calibration and testing on the calibration test screen. After that calibration, the participant was shown three images, of differently sized grids, and instructed to look at the corners of the grids, starting from the top left and moving clockwise. This served quality control in the post-processing stage. Each participant then completed nine blocks as defined above, with self-managed breaks. The participant looked at each UI image presented, for seven seconds, and was asked to examine the images as if in a corresponding real-world situation. Just as in other bottom-up saliency studies, no specific task was assigned. After the last block of UI images, the participant filled out a demographics questionnaire.

3.5 Data Processing

We double-checked the collected data to guarantee the dataset's quality, and we removed any user data exhibiting inaccurate calibration or duplicate results. Accordingly, the final dataset contains 94.86% of the raw data collected. Fixations beyond image boundaries (6.8% of the fixations) were not considered for analysis. We describe the UEyes dataset in detail in *Supplementary Materials*.

4 FINDINGS

With the discussion below, we examine the data related to location bias, color bias, saccade angle and amplitude, and visited vs. revisited elements, across all UI types.

4.1 Effect of Location

Figure 2 shows the location bias for each of the UI types, and Figure 3 displays the corresponding distribution of fixations by quadrant. We computed the location bias by normalizing the saliency distribution relative to the individual UI image's size and then aggregating all the UI saliency results associated with each UI type. Overall, in contrast against the recognized center bias with natural images [13], we noticed that the upper-left quadrant of all UI types tends to attract more fixations than the other quadrants. This general result indicates that participants paid more attention to the upper-left portion of the UIs. For the webpages, mobile UIs, and posters, fixations are spread across the entire upper-left region, while there are two bands of salient regions in desktop UIs: one right above the center of the UI and the other near the upper left-hand corner. The most salient area of webpages is around the center-right section of the upper-left quadrant, while that quadrant's uppermost portion attracts the most attention in the mobile UI condition. Desktop UIs and posters deviate from this pattern, with the most salient area appearing just above the center of the desktop UIs and posters.

An omnibus test revealed statistically significant differences in the average number of fixations per user for the visual content specific to each quadrant (where Q1 = top right, Q2 = top left, Q3

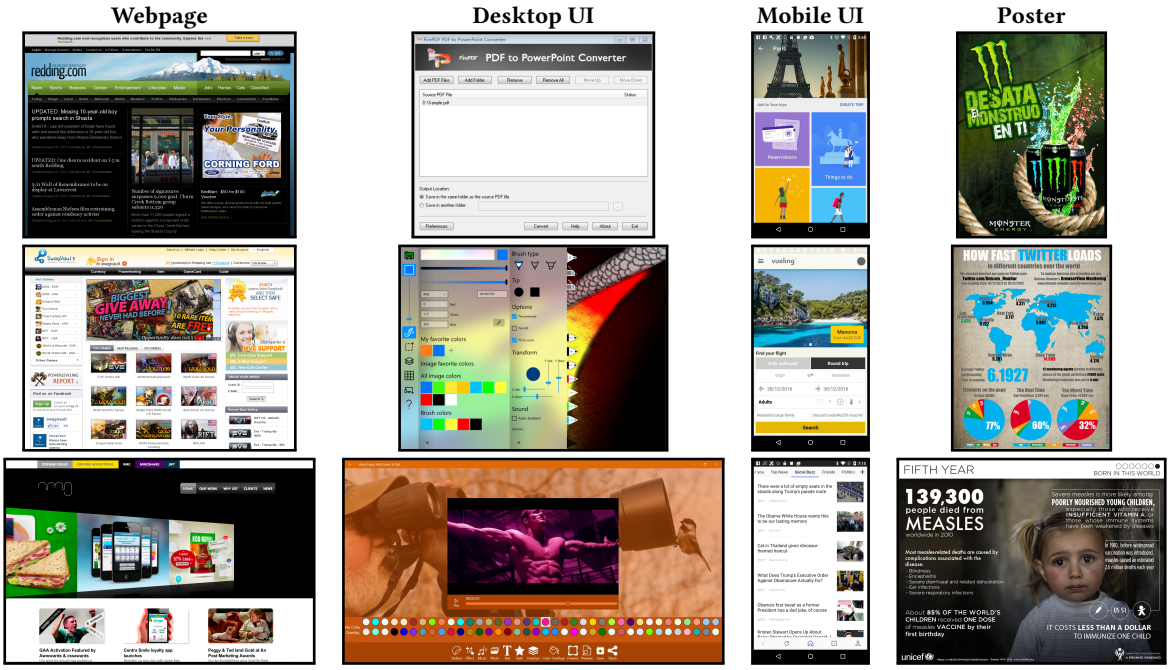
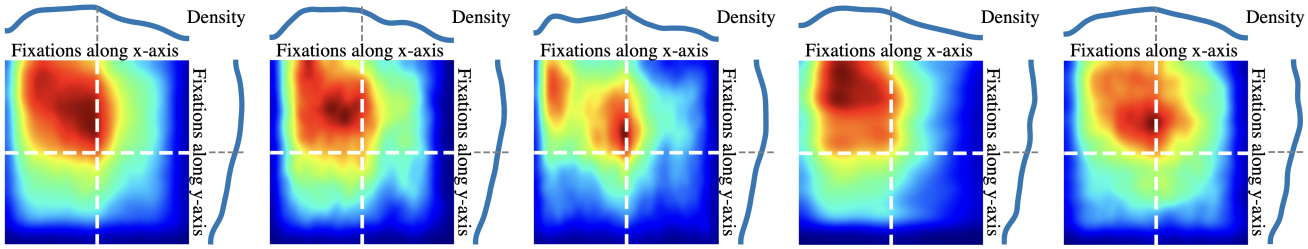


Figure 1: Examples of user interfaces in the UEyes dataset. The full dataset contains 495 images of each UI type: webpage, desktop UI, mobile UI, and poster.



(a) All UI types (b) Webpage (c) Desktop UI (d) Mobile UI (e) Poster

Figure 2: Location bias – the distribution of fixations over normalized screens. In contrast against the center bias of natural images, fixations in user-interface settings are mostly in the upper left.

= bottom left, and Q4 = bottom right). For example, in the general (all-UI-type) condition, $\chi^2(3) = 183.930, p < .0001$. Similar results were obtained for each specific type of UI.

We then ran Bonferroni-Holm corrected pairwise comparisons in post-hoc testing and found that the difference between Q1 and Q2 was statistically significant in all cases ($p < .001$). The Q1 vs. Q3 difference and the Q1 vs. Q4 one were statistically significant when users viewed the images for three seconds or longer ($p < .001$), Also, the difference between Q2 and Q3 and that between Q2 and Q4 was statistically significant in all cases ($p < .001$). Finally, the Q3 vs. Q4 difference was significant when the viewing time was three seconds or longer ($p = .018$).

4.2 Effect of Color

We show color bias across different UI types in Figure 4. The top color bar shows the 16 most prevalent colors in the original UI images for different UI types. The other color bars rank the top 16 colors by the number of fixations on those colors, sorted by frequency. We computed the 16 most prevalent colors using k-means clustering, therefore similar colors are merged together. Figure 4 characterizes the color bias across the UI types examined. The uppermost bar in each pane shows the 16 most prevalent colors in the original UI images, for the relevant UI type. The other bars present those top 16 colors ordered by the frequency of fixations on them. We computed the 16 most prevalent colors by using k-means clustering; therefore, similar colors are merged. Figure 5 compares the colors displayed ("All colors" in the plots) with those colors

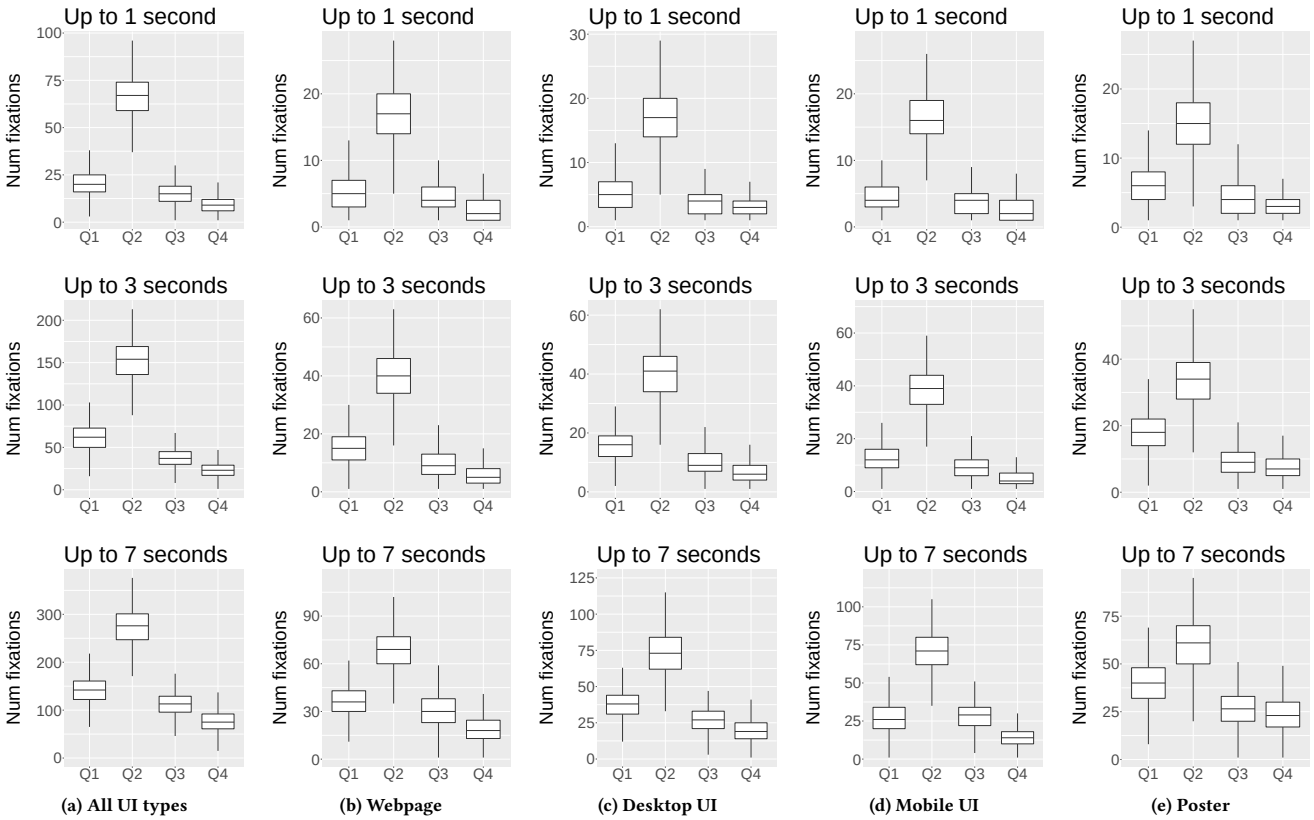


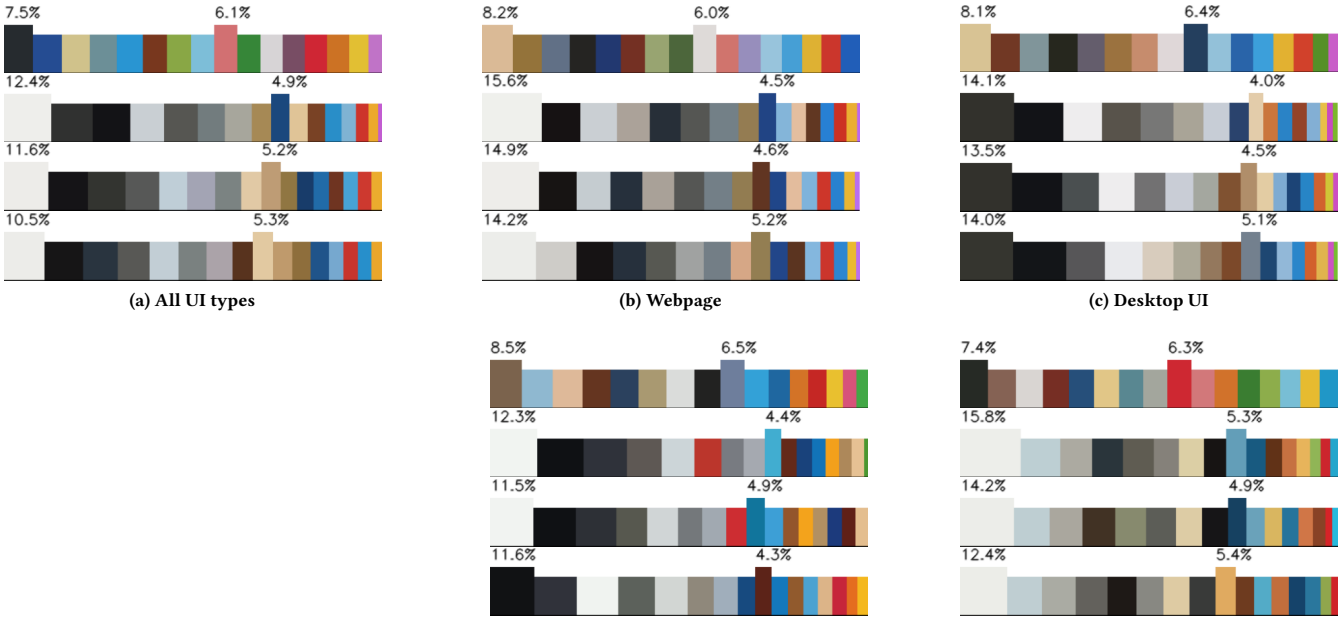
Figure 3: Location bias – fixations' distribution by quadrant. The upper-left quadrant tends to attract more fixations than other quadrants, across all UI types.

receiving fixations. This comparison of brightness reveals that, on average, brighter colors attract more attention than darker ones. Designs for webpages, desktop UIs, and mobile UIs seem to draw greater attention to more brightly colored areas relative to the color mix displayed. Posters constitute the only exception: the average brightness value of the colors where fixations occur is lower than that of the colors displayed. However, the single color at which participants look most often in posters is still a light one. Although desktop UIs' fixation-receiving colors are brighter, on average, than the colors shown overall, the three colors with the largest numbers of fixation points in these UIs are dark ones. To investigate further whether a reliable effect exists, we computed the pixel brightness values by using sRGB Luma coefficients (ITU Rec. 709) [6], which reflect the corresponding standard chromaticities, and compared distributions between fixation and non-fixation brightness values. Bartlett's test of homogeneity of variances was statistically significant neither for all UIs combined ($\chi^2(3) = 1.003, p = .8004$) nor for any UI type individually ($\chi^2(3) \leq 0.832, p \geq .8416$). Therefore, we conclude that color does not significantly affect visual saliency.

4.3 Saccade Angle and Amplitude

Saccade angle and amplitude reveal the tendency and speed of eye movements. Such data can facilitate optimizing UI elements' placement and the flow of information in a UI. By understanding these metrics, designers can align their designs well with the natural gaze behavior of users, thereby potentially promoting a better user experience. Figure 6 shows the distributions for the direction and distance between two consecutive fixation points, represented by the saccade angle and amplitude in the polar-coordinate system. We can see that, overall, user gaze moved mainly towards the right or bottom portion of the UIs. However, UI types do differ markedly in this respect. Users showed a greater preference for left-to-right movement in the webpage condition than with other UI types. Similarly, users tended to scan posters from left to right, with a small number of downward movements, but they showed greater variety in the distances by which the gaze moved rightward. In contrast, users looked both from left to right and from top to bottom when viewing desktop UIs and mobile UIs. The distances in moves toward the right are larger than those toward the bottom in the desktop condition. They remained in about the same range for mobile UI designs.

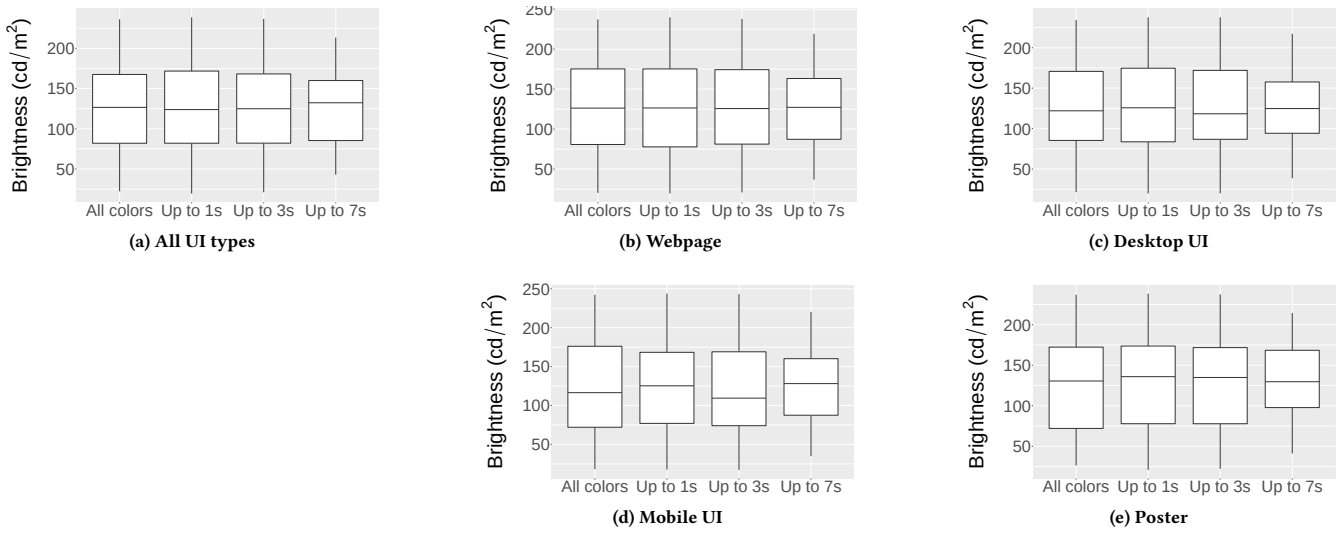
A Kruskal-Wallis chi-squared test showed statistical significance for all UI types (*e.g.*, $\chi^2(3) = 484.41$, $p < .0001$ overall), so we ran pairwise comparisons (Bonferroni-Holm corrected) as post-hoc tests, finding that all directions were significantly different from



(a) All UI types (b) Webpage (c) Desktop UI

(d) Mobile UI (e) Poster

Figure 4: Color bias – the 16 most prevalent colors in UIs (top row) and the 16 colors fixated upon most, by frequency, for fixations lasting up to 1 s (second row), up to 3 s (third row), and up to 7 s (bottom row).



(a) All UI types (b) Webpage (c) Desktop UI

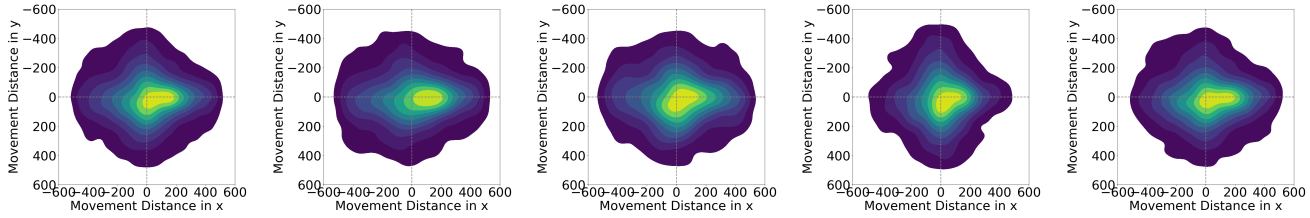
(d) Mobile UI (e) Poster

Figure 5: Color-brightness bias plots comparing the brightness of all the colors displayed and fixated upon. Overall, brighter colors tend to attract slightly more attention than darker ones, especially for short time spans.

each other for all UI types; the rightward direction is the most frequent, followed by motion toward the left, bottom, and top.

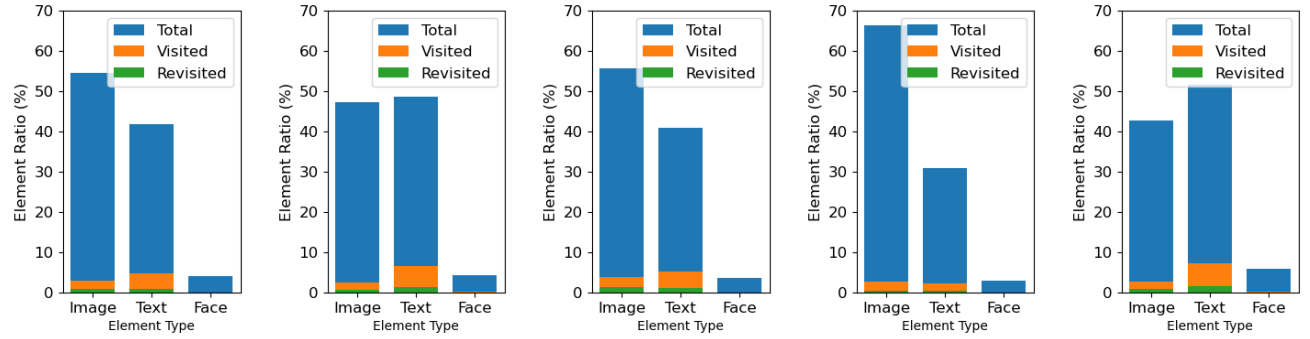
4.4 Visited vs. Revisited Elements

We segmented the UIs and classified the UI elements into three categories – image, text, and face – by extending the functionality of the UIED model [95], a model for detecting images and texts on UIs. Then, we counted the number of elements in each category that were visited (fixated upon) and revisited (fixated upon again). Once visited, an element is considered revisited if it receives at least three fixation points and the previous fixation was on another element. The results are shown in Figure 7. We observed that text



(a) All UI types (b) Webpage (c) Desktop UI (d) Mobile UI (e) Poster

Figure 6: Analyzing saccade bias reveals the direction and distance between consecutive fixation points. Gaze directions lead mainly toward the right or bottom portion of the UIs, with the distances being larger near the right – users prefer moving the gaze from left to right (with larger motions) and from top to bottom.



(a) All UI types (b) Webpage (c) Desktop UI (d) Mobile UI (e) Poster

Figure 7: Visit vs. revisit bias analysis showing the ratios of visited to revisited elements in three element categories. Text elements are more likely than images to be visited and to be revisited.

elements have a higher fixation probability in our data than images. While the dataset's desktop UIs feature many small images (such as icons), which are more prominent than text, the opposite was visible in posters: they had large images, typically in a small quantity. Webpages have about the same number of image and text elements. It is worth noting that mobile UIs exhibited lower visit and revisit ratios both than other UI types, reflecting mobile UIs' reduced opportunities for users returning to the same content later.

We found statistically significant differences in visit and in revisit ratios between element types (image, text, and face) for all UI types. For example, in comparison of the visit ratios for the overall condition, $\chi^2(2) = 9.295$, $p < .01$. Post-hoc pairwise comparisons (Bonferroni-Holm corrected) revealed statistical significance for all UI types compared. We conclude that text attracts fixations the most, followed by images and then faces.

4.5 Summary

We can summarize our results thus: The upper-left quadrant tends to attract the most fixations, while brighter colors do not attract significantly more fixations than less bright colors. As users gaze at UIs, their saccades move mainly from left to right and from top to bottom. Participants in our experiment tended to spend more time looking at text elements than images, which accounts for the saccade directions' left-to-right tendency. Overall, our findings related to mobile UIs are consistent with the results of Leiva et al. [13]. When introducing further analysis metrics, we found biases specific to each UI type. The following characteristics and differences emerged, recapped here by UI type:

Webpage: Participants preferred to scan more from left to right when looking at webpages, with larger distances between consecutive fixations than they showed with other UI types.

Desktop UI: Rather than fixations being spread over the top-left quadrant, the salient areas of desktop UIs are separated into two areas: right above the center and around the top-left corner.

Mobile UI: Mobile UIs exhibit lower visit and revisit ratios than other UI types. This indicates that users tend to focus more on a few elements of the UI (the most attractive ones) while ignoring others and that there is less likelihood of going back to look at the same elements.

Poster: In comparison to desktop UIs and mobile UIs, participants demonstrated a much stronger intention to scan from left to right, with only a small proportion of saccades being directed from top to bottom. The distances between consecutive fixation points show more significant variation here than with other UI types.

5 ASSESSING SALIENCY MAP MODELS

With the backdrop of the differences identified between UI types, we conducted a comparison among data-driven predictive models for saliency maps. We considered the state-of-the-art traditional optimization-based model (GBVS) and data-driven models (the SAM and UMSI) alongside improved versions that we developed ourselves, SAM++ and UMSI++.

Graph-Based Visual Saliency (GBVS) [34]. GBVS is a bottom-up saliency map model for detecting informative features on the basis of the entire image. It employs the saliency-based visual attention model proposed by Itti and Koch [37] to extract visual features as computed via linear center-surround operations with Gaussian pyramids for intensity, color, and orientation. It then forms graph-based activation maps from visual features and normalizes them to highlight conspicuity. The global visual feature extraction and graph-based activation maps enable the model to capture saliency maps at the global level, which is more efficient than prior approaches relying on local information.

Saliency Attentive Model (SAM) [21, 22]. SAM incorporates an attentive convolutional long-short term memory (Attentive ConvLSTM) saliency map model to focus on distinct spatial location features to enhance sequential predictions. The model iteratively and progressively refines the predicted saliency map results via the LSTM architecture. The SAM learns a set of prior maps generated with Gaussian functions to learn saliency priors, such as the center bias typical of human eye fixations, thereby obtaining improved feature-extraction capabilities without needing hand-crafted prior information.

UMSI [29]. UMSI is a unified model of saliency and importance trained on images from several design classes, including posters, infographics, mobile UIs, and natural images. It uses an encoder-decoder architecture and aggregates image information at multiple scales to predict visual importance in the input graphic designs. The UMSI employs an automatic classification module for the input graphic designs, to better capture the saliency patterns with class-specific information. It was trained on a dataset for visual importance from cursor-based crowdsourced data. Again, while the cursor is a good proxy for eye-tracking, it cannot properly simulate the results captured via data from eye trackers.

UMSI++ and SAM++ (Ours). UMSI++ and SAM++ are variants we created by employing new loss terms and a two-step training process. The main module of the original UMSI model was trained with KL-divergence [40] and Cross-Correlation [52] losses with coefficients 10 and -3. The output of the UMSI model is the flipped saliency maps requiring post-processing via black-to-white inversion. Our UMSI++ model employs an end-to-end joint training process that entails refining the model via multiple loss terms. Over the first 10 epochs of training, the model approaches the ground-truth saliency maps by using the Mean Squared Error (MSE) loss between the predicted and the ground-truth saliency maps. This helps the model accurately predict the saliency maps. For the remaining epochs, the model is trained with a combination of loss terms, including the KL-divergence and Cross-Correlation loss terms [52] used in the UMSI, alongside two additional loss terms: the Normalized Scanpath Saliency (NSS) loss and the Similarity loss. The NSS loss quantifies the average normalized saliency at fixation points, while the Similarity loss measures the intersection between the predicted and the ground-truth saliency maps. These loss terms help the model better capture fixations and improve its overall performance. Both KL-divergence and Cross-Correlation are distribution-based: they focus on the continuous distributions of the saliency maps, rather than on individual points or locations. In contrast, NSS and Similarity are location-based in that they focus on the locations of fixation points in the saliency maps. Together, these loss terms have been shown to perform well in predicting fixation points, and they can help the model better capture eye fixations [12, 99]. Computation details are given in *Supplementary Materials*. The training takes about an hour on one NVIDIA GeForce RTX 2080Ti GPU. For comparison, we apply the same training pipeline and loss terms to the SAM architecture to get the result for the SAM++ model.

5.1 Evaluation Metrics

We evaluated the accuracy by means of six widely applied metrics.

Area under ROC Curve (AUC). AUC is the most commonly used metric for saliency map performance. It evaluates the saliency map as a binary classifier of fixation points at various thresholds. The Receiver Operating Characteristic Curve (ROC Curve) shows the rates of the actual positive points and the false positive ones at multiple discrimination threshold values. AUC is defined as the area under such a curve measuring the true and false positive rates under the binary classifier, which one can compute by taking the integral of the area under the ROC curve in practice. AUC-Judd [14, 43] is a variation of AUC. The true positive rate is defined as the ratio of the number of true positive points to the number of ground-truth fixation points above various threshold values, while the false positive rate is that of the number of false positive points to the total number of non-fixation pixels.

Normalized Scanpath Saliency (NSS) [70]. NSS is the average normalized saliency at fixation points. Relative to the AUC metric, NSS is more sensitive in detecting false positive points. The AUC score can be high even when there are many false positive points, given a large number of true positive points – low-valued false positive points do not affect the AUC score. However, all false positive points decrease the normalized saliency value. Thus the NSS score penalizes all the false positive points.

Information Gain (IG) [49, 50]. IG is used for measuring saliency results beyond systematic bias.

Similarity (SIM) [76, 82]. SIM refers to the intersection between the predicted and the ground-truth saliency maps, thereby indicating the overlapping of the two maps. It is defined as the sum of the minimum value of the normalized predicted saliency map and of the normalized ground-truth map. The similarity score is lower for sparse maps. It is sensitive to failed detection of saliency points: the absence of saliency values points to zero similarity, hence reducing the similarity score.

Pearson's Correlation Coefficient (CC) [52]. CC is employed for evaluating the correlation or dependence between the predicted and the ground-truth saliency maps.

Kullback-Leibler (KL) Divergence [40]. KL Divergence quantifies the difference between the distributions of the saliency map prediction and the ground truth, while the other metrics listed here measure the similarity.

Computation details are provided in *Supplementary Materials*. The various metrics differ in their sensitivity to false positives or false negatives, what is measured, and the category of metric involved, thus:

Sensitivity: All these metrics are sensitive to false negatives, with the KL, IG, and SIM significantly penalizing false negatives, especially when the predicted values are close to zero. The normalization step of NSS increases the penalty for detecting false positives and thus makes it more sensitive to false positives than other metrics. The CC is, by definition, a symmetric metric, so it shows equal sensitivity to false positives and false negatives. The AUC score is insensitive to false positives – it can be high even if the resulting saliency maps have many true positives.

Measurement: The KL measurement assesses dissimilarity while the other metrics gauge similarity. Accordingly, better models have lower KL scores but higher scores for other metrics.

Metric category: The location-based metrics (AUC, NSS, and IG) evaluate models in terms of fixation points, while distribution-based ones (SIM, CC, and KL) compute evaluation based on saliency maps as the continuous distribution.

5.2 Results

To set a benchmark for saliency maps' prediction, we compared the computational saliency map models qualitatively and quantitatively and judged the predicted location bias. We used the dataset's first 52 image blocks (1,872 images) as the training data and the remaining three blocks of images (108 images) for testing. All the results shown here come from evaluation with the test data.

5.2.1 Qualitative Evaluation. We present the qualitative comparison of the various models by UI type in Figure 8. For all the models, false positive errors constitute the main kind of error in the results. All of them can capture informative areas such as images and text elements, but not all of these truly attract the user's attention, and sometimes, only a small part of an image is considered salient. Therefore, it is generally challenging for predictive models to distinguish between informative areas and salient areas. Both GBVS and the pretrained UMSI typically capture all image and text areas, which leads to high false positive error levels. Models trained on UEyes achieve better results than the pretrained models. Our improved model UMSI++ generates the saliency maps that most closely approach the ground-truth fixations, relative to the other models and across all the UI types.

5.2.2 Quantitative Comparison across UI Types. We begin by addressing the importance of training on multiple types of UIs, because training with only one type leads to accuracy reductions with other types. We trained our UMSI++ model on either mobile UI or webpage data from UEyes, respectively, and testing on all UI types used the same test set. Accuracy in predictions for other UI types (those different from what was seen during training) dropped significantly. For example, when the model was trained on mobile UIs, its accuracy fell from 0.899 to 0.844 when it was tested instead on webpages, from 0.890 to 0.803 for desktop UIs, and from 0.924 to 0.849 for posters. Similarly, when the model was trained on webpages, its accuracy decreased from 0.905 to 0.832 for mobile UIs, from 0.890 to 0.813 for desktop UIs, and from 0.924 to 0.848 for posters. How people perceive visual hierarchies and look at UIs when viewing any given type of UIs could not truly be generalized to other UI types.

Quantitative comparison of the models as evaluated via the metrics detailed in Section 5.1 attests that training on UEyes provides both the SAM and the UMSI models with higher accuracy and stronger generalization ability. Furthermore, our improved model, UMSI++, outperforms the state-of-the-art models by most metrics, as Table 1 indicates. Since AUC is a standard evaluation metric with a range of 0 to 1 for saliency map prediction (where larger values indicate higher accuracy), it lets us quantitatively evaluate and compare the models across UI types graphically in the manner shown in Figure 9. The pretrained SAM model performs better than the pretrained UMSI. However, after training on UEyes, the two perform similarly for all the UI types. By introducing new loss terms, UMSI++ achieved the best performance across all the UI types, while SAM++ does did not show greater accuracy than the original SAM trained on UEyes. For both SAM and UMSI architectures, the predictions for desktop UIs were the least accurate for every UI type. This is consistent with our observations from the qualitative results.

5.2.3 Predicted Location Bias. Figure 10 presents a visualization of the location bias of the saliency maps predicted by the various models considered. All models except GBVS can capture the upper-left location bias identified for UIs. The models trained on our UEyes dataset can capture that location bias more accurately than the pretrained ones. After training with UEyes, the SAM, SAM++, and UMSI models achieved similar results for saliency location bias. Our improved model displays the greatest similarity to the ground truth for location bias. It is clearly evident from the visualization that all the models over-capture the salient areas and produce many false positive errors, which aligns with what we found in the qualitative comparison. Saliency is tricky to detect for mapping predictions in contexts of webpages and (especially) desktop UIs. Salient areas are spread more sparsely in the upper-left quadrant of webpages than in other UI types. The salient portions of desktop UIs are separated into the two sub-areas mentioned earlier on, one right above the center line and the other near the top left. It is far more challenging for the models to simulate sparser areas. Still, the models trained on UEyes capture such sparse salient areas better, while other models can only capture the entire areas, with many more false positives. All of the models except GBVS can capture the most salient parts of mobile UI designs (the upper-left quadrant) and posters (right above the center) well. The ones trained on UEyes are similarly accurate in their location bias results across webpages, desktop UIs, and mobile UIs; however, UMSI++ reveals the location bias

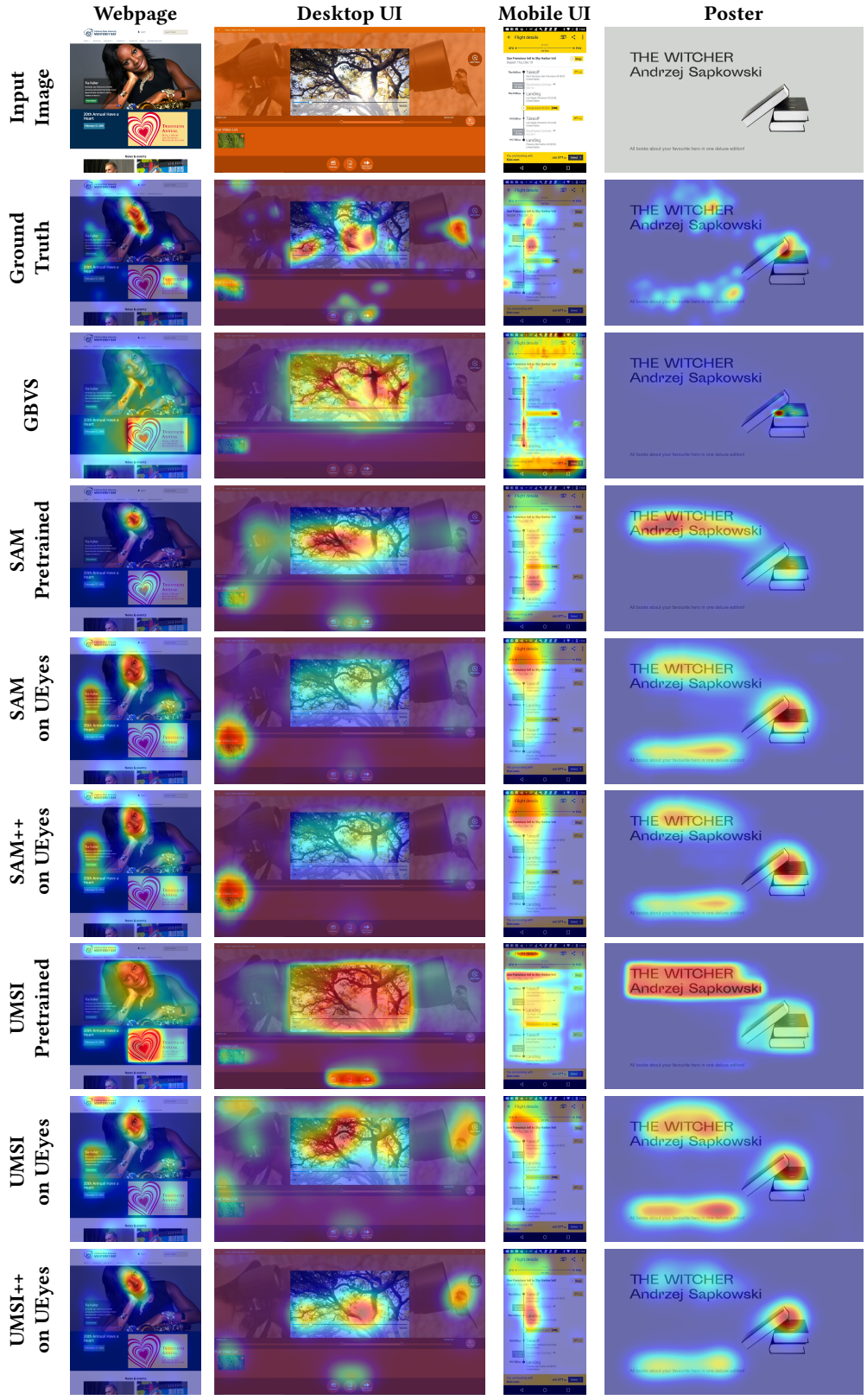


Figure 8: Saliency maps' qualitative comparison. Compared to other models, our improved model UMSI++ generates saliency maps closer to ground truth across all UI types.

Model	AUC-Judd ↑	NSS ↑	IG ↑	SIM ↑	CC ↑	KL ↓
GBVS	0.756 ± 0.104	0.256 ± 0.197	3.214 ± 0.668	0.513 ± 0.097	0.314 ± 0.193	3.916 ± 2.630
SAM Pretrained	0.822 ± 0.074	0.377 ± 0.170	3.143 ± 0.768	0.562 ± 0.081	0.522 ± 0.146	2.721 ± 1.457
SAM on UEyes	0.885 ± 0.057	**0.434 ± 0.185**	3.337 ± 0.774	0.663 ± 0.081	0.720 ± 0.127	2.016 ± 1.263
SAM++ on UEyes	0.868 ± 0.060	0.414 ± 0.179	3.165 ± 0.774	0.666 ± 0.080	0.717 ± 0.127	1.604 ± 1.185
UMSI Pretrained	0.778 ± 0.090	0.346 ± 0.178	3.177 ± 0.796	0.521 ± 0.078	0.431 ± 0.155	3.757 ± 1.769
UMSI on UEyes	0.878 ± 0.066	0.424 ± 0.187	**3.376 ± 0.807**	0.639 ± 0.085	0.699 ± 0.156	2.676 ± 1.408
UMSI++ on UEyes	**0.905 ± 0.044**	0.401 ± 0.173	3.320 ± 0.744	**0.733 ± 0.069**	**0.833 ± 0.078**	**1.166 ± 0.772**

Table 1: Saliency maps' quantitative evaluation, with mean ± SD reported for each metric. Arrows indicate the direction of the importance; e.g., ↑ means "higher is better". The best result in each column is presented in bold. UMSI++ outperforms the other models for most evaluation metrics.

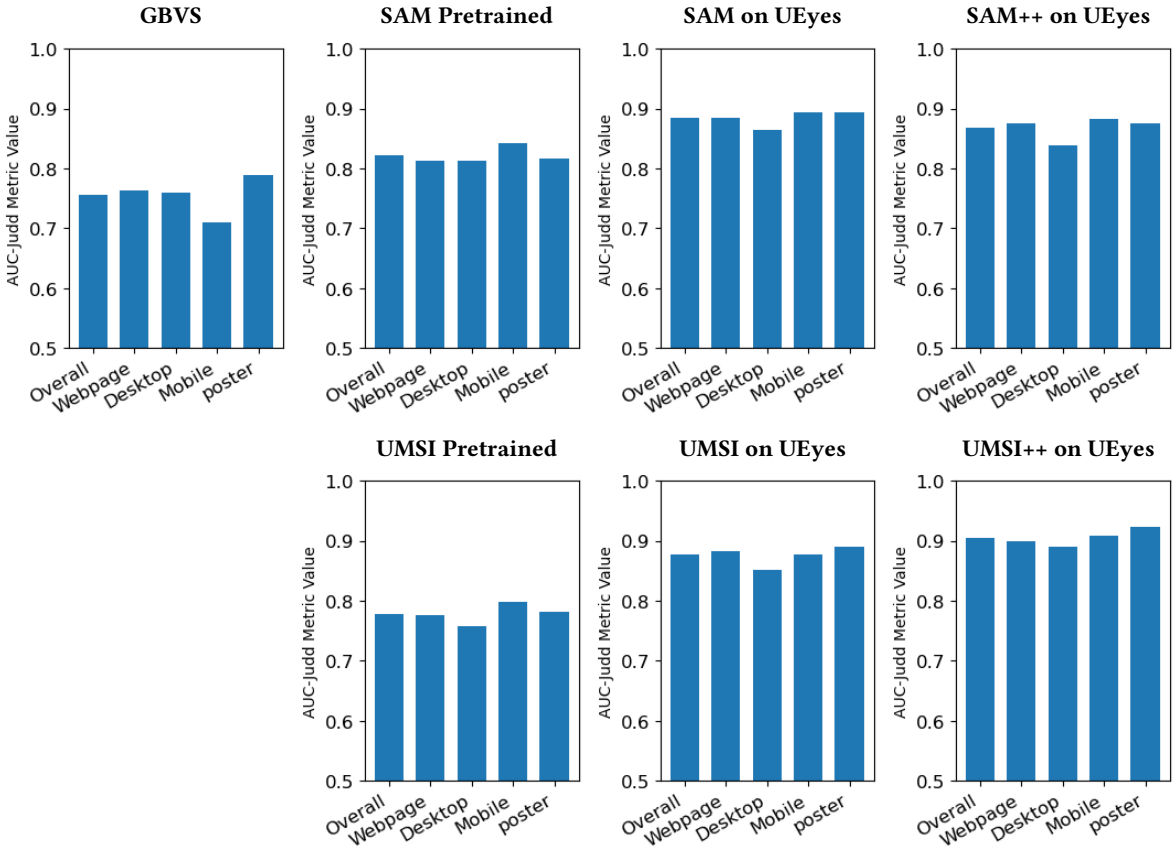


Figure 9: Comparison of saliency map models' predictive accuracy, with AUC-Judd (designed to measure saliency map performance) as the metric. Larger values indicate higher accuracy. The figure shows that UMSI++ performs best across all the UI types.

connected with posters better than the other models do, thanks to its detection of lower saliency at the top of posters.

6 ASSESSING SCANPATH MODELS

In scanpath prediction, the goal is to predict a sequence of fixations. The problem is much more challenging than that of saliency maps because the order of the fixations must be retained. Below, we report on how well computational models fared with the four UI types, from a comparison of four models:

Itti-Koch-based model [37]. The Itti-Koch-based model is a model proposed in the pre-deep-learning era. It generates a saliency map by extracting visual features for intensity, color, and orientation through a set of linear center-surround operations, then employs a "winner-takes-all" strategy to select the attended position. The

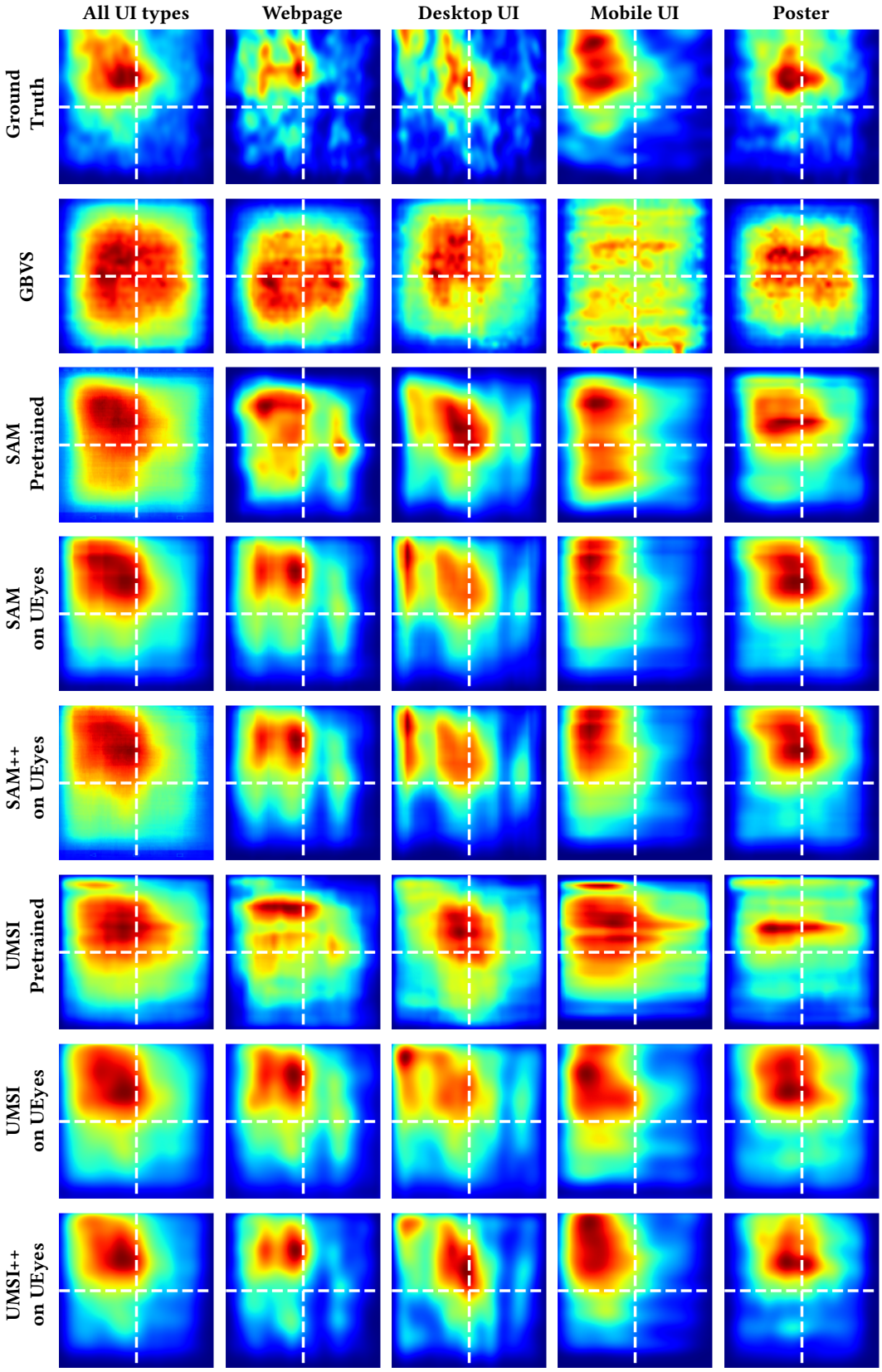


Figure 10: Comparison of the location bias of saliency maps predicted by different models across UI types. UMSI++ shows the greatest similarity to the ground-truth location bias.

model repeatedly applies inhibition of return feedback to inhibit the chosen position in the saliency map and, thereby, arrive at the resulting scanpath.

DeepGaze III [47]. DeepGaze III predicts the sequence of fixation points in scanpaths over static images. It takes both the input image and the positions of the previous four fixation points to predict the density/probabilistic map for the next fixation point. After that, it generates the scanpath by recursively selecting the next fixation point with the highest probability value from the density map and adding the new predicted fixation point to infer the density map for the next point. DeepGaze III's method concentrates on applying fixation point detection to form the final scanpath.

DeepGaze++ (Ours). DeepGaze++ offers an alternative. Although DeepGaze III can take the information of previous fixation points to generate the density of the next point, it often arrives at similar density maps for consecutive fixation point predictions. In the proposed modification, we repeatedly select the position with the highest probability from the density map and apply inhibition of return to inhibit the chosen position in the saliency map. For the ith previous fixation point's information, we assign a weight of $1 - 0.1 \cdot (i - 1)$ to the inhibition of return feedback so that older fixation points have less effect on the prediction results.

PathGAN [3]. PathGAN is a deep convolutional-recurrent neural network trained on adversarial examples. The generator takes the image as input to generate the corresponding scanpath. The discriminator encodes both the image and the scanpath to ascertain whether a scanpath is realistic for a given image. This enables PathGAN to generate more realistic scanpaths. However, it focuses exclusively on the path and cannot predict fixation points.

PathGAN++ (Ours). The PathGAN model can generate more accurate trajectories for scanpaths than other models, but we can increase the scanpaths' accuracy still further to have PathGAN++ by adding a Dynamic Time Warping (DTW) loss term that maximizes the similarity between the predicted scanpath and the ground truth in temporal order.

6.1 Evaluation Metrics

We used six metrics, measuring various properties, to evaluate the scanpath models. All are commonly applied for scanpath evaluation [1, 28], with the first three described below seeing the most frequent use in this field, since they capture the temporal and spatial aspects of visual attention. The final three metrics were employed for completeness.

Dynamic Time Warping (DTW). DTW is a standard metric for similarity between two temporal sequences, of different lengths [5, 77]. The DTW metric finds the optimal match and computes the distance for two scanpaths monotonically without missing essential features.

Time Delay Embedding (TDE). TDE creates the sets of time-delay embedding vectors for the predicted and the ground-truth scanpaths by collecting all the consecutive subscanpaths of a given length as vectors [84, 88]. We look for the vector from the predicted scanpath for each time-delay embedding vector from the ground-truth scanpath with the minimal distance. Thus, TDE measures the differences between subscanpaths to evaluate the scanpaths.

Eyenalysis. Eyenalysis performs a double mapping between two scanpaths [60]: for each fixation point along one scanpath, the procedure finds the spatially closest fixation point on the other scanpath, and then it performs the same procedure the other way around. Eyenalysis measures the average distances for all the closest pairs found.

Cross-Recurrence (REC). REC involves measuring the matching ratio of fixation points within the two scanpaths [98]. To use this metric, we truncate the two scanpaths to the same length, that of the shorter of the two scanpaths. Then, we define fixation pairs whose distance is below a certain threshold value as recurrences (we set the threshold to be the image size scaled by 0.05). The REC process counts the recurrences and computes the percentage of recurrences out of all the fixation pairs on the two scanpaths.

Weighted Determinism (DET). DET is the percentage of recurrent fixation points on subscanpaths in which all the pairs of corresponding fixation points are recurrences and all such recurrent fixation point pairs contain different fixation points from both scanpaths [1, 28]. In its original formulation, the Determinism metric [1] produces only the number of corresponding subscanpaths. We propose computing their percentage, to measure the subscanpaths better.

Center of Recurrence Mass (CORM). CORM refers to the distance between the center of recurrences, thus indicating the dominant lag of recurrences [1, 28]. The CORM score is lower when the recurrent fixation point pairs on the scanpaths are closer in time.

Whereas DTW, TDE, and Eyenalysis measure fixations' position and sequence in temporal order as they match the two sequences differently, REC and DET measure only the similarity of fixation positions. They have higher values if the fixation points in the two sequences are close, irrespective of the temporal order. The CORM measurement serves to detect the dominant lag of recurrences.

6.2 Results

6.2.1 Qualitative Evaluation. Our qualitative comparison of the various models across UI types is depicted in Figure 11. Overall, the models cannot predict results accurately relative to the ground-truth data. The pretrained PathGAN model and the DeepGaze III model get stuck in local areas, so the predicted points end up in clusters. Because of the similar density maps predicted by DeepGaze III for consecutive fixation points, that model selects positions for fixation points that are nearby, thereby producing a cluster of points and getting "bogged down" in that cluster. PathGAN can only generate the scanpath, without considering fixation points. Therefore, it is impossible to infer which points users give more visual attention from the PathGAN results. The Itti-Koch-based model, PathGAN trained on UEyes, DeepGaze++, and PathGAN++ show better prediction results. That said, most scanpaths predicted by PathGAN and PathGAN++ trained on UEyes are biased to be around the center of the UIs. Also, all the models tend to predict scanpaths with many fixation points, not all of them in salient areas.

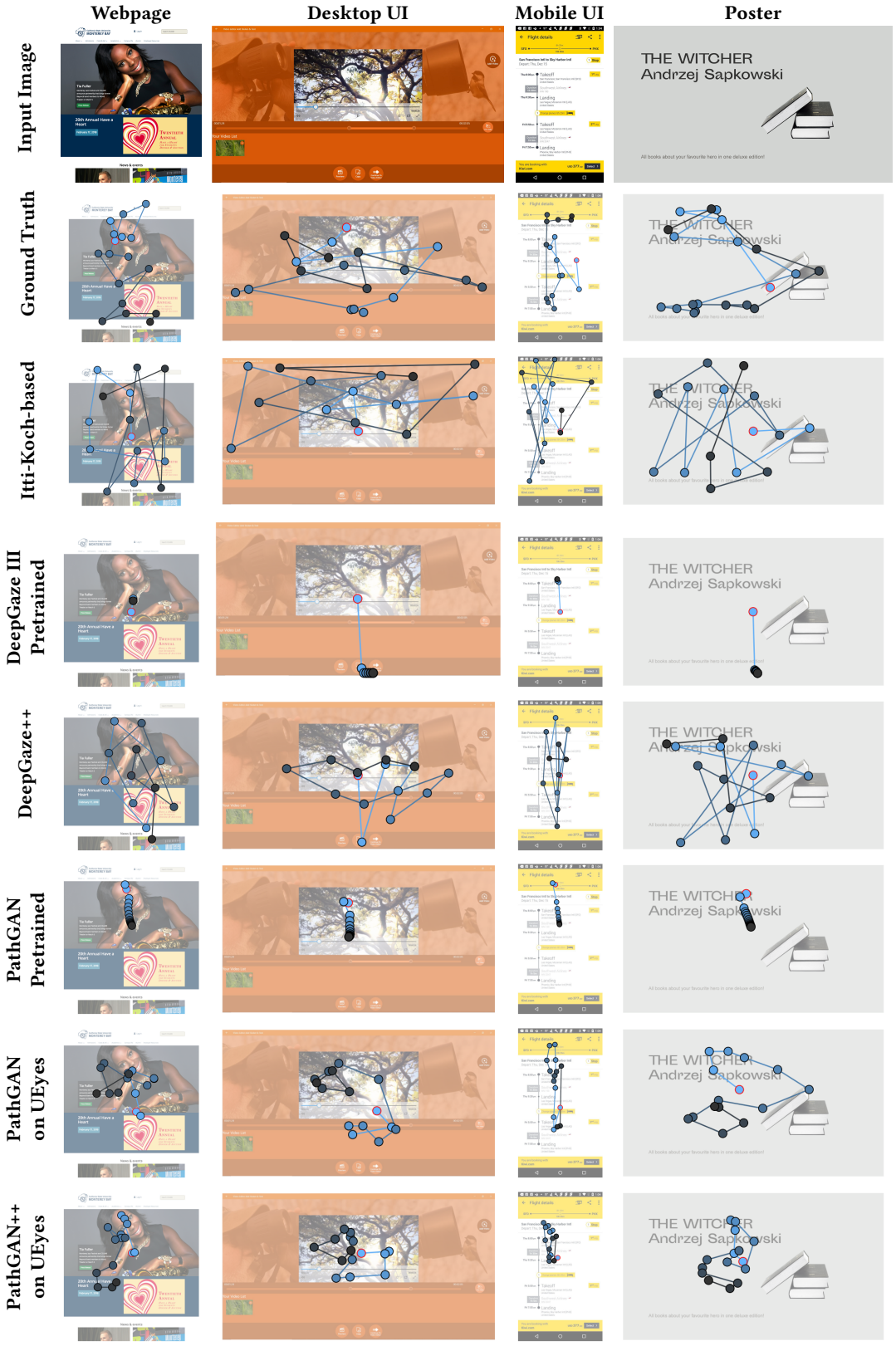


Figure 11: Scanpaths' qualitative comparison. DeepGaze++ can predict fixation points better but cannot predict realistic scanpaths. PathGAN++ is able to predict a realistic trajectory but not accurate fixation points. Trajectories' first portion is presented in blue, and their end in black (color gradient). The starting point is highlighted with a red border.

6.2.2 Quantitative Evaluation. Table 2 presents how the models stack up by each of the evaluation metrics outlined in Section 6.1. For a fair comparison, since these metrics depend on the scanpath length, we made sure that the predicted scanpaths generated by all the models have exactly 15 fixation points. Since DTW is a standard metric for scanpaths' evaluation (where smaller values indicate higher accuracy), we can elaborate on our comparison of the models across UI types in the manner shown in Figure 12. For all models apart from the Itti-Koch-based one, desktop UIs have higher DTW values, indicating lower accuracy of the predicted values than seen with other UI types: scanpaths in desktop UI conditions are harder to predict. All of the models are at their best with mobile UIs. DeepGaze III was the worst-performing model for all UI types except mobile UIs. The PathGAN++ model shows superior performance in comparison to the other models by the DTW, TDE, and Eyenalysis metrics. This is a testament to our model's ability to simulate real scanpath trajectories. However, the results are still qualitatively inaccurate. We can conclude, then, that the metrics currently applied for evaluating scanpaths may not be sufficient to capture the more nuanced aspects of eye movements.

6.2.3 Comparison between PathGAN++ and DeepGaze++. Comparing the performance of PathGAN++ and DeepGaze++ reveals that each model has its own strengths and limitations. Though PathGAN++ excels at generating realistic trajectories by dint of the discriminative component in the model architecture, it falls short in predicting proper fixation points, and the points generated often lie outside the areas of interest. DeepGaze++, on the other hand, is better at predicting fixation points, since its operation is based on saliency maps that highlight elements in the UIs. However, it can suffer from repetitive density maps for consecutive fixation point predictions, leading to unrealistic scanpaths. Additionally, the mechanism for inhibition of return is deterministic and not differentiable, so it cannot be optimized by means of any loss terms. This trait can hamper its optimization.

6.2.4 Saccade Angle and Amplitude Distribution. Figure 13 characterizes the saccade-angle and amplitude-distribution aspect of our comparison. None of the models can capture the same distributions as the ground-truth data. Human saccade directions are primarily from left to right, with a small proportion of motions from top to bottom. The pretrained PathGAN model and DeepGaze III have clustered distributions due to the "stuck points" on the predicted scanpaths. PathGAN trained on UEyes and PathGAN++ both display an incorrect center bias to the distribution. Furthermore, the inhibition of return implemented in the Itti-Koch-based model and DeepGaze++ avoids small distances between consecutive fixation points. Hence, the saccade amplitudes are more significant than in the ground truth. We found that most saccade directions predicted by DeepGaze++ are rightward ones for desktop UIs, mobile UIs, and posters, which demonstrates that DeepGaze++ can capture the actual tendencies visible with these UI types. However, it is still incorrect for webpages and cannot predict the gaze's tendency to move toward the bottom of the UIs.

6.2.5 Visited and Revisited Elements. Our comparison of visited- and revisited-element ratios for the various models is described in *Supplementary Materials*. All the models can correctly predict that text elements are more likely to receive fixations than images are. The pretrained PathGAN model and DeepGaze III underestimate the visiting and revisiting ratios both. DeepGaze++ displays the best prediction for the former but overestimates the revisit ratios for all UI types. All the models reflect the fact that both ratios are lower with mobile UIs than with other UI types. Still, every model except DeepGaze++ underestimates the visiting and the revisiting ratio for this type of UI. Most models' predictions for element visit and revisit ratios are the closest to the ground truth in the case of poster designs. PathGAN++ is the model that yields the predictions closest to the ground truth for visiting and revisiting of elements, with the exception of its underestimation for mobile UIs.

7 DISCUSSION

Our study sheds new light on the eye-movement behavior that occurs with specific UI types. Here, we summarize the main findings pertaining to how people look at UIs, then discuss the challenges and limitations that accompany current computational models.

7.1 How People Look at UIs

We have found that, in general, users pay more attention to the upper-left region in a UI. While prior work demonstrated this for mobile UIs [53], we can now confirm a similar pattern extending across all types of UI considered in our project, inclusive of poster designs. Also, we have found that saccades take the gaze mainly toward the right or bottom portion of the UI. Further, movements of the gaze toward the right exhibit larger distances between consecutive points than motions toward the bottom.

At the same time, we found that text elements are more likely to be fixated on than images, which explains saccades' typical motion from left to right rather than *vice versa*, although the latter result may be an artifact of our dataset, since most of our UIs, being in the English language, forced participants to read the text from left to right. Still, the ratio of images to text does not affect the ratios of visited and revisited elements in these two element categories. Another noteworthy finding is that saccades toward the right-hand part of the UIs show larger distances between consecutive points than those landing nearer the bottom. It is among the evidence that user interfaces are not glanced at in the same manner as natural scenes [53]. Instead of a center bias, there is a strong top-left bias.

Our data allow a deeper dive into various subtle differences among the types of UIs examined. Several distinctions exemplify this:

- **Webpages:** We found that users tend to look from left to right on webpages, showing larger inter-fixation distances than with the other interface types. These large distances might explain why computational scanpath models exhibit their worst performance with webpages while computational saliency models perform quite well with other types of UIs.
- **Desktop UIs:** Because desktop UIs have two salient areas (one just above the center and the other at top left), it proves difficult for computational models of saliency maps and scanpaths to deliver accurate predictions. These were found to perform poorly with the multi-modal gaze distributions in such conditions.

Model	DTW ↓	TDE ↓	Eyenalysis ↓	REC ↑	DET ↑	CORM ↓
Itti-Koch-based	6.282 ± 0.973	0.147 ± 0.027	0.043 ± 0.022	2.224 ± 2.053	2.021 ± 10.854	34.497 ± 22.890
DeepGaze III Pretrained	7.650 ± 2.899	0.250 ± 0.078	0.124 ± 0.072	1.290 ± 3.281	1.025 ± 8.510	**13.838 ± 24.082**
DeepGaze++	5.230 ± 1.180	0.133 ± 0.031	**0.043 ± 0.022**	1.876 ± 1.700	1.778 ± 10.046	31.590 ± 23.120
PathGAN Pretrained	4.381 ± 1.559	0.160 ± 0.054	0.072 ± 0.036	**3.896 ± 5.049**	**7.039 ± 18.651**	22.528 ± 22.970
PathGAN on UEyes	4.354 ± 1.322	0.121 ± 0.040	0.045 ± 0.024	2.414 ± 2.455	5.687 ± 17.960	27.613 ± 21.644
PathGAN++ on UEyes	**4.236 ± 1.332**	**0.120 ± 0.041**	**0.043 ± 0.022**	2.810 ± 2.743	5.761 ± 16.053	27.956 ± 21.544

Table 2: Evaluation of scanpaths, with the mean ± SD reported for each metric. Arrows denote the direction of the importance; e.g., ↑ means "higher is better." Each column's best result is highlighted in boldface. PathGAN++ outperforms the other models by all three metrics applied for measuring the fixation sequence in temporal order (DTW, TDE, and Eyenalysis).

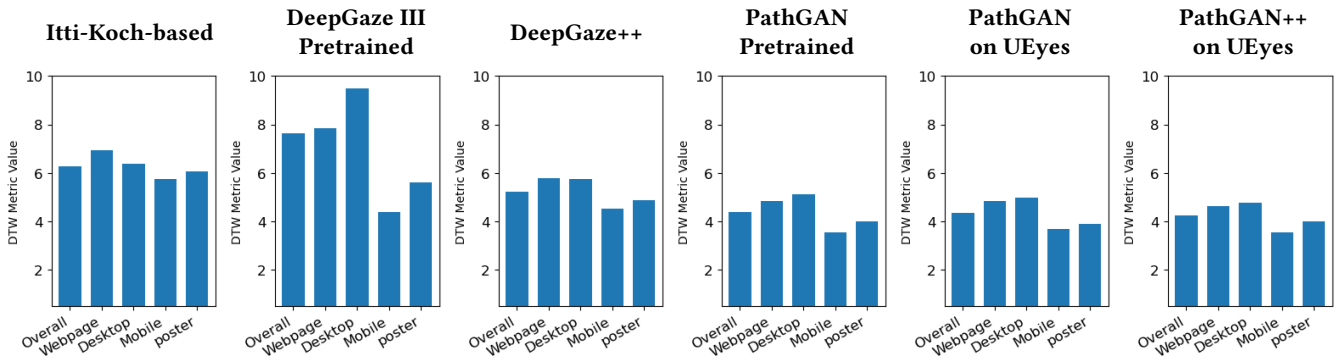


Figure 12: Comparison of predictive accuracy for scanpath models, with DTW as the metric (measuring the fixation sequence in temporal order). Smaller values indicate higher accuracy. The figure shows that DeepGaze++ performs best, across all UI types.

Mobile UIs: Mobile UIs have lower visit and revisit ratios than other UI types, indicating that users focus more on the "attractive" elements while ignoring the others and that there is less of a tendency to return to the same content. Further, we noticed that most scanpath models can predict the low visit ratio of mobile UIs and, accordingly, display better predictive accuracy with mobile UIs than with all other UI types.

Posters: As with other UIs, users tend to scan posters from left to right, with a small number of saccades toward the bottom and with highly varied fixation distances. Here, the distances of consecutive fixation points show more pronounced variation than other UI types'. This renders their prediction by current computational scanpath models harder.

7.2 Current Computational Models

Our results highlight that, in efforts to predict visual saliency, training of computational models with eye movement over user interfaces yields superior performance to training with proxy data, such as mouse movements or manual annotations, or even training with data collected from viewing of natural scenes. While that is unsurprising, we have demonstrated this superiority quantitatively. In particular, we showed that training the UMSI on UEyes increases its AUC performance score from 0.778 to 0.878. Upon inspecting the predictions, we found that much of this difference can be attributed to cases of over-detection by the UMSI: it predicts saliency across expanses of the UI more extensive than what the user may have had time to inspect, and this is reflected in its high false positive rate. That said, after training on our dataset and with our modifications to the model, the accuracy of the UMSI improved considerably.

7.3 Limitations and Future Work

7.3.1 The Mobile UI Viewing Setting. The fixed-screen setting used in our experiment, while guaranteeing consistent data collection and analysis across UI types, does not accurately simulate the real-world experience of viewing a mobile UI while holding a cellular phone. To improve the realism in this regard, one could rescale the mobile UI screenshots for a mean viewing distance of 30 cm, as prior literature recommends [53, 57]. With our roughly 60 cm distance between participants' eyes and the screen, the physical size of the stimuli displayed should be about twice what it is on a mobile screen. This issue notwithstanding, our findings for mobile UIs corroborate reports by Leiva et al. [53].

7.3.2 Semantic Understanding of UI Elements. The current classification of visited and revisited UI elements into broad categories (text, image, and face) does not capture the semantic differences within each category. Future work could focus on developing more detailed and nuanced classification of visited and revisited UI elements by extracting their semantic meaning [93] to afford a greater understanding of users' gaze-related behaviors.

7.3.3 False Positives in Saliency Maps. While computational models of saliency maps can capture informative areas such as images and text regions, they still tend to generate false positive errors and

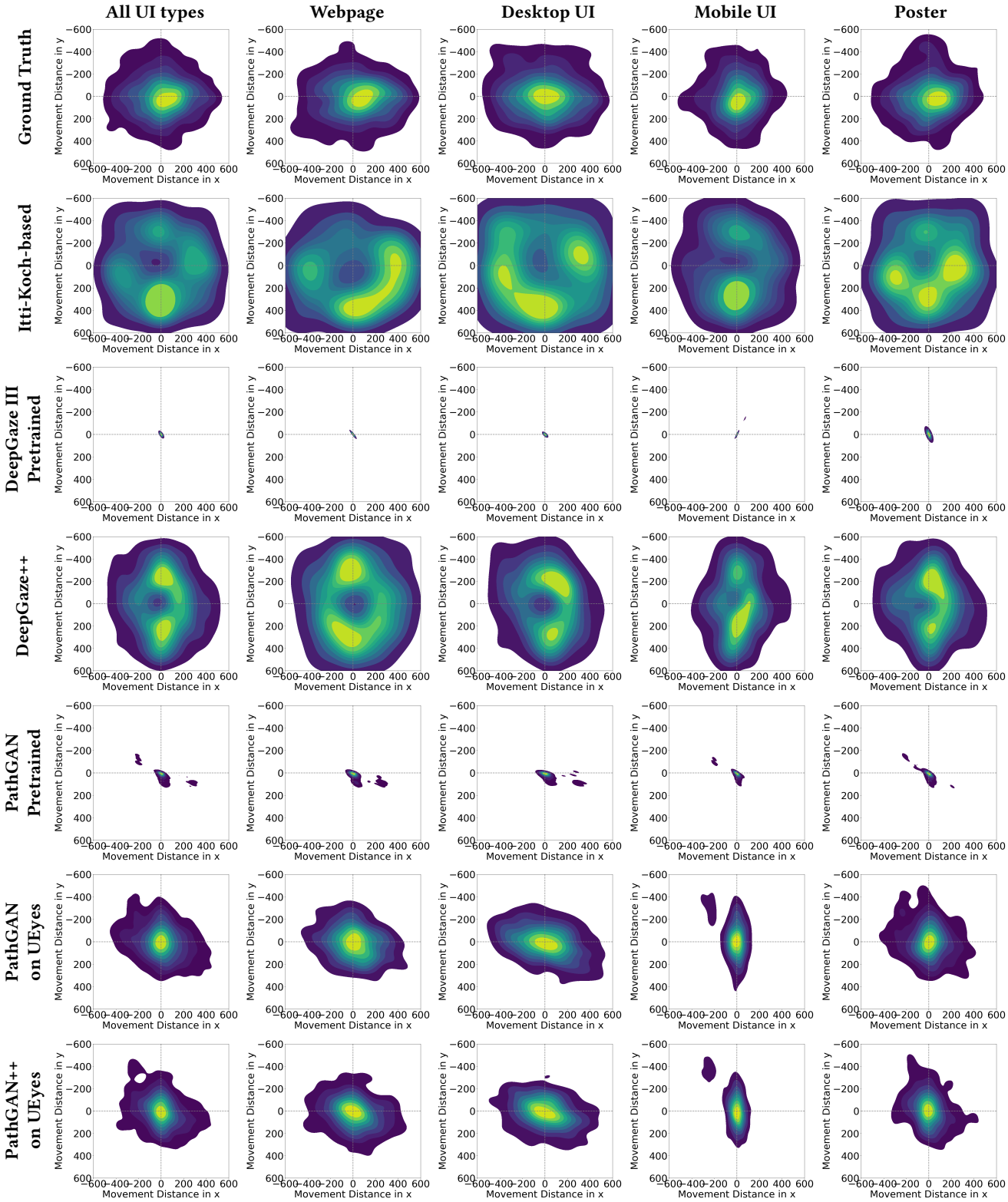


Figure 13: Distributions of saccade angles and amplitudes compared across scanpath-based predictive models. Human saccade directions are primarily left-to-right, with a small proportion being from top to bottom. None of the models can capture the same distributions as the ground-truth data.

over-detect salient areas, thereby producing low accuracy and reliability. Future work can improve the model's ability to differentiate between truly salient areas and false positives, and it could bring additional features, such as user task goals, into play to guide the saliency prediction.

7.3.4 Inaccurate Scanpath Models. Today, no single scanpath model can accurately capture both the scanpath trajectories and the fixation points of human eye movements. Further improving the model requires a fuller understanding of the factors that influence gaze behavior (such as visit and revisit tendencies) and incorporation of those factors into the model. Additionally, better metrics are needed for assessing the quality of predicted scanpaths. Developing such metrics should contribute to a deeper understanding of the scanpath models' performance and, by doing so, guide the design of better models.

7.3.5 Individual-Specific Differences. Individuals differ in the viewing strategies they apply when looking at user interfaces. Person-to-person variations in viewing strategy can affect gaze behavior, and predictive models need to take them into account. Future work could focus on understanding and modeling the individual-to-individual differences in viewing strategies, in general terms and for each of the UI types. This can be accomplished by means of personalized predictive models that account for differences between individuals.

8 CONCLUSION

In this paper, we present UEyes, a large-scale eye-tracking dataset that covers 1,980 UIs of various types, along with multi-duration saliency maps and scanpaths. Moreover, we present the first in-depth analysis and comparison of eye-movement tendencies across common UI types. We also contribute solid performance analysis of state-of-the-art predictive models for saliency maps and scanpaths across the various UI types.

Open Science

The dataset and trained models are available at https://userinterfaces.aalto.fi/ueyeschi23. The dataset includes raw CSV log files recorded with the GP3 HD eye tracker, associated heatmaps and scanpaths, the image stimuli (screenshots), and metadata referring to the design type.

ACKNOWLEDGMENTS

This work was supported by Aalto University's Department of Information and Communications Engineering, the Finnish Center for Artificial Intelligence (FCAI), the Academy of Finland through the projects Human Automata (grant 328813) and BAD (grant 318559), the Horizon 2020 FET program of the European Union (grant CHIST-ERA-20-BCI-001), and the European Innovation Council Pathfinder program (SYMBIOTIK project, grant 101071147). We appreciate Chuhan Jiao's initial implementation of the baseline methods for saliency prediction and active discussion with Yao (Marc) Wang.

REFERENCES

[1] Nicola C Anderson, Fraser Anderson, Alan Kingstone, and Walter F Bischof. 2015. A comparison of scanpath comparison methods. *Behavior research methods* 47, 4 (2015), 1377–1392.

[2] Marc Assens, Xavier Giro-i Nieto, Kevin McGuinness, and Noel E. O'Connor. 2017. SaltiNet: Scan-Path Prediction on 360 Degree Images Using Saliency Volumes. In *2017 IEEE International Conference on Computer Vision Workshops (ICCVW)*. 2331–2338. https://doi.org/10.1109/ICCVW.2017.275

[3] Marc Assens, Xavier Giro i Nieto, Kevin McGuinness, and Noel E. O'Connor. 2018. PathGAN: Visual Scanpath Prediction with Generative Adversarial Networks. *ECCV Workshop on Egocentric Perception, Interaction and Computing (EPIC)*.

[4] Roman Bednarik and Markku Tukiainen. 2007. Validating the restricted focus viewer: A study using eye-movement tracking. *Behavior research methods* 39, 2 (2007).

[5] Donald J Berndt and James Clifford. 1994. Using dynamic time warping to find patterns in time series.. In *KDD workshop*, Vol. 10. Seattle, WA, USA:, 359–370.

[6] Sergey Bezryadin, Pavel Bourov, and Dmitry Ilinih. 2007. Brightness Calculation in Digital Image Processing. In *Proc. TDPF Symposium*.

[7] Ali Borji. 2019. Saliency Prediction in the Deep Learning Era: Successes, Limitations, and Future Challenges. In *CoRR abs/1810.03716 (arXiv preprint)*.

[8] A. Borji and L. Itti. 2013. State-of-the-art in visual attention modeling. *IEEE Trans. Pattern Anal. Mach. Intell.* 35, 1 (2013).

[9] Ali Borji and Laurent Itti. 2015. CAT2000: A Large Scale Fixation Dataset for Boosting Saliency Research. *CVPR 2015 workshop on "Future of Datasets"* (2015). arXiv preprint arXiv:1505.03581.

[10] Ali Borji, Hamed R. Tavakoli, Dicky N. Sihite, and Laurent Itti. 2013. Analysis of Scores, Datasets, and Models in Visual Saliency Prediction. In *Proc. ICCV*.

[11] Michelle A Borkin, Zoya Bylinskii, Nam Wook Kim, Constance May Bainbridge, Chelsea S Yeh, Daniel Borkin, Hanspeter Pfister, and Aude Oliva. 2015. Beyond memorability: Visualization recognition and recall. *IEEE transactions on visualization and computer graphics* 22, 1 (2015).

[12] Maximilian D Broda and Benjamin de Haas. 2022. Individual fixation tendencies in person viewing generalize from images to videos. *i-Perception* 13, 6 (2022), 20416695221128844.

[13] Neil Bruce and John Tsotsos. 2005. Saliency based on information maximization. *Advances in neural information processing systems* 18 (2005).

[14] Zoya Bylinskii, Tilke Judd, Ali Borji, Laurent Itti, Frédo Durand, Aude Oliva, and Antonio Torralba. 2015. Mit saliency benchmark. (2015).

[15] Zoya Bylinskii, Nam Wook Kim, Peter O'Donovan, Sami Alsheikh, Spandan Madan, Hanspeter Pfister, Fredo Durand, Bryan Russell, and Aaron Hertzmann. 2017. Learning Visual Importance for Graphic Designs and Data Visualizations. In *Proc. UIST*.

[16] Ying Cao, Rynson WH Lau, and Antoni B Chan. 2014. Look over here: Attention-directing composition of manga elements. *ACM Transactions on Graphics (TOG)* 33, 4 (2014).

[17] Zhaohui Che, Ali Borji, Guangtao Zhai, Xiongkuo Min, Guodong Guo, and Patrick Le Callet. 2020. How is Gaze Influenced by Image Transformations? Dataset and Model. *IEEE Transactions on Image Processing* 29 (2020), 2287–2300. https://doi.org/10.1109/TIP.2019.2945857

[18] Zhenzhong Chen and Wanjie Sun. 2018. Scanpath Prediction for Visual Attention Using IOR-ROI LSTM *(IJCAI'18)*. AAAI Press, 642–648.

[19] L. Cooke. 2006. Is the mouse a poor man's eye tracker?. In *Proc. STC*.

[20] Marcella Cornia, Lorenzo Baraldi, Giuseppe Serra, and Rita Cucchiara. 2016. A Deep Multi-Level Network for Saliency Prediction. In *International Conference on Pattern Recognition (ICPR)*.

[21] Marcella Cornia, Lorenzo Baraldi, Giuseppe Serra, and Rita Cucchiara. 2018. Predicting Human Eye Fixations via an LSTM-Based Saliency Attentive Model. *IEEE Transactions on Image Processing* 27, 10 (2018). https://doi.org/10.1109/TIP.2018.2851672

[22] Marcella Cornia, Lorenzo Baraldi, Giuseppe Serra, and Rita Cucchiara. 2018. SAM: Pushing the Limits of Saliency Prediction Models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision and Pattern Recognition Workshops*.

[23] Desktop UI Dataset. 2020. . https://github.com/waltteri/desktop-ui-dataset

[24] Biplab Deka, Zifeng Huang, Chad Franzen, Joshua Hibschman, Daniel Afergan, Yang Li, Jeffrey Nichols, and Ranjitha Kumar. 2017. Rico: A Mobile App Dataset for Building Data-Driven Design Applications. In *Proceedings of the 30th Annual Symposium on User Interface Software and Technology (UIST '17)*.

[25] Guanqun Ding, Nevrez İmamoğlu, Ali Caglayan, Masahiro Murakawa, and Ryosuke Nakamura. 2022. SalFBNet: Learning pseudo-saliency distribution via feedback convolutional networks. *Image and Vision Computing* 120 (2022), 104395. https://doi.org/10.1016/j.imavis.2022.104395

[26] Richard Droste, Jianbo Jiao, and J. Alison Noble. 2020. Unified Image and Video Saliency Modeling. In *Computer Vision – ECCV 2020*, Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm (Eds.). Springer International Publishing, Cham, 419–435.

[27] Sergio Etchebehere and Elena Fedorovskaya. 2017. On the Role of Color in Visual Saliency. *Intl. Symp. Electronic Imaging* 6 (2017).

[28] Ramin Fahimi and Neil DB Bruce. 2021. On metrics for measuring scanpath similarity. *Behavior Research Methods* 53, 2 (2021), 609–628.

[29] Camilo Fosco, Vincent Casser, Amish Kumar Bedi, Peter O'Donovan, Aaron Hertzmann, and Zoya Bylinskii. 2020. Predicting visual importance across graphic design types. In *Proceedings of the 33rd Annual ACM Symposium on User Interface Software and Technology*. 249–260.

[30] Camilo Fosco, Anelise Newman, Pat Sukhum, Yun Bin Zhang, Nanxuan Zhao, Aude Oliva, and Zoya Bylinskii. 2020. How much time do you have? modeling multi-duration saliency. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 4473–4482.

[31] S. Frintrop, E. Rome, and H. I. Christensen. 2010. Computational visual attention systems and their cognitive foundations: A survey. *ACM Trans. Appl. Percept.* 7, 1 (2010).

[32] Shahrbanoo Hamel, Nathalie Guyader, Denis Pellerin, and Dominique Houzet. 2014. Contribution of Color Information in Visual Saliency Model for Videos. In *Proc. ICISP*. 213–221.

[33] Rui Han and Shuangjiu Xiao. 2018. Human Visual Scanpath Prediction Based on RGB-D Saliency. In *Proceedings of the 2018 International Conference on Image and Graphics Processing* (Hong Kong, Hong Kong) *(ICIGP 2018)*. Association for Computing Machinery, New York, NY, USA, 180–184. https://doi.org/10.1145/3191442.3191463

[34] Jonathan Harel, Christof Koch, and Pietro Perona. 2006. Graph-based visual saliency. *Advances in neural information processing systems* 19 (2006).

[35] J. M. Henderson. 1993. Eye movement control during visual object processing: effects of initial fixation position and semantic constraint. *Can. J. Exp. Psychol.* 47, 1 (1993).

[36] Xun Huang, Chengyao Shen, Xavier Boix, and Qi Zhao. 2015. SALICON: Reducing the Semantic Gap in Saliency Prediction by Adapting Deep Neural Networks. In *2015 IEEE International Conference on Computer Vision (ICCV)*. 262–270. https://doi.org/10.1109/ICCV.2015.38

[37] Laurent Itti, Christof Koch, and Ernst Niebur. 1998. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on pattern analysis and machine intelligence* 20, 11 (1998), 1254–1259.

[38] Sen Jia. 2018. EML-NET: An Expandable Multi-Layer NETwork for Saliency Prediction. *CoRR* abs/1805.01047 (2018). arXiv:1805.01047 http://arxiv.org/abs/1805.01047

[39] Ming Jiang, Shengsheng Huang, Juanyong Duan, and Qi Zhao. 2015. SALICON: Saliency in Context. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 1072–1080. https://doi.org/10.1109/CVPR.2015.7298710

[40] James M Joyce. 2011. Kullback-leibler divergence. In *International encyclopedia of statistical science*. Springer, 720–722.

[41] Tilke Judd, Frédo Durand, and Antonio Torralba. 2012. A Benchmark of Computational Models of Saliency to Predict Human Fixations. In *MIT Technical Report*.

[42] T. Judd, K. Ehinger, F. Durand, and A. Torralba. 2009. Learning to predict where humans look. In *Proc. ICCV*.

[43] Tilke Judd, Krista Ehinger, Frédo Durand, and Antonio Torralba. 2009. Learning to predict where humans look. In *2009 IEEE 12th international conference on computer vision*. IEEE, 2106–2113.

[44] Nam Wook Kim, Zoya Bylinskii, Michelle A Borkin, Krzysztof Z Gajos, Aude Oliva, Frēdo Durand, and Hanspeter Pfister. 2017. BubbleView: an interface for crowdsourcing image importance maps and tracking visual attention. *ACM Transactions on Computer-Human Interaction (TOCHI)* 24, 5 (2017). https://doi.org/10.1145/3131275

[45] Nam Wook Kim, Zoya Bylinskii, Michelle A Borkin, Aude Oliva, Krzysztof Z Gajos, and Hanspeter Pfister. 2015. A crowdsourced alternative to eye-tracking for visualization understanding. In *Proceedings of the 33rd Annual ACM Conference Extended Abstracts on Human Factors in Computing Systems*. 1349–1354.

[46] Alexander Kroner, Mario Senden, Kurt Driessens, and Rainer Goebel. 2020. Contextual encoder–decoder network for visual saliency prediction. *Neural Networks* 129 (2020), 261–270. https://doi.org/10.1016/j.neunet.2020.05.004

[47] Matthias Kümmerer, Matthias Bethge, and Thomas SA Wallis. 2022. DeepGaze III: Modeling free-viewing human scanpaths with deep learning. *Journal of Vision* 22, 5 (2022).

[48] Matthias Kümmerer, Lucas Theis, and Matthias Bethge. 2014. Deep gaze i: Boosting saliency prediction with feature maps trained on imagenet. *arXiv preprint arXiv:1411.1045* (2014).

[49] Matthias Kümmerer, Thomas Wallis, and Matthias Bethge. 2014. How close are we to understanding image-based saliency? *arXiv preprint arXiv:1409.7686* (2014).

[50] Matthias Kümmerer, Thomas SA Wallis, and Matthias Bethge. 2015. Information-theoretic model comparison unifies saliency metrics. *Proceedings of the National Academy of Sciences* 112, 52 (2015), 16054–16059.

[51] Matthias Kummerer, Thomas SA Wallis, Leon A Gatys, and Matthias Bethge. 2017. Understanding low-and high-level contributions to fixation prediction. In *Proceedings of the IEEE international conference on computer vision*. 4789–4798.

[52] Olivier Le Meur, Patrick Le Callet, and Dominique Barba. 2007. Predicting visual fixations on video based on low-level visual features. *Vision research* 47, 19 (2007), 2483–2498.

[53] Luis A Leiva, Yunfei Xue, Avya Bansal, Hamed R Tavakoli, Tuðçe Köroðlu, Jingzhou Du, Niraj R Dayama, and Antti Oulasvirta. 2020. Understanding visual saliency in mobile user interfaces. In *22nd International conference on human-computer interaction with mobile devices and services*. 1–12.

[54] Guanbin Li and Yizhou Yu. 2015. Visual Saliency Based on Multiscale Deep Features. In *Proc. CVPR*. 5455–5463.

[55] Akis Linardos, Matthias Kümmerer, Ori Press, and Matthias Bethge. 2021. DeepGaze IIE: Calibrated prediction in and out-of-domain for state-of-the-art saliency modeling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*. 12919–12928.

[56] Gitte Lindgaard, Gary Fernandes, Cathy Dudek, and J. Brown. 2006. Attention web designers: You have 50 milliseconds to make a good first impression! *Behav. Inform. Technol.* 25, 2 (2006).

[57] Jennifer Long, Rene Cheung, Simon Duong, Rosemary Paynter, and Lisa Asper. 2017. Viewing distance and eyestrain symptoms with prolonged viewing of smartphones. *Clinical and Experimental Optometry* 100, 2 (2017), 133–137.

[58] Jianxun Lou, Hanhe Lin, David Marshall, Dietmar Saupe, and Hantao Liu. 2022. TranSalNet: Towards perceptually relevant visual saliency prediction. *Neurocomputing* 494 (2022), 455–467. https://doi.org/10.1016/j.neucom.2022.04.080

[59] S. Marat, A. Rahman, D. Pellerin, N. Guyader, and D. Houzet. 2013. Improving Visual Saliency by Adding 'Face Feature Map' and 'Center Bias'. *Cogn. Comput.* 5, 1 (2013).

[60] S Mathot, F Cristino, ID Gilchrist, and J Theeuwes. 2012. Eyenalysis: A similarity measure for eye movement patterns. *Journal of Eye Movement Research* 5 (2012), 1–15.

[61] Aliaksei Miniukovich and Antonella De Angeli. 2014. Visual Impressions of Mobile App Interfaces. In *Proc. NordiCHI*. 31–40.

[62] Aliaksei Miniukovich and Maurizio Marchese. 2020. Relationship between visual complexity and aesthetics of webpages. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*. 1–13.

[63] A. Mishra, Y. Aloimonos, and C.L. Fah. 2009. Active segmentation with fixation. In *Proc. ICCV*. 468–475.

[64] Thuyen Ngo and B.S. Manjunath. 2017. Saccade gaze prediction using a recurrent neural network. In *2017 IEEE International Conference on Image Processing (ICIP)*. 3435–3439. https://doi.org/10.1109/ICIP.2017.8296920

[65] A. Nuthmann and J. M. Henderson. 2014. Object-based attentional selection in scene viewing. *J. Vis.* 10, 8 (2014).

[66] J. P. Ossandon, S. Onat, and P. König. 2014. Spatial biases in viewing behavior. *J. Vis.* 14, 2 (2014).

[67] Peter O'Donovan, Aseem Agarwala, and Aaron Hertzmann. 2014. Learning layouts for single-pagegraphic designs. *IEEE transactions on visualization and computer graphics* 20, 8 (2014).

[68] Junting Pan, Cristian Canton, Kevin McGuinness, Noel E. O'Connor, Jordi Torres, Elisa Sayrol, and Xavier and Giro-i Nieto. 2017. SalGAN: Visual Saliency Prediction with Generative Adversarial Networks. In *arXiv*.

[69] Xufang Pang, Ying Cao, Rynson WH Lau, and Antoni B Chan. 2016. Directing user attention via visual flow on web designs. *ACM Transactions on Graphics (TOG)* 35, 6 (2016), 1–11.

[70] Robert J Peters, Asha Iyer, Laurent Itti, and Christof Koch. 2005. Components of bottom-up gaze allocation in natural images. *Vision research* 45, 18 (2005), 2397–2416.

[71] Hamed R. Tavakoli, Ali Borji, Jorma Laaksonen, and Esa Rahtu. 2017. Exploiting inter-image similarity and ensemble of extreme learners for fixation prediction using deep features. *Neurocomputing* 244 (2017), 10–18. https://doi.org/10.1016/j.neucom.2017.03.018

[72] Subramanian Ramanathan, Harish Katti, Nicu Sebe, Mohan Kankanhalli, and Tat-Seng Chua. 2010. An eye fixation database for saliency detection in images. In *European conference on computer vision*. Springer, 30–43.

[73] K. Rayner, S. P. Liversedge, A. Nuthmann, R. Kliegl, and Underwood G. 2009. Rayner's 1979 paper. *Perception* 38, 6 (2009).

[74] Hamed Rezazadegan Tavakoli, Esa Rahtu, and Janne Heikkilä. 2013. Stochastic bottom–up fixation prediction and saccade generation. *Image and Vision Computing* 31, 9 (2013), 686–693. https://doi.org/10.1016/j.imavis.2013.06.006

[75] Ruth Rosenholtz, Amal Dorai, and Rosalind Freeman. 2011. Do Predictions of Visual Perception Aid Design? *ACM Trans. Appl. Percept.* 8, 2 (2011).

[76] Yossi Rubner, Carlo Tomasi, and Leonidas J Guibas. 2000. The earth mover's distance as a metric for image retrieval. *International journal of computer vision* 40, 2 (2000), 99–121.

[77] Stan Salvador and Philip Chan. 2007. Toward accurate dynamic time warping in linear time and space. *Intelligent Data Analysis* 11, 5 (2007), 561–580.

[78] Peggy Seriès and Aaron Seitz. 2013. Learning what to expect (in visual perception). *Front. Hum. neurosci.* 7 (2013).

[79] Evan Shelhamer, Jonathan Long, and Trevor Darrell. 2017. Fully Convolutional Networks for Semantic Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* 39, 4 (2017).

[80] Chengyao Shen and Qi Zhao. 2014. Webpage saliency. In *European conference on computer vision*. Springer, 33–46.

[81] Jeremiah D. Still and Christopher M. Masciocchi. 2010. A Saliency Model Predicts Fixations in Web Interfaces. In *Proc. MDDAUI Workshop*.

[82] Michael J Swain and Dana H Ballard. 1991. Color indexing. *International journal of computer vision* 7, 1 (1991), 11–32.
[83] Hamed R. Tavakoli, Fawad Ahmed, Ali Borji, and Jorma Laaksonen. 2017. Saliency Revisited: Analysis of Mouse Movements Versus Fixations. In *Proc. CVPR*.
[84] Sauer Tim, A Yorke James, and Casdagli Martin. 1991. Embedology. *Journal of statistical Physics* 65, 3-4 (1991), 579–616.
[85] Richard Veale, Ziad M. Hafed, and Masatoshi Yoshida. 2017. How is visual salience computed in the brain? Insights from behaviour, neurobiology and modelling. *Philos. Trans. R. Soc. Lond. B. Biol. Sci.* 372, 1714 (2017).
[86] Ashish Verma and Debashis Sen. 2019. HMM-based Convolutional LSTM for Visual Scanpath Prediction. In *2019 27th European Signal Processing Conference (EUSIPCO)*. 1–5. https://doi.org/10.23919/EUSIPCO.2019.8902643
[87] Eleonora Vig, Michael Dorr, and David Cox. 2014. Large-Scale Optimization of Hierarchical Features for Saliency Prediction in Natural Images. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
[88] Wei Wang, Cheng Chen, Yizhou Wang, Tingting Jiang, Fang Fang, and Yuan Yao. 2011. Simulating human saccadic scanpaths on natural images. In *CVPR 2011*. IEEE, 441–448.
[89] Yixiu Wang, Bin Wang, Xiaofeng Wu, and Liming Zhang. 2017. Scanpath estimation based on foveated image saliency. *Cognitive processing* 18, 1 (2017).
[90] Alexa Top 500 Websites. 2022. https://www.expireddomains.net/alexa-top-websites/.
[91] Calden Wloka, Iuliia Kotseruba, and John K. Tsotsos. 2018. Active Fixation Control to Predict Saccade Sequences. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*. IEEE. https://doi.org/10.1109/cvpr.2018.00336
[92] Jeremy M. Wolfe and Todd S. Horowitz. 2004. What attributes guide the deployment of visual attention and how do they do it? *Nat. Rev. Neurosci.* 5, 6 (2004).
[93] Jason Wu, Xiaoyi Zhang, Jeff Nichols, and Jeffrey P Bigham. 2021. Screen Parsing: Towards Reverse Engineering of UI Models from Screenshots. In *The 34th Annual ACM Symposium on User Interface Software and Technology*. 470–483.
[94] Chen Xia, Junwei Han, Fei Qi, and Guangming Shi. 2019. Predicting Human Saccadic Scanpaths Based on Iterative Representation Learning. *IEEE Transactions on Image Processing* 28, 7 (2019), 3502–3515. https://doi.org/10.1109/TIP.2019.2897966
[95] Mulong Xie, Sidong Feng, Zhenchang Xing, Jieshan Chen, and Chunyang Chen. 2020. UIED: A Hybrid Tool for GUI Element Detection. In *Proceedings of the 28th ACM Joint Meeting on European Software Engineering Conference and Symposium on the Foundations of Software Engineering* (Virtual Event, USA) *(ESEC/FSE 2020)*. Association for Computing Machinery, New York, NY, USA, 1655–1659. https://doi.org/10.1145/3368089.3417940
[96] Pingmei Xu, Krista A Ehinger, Yinda Zhang, Adam Finkelstein, Sanjeev R Kulkarni, and Jianxiong Xiao. 2015. Turkergaze: Crowdsourcing saliency with webcam based eye tracking. *arXiv preprint arXiv:1504.06755* (2015).
[97] Amir R. Zamir, Te-Lin Wu, Lin Sun, William B. Shen, Bertram E. Shi, Jitendra Malik, and Silvio Savarese. 2017. Feedback Networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
[98] Joseph P Zbilut and Charles L Webber Jr. 2006. Recurrence quantification analysis. *Wiley encyclopedia of biomedical engineering* (2006).
[99] Ciheng Zhang, Decky Aspandi, and Steffen Staab. 2022. Predicting Eye Gaze Location on Websites. *arXiv preprint arXiv:2211.08074* (2022).
[100] Qi Zhao and Christof Koch. 2013. Learning saliency-based visual attention: A review. *Signal Process.* 93 (2013).