AN IN-DEPTH ANALYSIS OF VISUAL SALIENCY IN RENDERED VIRTUAL REALITY ENVIRONMENTS

OLUŞTURULMUŞ SANAL GERÇEKLİK ORTAMLARINDA GÖRSEL DİKKAT-ÇEKERLİĞİN DERİNLEMESİNE BİR ANALİZİ

MEHMET BAHADIR AŞKIN

ASSOC. PROF. DR. TUNCA DOĞAN

Supervisor

ASST. PROF. DR. UFUK ÇELIKCAN

2nd Supervisor

Submitted to

Graduate School of Science and Engineering of Hacettepe University

as a Partial Fulfillment to the Requirements

for the Award of the Degree of Master of Science

in Computer Engineering

January 2023

ABSTRACT

AN IN-DEPTH ANALYSIS OF VISUAL SALIENCY IN RENDERED VIRTUAL REALITY ENVIRONMENTS

Mehmet Bahadır Aşkın
Master of Science, Computer Engineering
Supervisor: Assoc. Prof. Dr. Tunca Doğan
2nd Supervisor: Asst. Prof. Dr. Ufuk Çelikcan
January 2023, 63 pages

In virtual reality (VR), visual saliency has been exploited in order to solve different problems in the literature. However, the studies to reveal the underlying mechanism and the potential of visual saliency in VR are still inadequate. In this study, we give a comprehensive analysis of visual saliency in VR displayed by modern head-mounted-displays (HMDs) by comparing various heuristic-based and learning-based saliency prediction approaches in different virtual environments. Our work consists of two main parts which zoom in on the role of the real-timeliness and depth information. To this end, in the first part of our work, we give an analysis of seven heuristic-based real-time saliency prediction methods (5 RGB, 2 RGB-D) in three different virtual environments in desktop (2D) and VR (3D) displaying situations. Our results showed that the methods that use 2D cues based on the relation between objects and their backgrounds have superiority over other methods in the real-time saliency prediction of virtual environments. In the second part, we analyze 5 RGB-D image saliency prediction methods (2 heuristic-based, 3 learning-based) in the same virtual environments under 2D and 3D displaying situations. Moreover, we deepen our investigation and include a heuristic-based RGB video saliency prediction approach

with its modified version obtained by injecting depth information into it. Our results in this part showed that the prediction achievements of the learning-based saliency approaches have a strong domination against other methods and the depth cue is a crucial factor in the saliency prediction of virtually created scenes by learning-based approaches. Additionally, we investigate the viewing behaviors of the users under 2D and 3D viewing conditions and reveal the important differences between them by using the data collected throughout a user study. Our findings point out a center bias in both viewing conditions however, it is found to be more apparent in 3D viewing.

Keywords: Computer Graphics, Deep Learning, Virtual Environments, Virtual Reality, Visual Saliency

ÖZET

OLUŞTURULMUŞ SANAL GERÇEKLİK ORTAMLARINDA GÖRSEL DİKKAT-ÇEKERLİĞİN DERİNLEMESİNE BİR ANALİZİ

Mehmet Bahadır Aşkın
Yüksek Lisans, Bilgisayar Mühendisliği
Danışman: Doç. Dr. Tunca Doğan
Eş Danışman: Dr. Öğr. Üyesi Ufuk Çelikcan
Ocak 2023, 63 sayfa

Görsel dikkat-çekerlik, sanal gerçeklikte (VR) çeşitli amaçlar için kullanılırken, VR'deki dikkat-çekerlik mekanizmasını doğru bir şekilde anlama çabaları yetersiz kalmaktadır. Bu makalede, özellikle derinliğin ve gerçek zamanlılığın katkısına odaklanarak başa takılan ekranlar (HMD'ler) kullanılarak deneyimlenen sürükleyici VR'de görsel dikkat-çekerlik tahminine yönelik öğrenme tabanlı ve sezgisel tabanlı yaklaşımların kapsamlı bir karşılaştırmalı analizini sunuyoruz. Bu amaçla, çalışmamızın ilk bölümünde beş sezgisel tabanlı gerçek zamanlı RGB ve iki sezgisel tabanlı gerçek zamanlı RGB-D dikkat-çekerlik algılama yönteminin sanal ortamlarda 2D ve 3D görüntüleme koşullarında analizini veriyoruz. Elde ettiğimiz sonuçlar, sanal ortamların gerçek zamanlı dikkat-çekerlik tahmininde, nesneler ve arka planları arasındaki ilişkileri temel alan 2D özellikleri kullanan yöntemlerin diğer yöntemlere göre üstün olduğunu göstermiştir. İkinci bölümde, 2D ve 3D görüntüleme koşulları altında üç farklı sanal ortamdan elde edilmiş bir VR veri kümesi üzerinde öğrenme tabanlı üç RGB-D görüntü dikkat-çekerlik algılama yöntemi ve iki sezgisel tabanlı RGB-D görüntü dikkat-çekerlik algılama yöntemini kullanıyoruz. Ek olarak, sezgisel tabanlı bir RGB video dikkat-çekerlik algılama yöntemi ve bunun derinlik

bilgisi eklenerek dönüştürülmüş versiyonunu dahil ederek analizimizi genişletiyoruz. Bu yedi yöntem kullanılarak elde edilen sonuçlar, VR'deki öğrenmeye dayalı RGB-D görüntü dikkat-çekerlik tahmin yöntemlerinin üstünlüğünü ortaya koyuyor ve sanal ortamların dikkat-çekerlik tahmininde derinlik bilgisinin önemini doğruluyor. Ek olarak, kullanıcıların 2D ve 3D izleme koşullarındaki izleme davranışlarını araştırıyor ve bir kullanıcı çalışması boyunca toplanan verileri kullanarak aralarındaki önemli farklılıkları ortaya koyuyoruz. Bulgularımız, her iki görüntüleme koşulunda da bir merkez yanlılığına işaret ediyor, ancak bunun 3 boyutlu görüntülemede daha belirgin olduğu görülüyor.

Keywords: Bilgisayar Grafiği, Derin Öğrenme, Sanal Ortamlar, Sanal Gerçeklik, Görsel Dikkat-Çekerlik

ACKNOWLEDGEMENTS

First and foremost, I would like to state my special thanks to my supervisor Asst. Prof. Dr. Ufuk Çelikcan for his priceless support and guidance during my graduate study.

Besides, I am grateful to Prof. Dr. Tolga Çapın who is one of my thesis committee members and supported me with his knowledge during the research project that this thesis emerge from.

I also would like to thank my other thesis committee members Assoc. Prof. Dr. Tunca Doğan and Assoc. Prof. Dr. Aydın Kaya for reviewing my thesis elaborately and giving inspirational comments.

I would thank Alper Özkan and Berk Cebeci who are members of Hacettepe University Computer Graphics and Game Studies Laboratory (HUCGLAB) for their help.

Finally, I would like to thank my family and girlfriend, Dilan, for their moral support whenever I feel stuck throughout my graduate studies.

This work is supported by the Scientific and Technical Research Council of Turkey (TUBITAK, project grant 116E280).

CONTENTS

Page

ABSTRACT i
ÖZET iii
ACKNOWLEDGEMENTS v
CONTENTS vi
TABLES ix
FIGURES x
ABBREVIATIONS xii
1. INTRODUCTION 1
2. PREVIOUS WORK 4
3. EVALUATED METHODS 7
3.1. Real-Time Saliency Detection Methods 7
3.1.1. RGB Saliency Detection Methods 8
Boolean Map Saliency (BMS) 8
Fast, Accurate, and Size-Aware Salient Object Detection (FASA) 8
Fine-Grained Saliency (FGS) 8
Minimum Barrier Salient Object Detection (MB, MB+) 9
3.1.2. RGB-D Saliency Detection Methods 9
Anisotropic Center-Surround Difference (ACSD) 9
Local Background Enclosure for RGB-D Salient Object Detection (LBE) .. 10
3.2. Non-Real-Time Saliency Detection Methods 10
3.2.1. Heuristic-Based Saliency Detection Methods 10
3.2.1.1. Image Saliency Detection Methods 10
Center-Dark Channel Prior for Salient Object Detection (CDCP) 11
Multilayer Backpropagation Saliency Detection (MBSD) 11
3.2.1.2. Video Saliency Detection Methods 12
Video Saliency Detection via Spatial-Temporal Fusion and Low-Rank Coherency Diffusion (SLC) 12

Depth-Transformed Video Saliency Detection via Spatial-Temporal Fusion and Low-Rank Coherency Diffusion (DSLC) 13
3.2.2. Learning-Based Saliency Detection Methods 13
Deep Depth-Depurator Network (D^3Net) 13
Joint Learning and Densely-Cooperative Fusion (JL-DCF) 14
Depth-Induced Multi-scale Recurrent Attention Network (DMRANet) 15
4. METHODOLOGY 16
4.1. User Study Design 16
4.1.1. Subjects 16
4.1.2. Hardware 16
4.1.3. Virtual Environments 17
4.1.4. Procedure 17
4.2. Dataset 19
4.3. Experiments 19
4.3.1. Real-Time Saliency Detection Methods 19
4.3.2. Non-Real-Time Saliency Detection Methods 20
5. RESULTS 22
5.1. Viewing Behaviors in 2D and 3D Viewing Conditions 22
5.2. Results of Real-Time Saliency Detection Methods 25
5.2.1. Metric Scores 25
5.2.2. Frame Rates 26
5.3. Results of Non-Real-Time Saliency Detection Methods 27
5.3.1. Metric Scores 27
5.3.2. Computation Times 28
6. DISCUSSION 32
6.1. Viewing Behaviors 32
6.2. Metric Scores 32
6.2.1. Real-Time Methods 33
6.2.2. Non-Real-Time Methods 34
6.3. Run-Time Analysis 36

6.3.1. Real-Time Methods 36
6.3.2. Non-Real-Time Methods 36
6.4. Limitations and Future Work 37
7. CONCLUSION 39

TABLES

Page

Table 3.1 Overview of the saliency prediction methods that are evaluated in this study. 7

Table 5.1 Average scores of the real-time saliency detection methods for desktop and VR displaying situations. The scores shown below are truncated to two decimal places for illustrative purposes. 26

Table 5.2 Average scores of the non-real-time saliency detection methods for desktop and VR displaying situations. The scores shown below are truncated to two decimal places for illustrative purposes. 28

FIGURES

Page

Figure 4.1 Sample frames from the VEs. 18

Figure 5.1 Combined GTs in heatmap forms of all frames from (a,b) abstract VE, (c,d) indoor VE, (e,f) outdoor VE in 2D (left column) and 3D (right column) viewing conditions. The maps are scaled to 0-255..... 23

Figure 5.2 The GTs of the images which have the most different viewing behavior between desktop and VR displaying situations of the abstract (first row), indoor (second row), and outdoor (third row) VEs. 24

Figure 5.3 The GTs of the images which have the least different viewing behavior between desktop and VR displaying situations of the abstract (first row), indoor (second row), and outdoor (third row) VEs. 25

Figure 5.4 Recorded frame rates of the real-time saliency detection methods in the VEs with 2D and 3D viewing conditions. The values are represented in frame per second (FPS). Vertical lines present the minimum and maximum FPS values recorded during the sessions while dots give the mean FPS value in the session. 27

Figure 5.5 Scores obtained by the evaluated methods are presented as strip-plots that give the range and distribution of scores, s.t., strip becomes more transparent where attained scores are sparser. The vertical axis in each plot is specific to the range achieved by the respective metric, while the scores are grouped for the 2D and 3D viewing conditions on the left and the right, respectively. Diamonds mark the corresponding means..... 29

Figure 5.6 Saliency maps (in the latter 10 rows) of sample frames (in the 1st row) from the dataset along with their corresponding depth maps, 2D and 3D ground truths (in the 2nd, 3rd and 4th rows, respectively). The first, the second and the third pair of frames belong to abstract, indoor and outdoor VEs, respectively. 30

Figure 5.7 Average per frame saliency map generation times in seconds of (a) heuristic-based methods and (b) learning-based methods. 31

ABBREVIATIONS

2D	:	**2**-**D**imensional
3D	:	**3**-**D**imensional
ACSD	:	**A**nisotropic **C**enter-**S**urround **D**ifference
AR	:	**A**ugmented **R**eality
AUC	:	**A**rea **U**nder the **C**urve
BMS	:	**B**oolean **M**ap **S**aliency
CC	:	Pearson's Linear Coefficient
CDCP	:	**C**enter-**D**ark **C**hannel **P**rior for Salient Object Detection
CNN	:	**C**onvolutional **N**eural **N**etwork
ConvLSTM	:	**Conv**olutional **L**ong **S**hort-**T**erm **M**emory
D^{3}Net	:	**D**eep **D**epth-**D**epurator **Net**work
DCF	:	**D**ensely-**C**ooperative **F**usion
DDU	:	**D**epth-**D**epurator **U**nit
DMRANet	:	**D**epth-Induced **M**ulti-Scale **R**ecurrent **A**ttention **Net**work
DSLC	:	**D**epth-Transformed Video Saliency Detection via **S**patial-Temporal Fusion and **L**ow-Rank**C**oherency Diffusion
FA	:	**F**eature **A**ggregation
FASA	:	**F**ast, **A**ccurate, and **S**ize-**A**ware Salient Object Detection
FGS	:	**F**ine-**G**rained **S**aliency
FLM	:	**F**eature **L**earning **M**odule
FPN	:	**F**eature **P**yramid **N**etwork
FPS	:	**F**rame **P**er **S**econd
GT	:	**G**round **T**ruth
HMD	:	**H**ead-**M**ounted-**D**isplay
HVS	:	**H**uman **V**isual **S**ystem
JL	:	**J**oint **L**earning

JL-DCF	:	**J**oint **L**earning and **D**ensely-**C**ooperative **F**usion
KL	:	**K**ullback-**L**iebler Divergence
LBE	:	**L**ocal **B**ackground **E**nclosure for RGB-D Salient Object Detection
MB, MB+	:	**M**inimum **B**arrier Salient Object Detection
MBD	:	**M**inimum **B**arrier **D**istance
MBSD	:	**M**ultilayer **B**ackpropagation **S**aliency **D**etection
NSS	:	**N**ormalized **S**canpath **S**aliency
ReLU	:	**Re**ctified **L**inear **U**nit
SIM	:	**Sim**ilarity
SLC	:	Video Saliency Detection via **S**patial-Temporal Fusion and **L**ow-Rank **C**oherency Diffusion
VAM	:	**V**isual **A**ttention **M**echanism
VE	:	**V**irtual **E**nvironment
VR	:	**V**irtual **R**eality

1. INTRODUCTION

In addition to providing color vision, another crucial function of *human visual system* (HVS) is to maintain depth perception and thus creating a 3-dimensional (3D) reconstruction from 2-dimensional (2D) projections of the world by means of binocular vision, so that humans can perceive their surroundings in 3D. On the other hand, humans do not perceive each object at the same level of importance. HVS makes use of *visual attention mechanism* (VAM) to label the most attention-grabbing object in the view resulting in eyes shifting the focus on that object. As a result of this process, the view is blurred except for the periphery of the focus region, providing comfortable viewing. VAM exploits different features such as color, depth, the distances between objects, movements etc., while detecting the most attention-grabbing object.

In computer science, VAM is studied generally as part of the visual saliency prediction problem. Saliency prediction approaches can be mainly divided into 2 groups as heuristic-based and learning-based [1]. Earlier studies [2–4] generally focused on traditional heuristic-based approaches while more recent studies [5–7] proposed learning-based ones, thanks to increasing computational power and advances in machine learning. According to the medium they target, saliency prediction approaches can also be grouped as image saliency methods [8], video saliency methods [9], and co-saliency methods [10]. Regardless of the medium, a saliency method must take advantage of the extracted features obtained from feature maps and/or saliency cues found in inputs to detect salient regions. While edges, lines, corners and blobs are among the most widely used features in the learning-based approaches [11], saliency cues such as contrast prior, motion, and depth are generally exploited in the heuristic-based approaches. Saliency detection methods may adopt these features or cues either individually or collaboratively. Specifically, depth has proven to be a very valuable cue in general with its flexibility to be used with other cues or features in both heuristic-based [12, 13] and learning-based methods [14, 15].

Today, a very prominent field to take advantage of VAM is virtual reality (VR). Although,

the beginning of VR dates back to 1960s, it became more popular with the emergence of modern head-mounted displays (HMDs) in the past decade. These devices mimic HVS with the aim of presenting realistic 3D viewing while providing a comfortable experience as much as possible. VR simulations that are experienced with HMDs have been extensively utilized for medical [16] and educational [17] purposes. Despite the recent advances in VR hardware and software, serious challenges still arise while developing such systems. Cybersickness is one such issue that is closely related to visual discomfort experienced with VR [18]. Cebeci et al. [19] investigated the emotional and physiological effects of different types of content in VR. Their results showed that the users feel cybersickness even if they experience a virtual environment (VE) which is created with the aim of minimizing the cybersickness symptoms. Similar to how HVS exploits VAM to compose a comfortable viewing in the real world, VR technology can benefit from saliency to compose a realistic and comfortable 3D viewing by alleviating the cybersickness factors in VR environments.

To be able to use saliency information effectively in immersive VR environments, its potential with VR must be explored explicitly. Most VR systems present dynamic interactive experiences to their users in VEs. The nature of such interactive VEs makes real-time requirements in VR inevitable because these systems are highly responsive and delays are intolerable. In this context, the situation of the fast saliency detection approaches in VR must be revealed. Although speed is an important factor in VR applications, VR is not restricted to applications rendered in real-time and focusing on this exclusively urges to overlook other important aspects of saliency prediction in VR.

In particular, depth information plays a crucial role in VR since a vital concern is to present realistic 3D viewings in VEs. However, virtually all previous work analyzing VR saliency has been carried on the domain of 360-degree images and videos and, according to the our survey, none of these studies on 360-degree contents made an in-depth investigation of the contribution of depth information to saliency prediction in VR. Aside from depth, another important aspect is that most VR experiences have similarities with traditional videos comprising sequences of interdependent frames shot in scenes that are either generated synthetically or captured from real life. Thus, application of video saliency to immersive VR

is another aspect that needs to be addressed. Furthermore, as convolutional neural networks (CNNs) continue to make remarkable progress in the deep learning, learning-based RGB-D image saliency prediction approaches have been taking traditional ones' place. This leads to surmising the usability of the RGB-D image saliency detection methods, which employ learning processes, in VR environments besides the heuristic-based ones.

Revealing the current landscape of the saliency detection approaches in VR from various perspectives would facilitate a more thorough understanding and, in consequence, more effective use of visual saliency in VEs. With this aim, addressing the aforementioned facets that are mostly omitted in the existing body of research on VR visual saliency, this work presents a comparative analysis of different saliency prediction approaches in VR as 2 parts. In the first part, 7 real-time heuristic-based saliency detection approaches (5 RGB, 2 RGB-D) in VR exploiting different saliency cues are examined comprehensively. In the second part, 3 deep learning-based and 4 heuristic-based approaches for offline prediction of visual saliency in VR are examined. Also, 2 of the heuristic-based methods work on the video domain in order to extend the analysis with an evaluation of video saliency against image saliency in VR. Due to the particular importance of depth in VR, all 5 image saliency prediction methods in this part work on RGB-D images, i.e., make use of depth cues in addition to 2D image cues. Moreover, while one of the video saliency prediction approaches takes traditional RGB videos as input, the other one is a depth-infused version of the former.

The rest of the paper is organized as follows. In Section 2., we discuss the previous work on visual saliency prediction in VR. Then, in Section 3., we detail the mechanisms of the visual saliency prediction methods evaluated in this work. Section 4. outlines the experimental procedure and the dataset used with it for our evaluation. In Section 5., the results obtained from the experiments are presented with different saliency evaluation metrics along with a run-time analysis. Finally, Sections 6. and 7. give a discussion of our findings and conclude the paper.

2. PREVIOUS WORK

Despite several studies striving to explore saliency in immersive VR, the current state of the literature remains insufficient to reveal its full potential. Recently, Albayrak et al. [20] evaluated real-time heuristic-based image saliency detection methods with a dataset containing samples from both desktop (henceforth abbreviated as 2D) and immersive VR (henceforth abbreviated as 3D) viewing conditions. As the continuation of that study and as a part of this thesis work, Celikcan et al. [21] conducted a more extensive user study by enlarging the dataset with more participants and reducing navigation speeds. They gave an extensive analysis of visual saliency in VR via a comparison of user behaviors between the 2D and 3D viewing conditions. Although they examined different saliency cues including the depth in their study, they focused on revealing the potential of real-time saliency detection methods in immersive VR. Again as a part of this thesis work, Askin et al. [22] examined the prediction performances and computational costs of seven offline saliency detection methods in desktop viewing and VR viewing with an HMD with the same dataset in [21]. Their purpose was to reveal the full potential of the depth cue in saliency detection of VEs without another consideration such as real-timeliness etc. In another study by Marmitt et al. [23], an attentional model was evaluated with two novel temporal analysis methods by comparing the human and predicted scan paths in the immersive VR, and the results showed that the model could not predict the scan paths correctly.

Hu et al.'s gaze analysis [24] showed that the gaze positions are closely related to rotation speeds, salient regions, and the placement of dynamic scene contents in the static and dynamic immersive VR environments. Rivu et al. [25] discovered that manipulating the saliency in VR using the color cue does not have significant effect on visual attention if specific tasks are assigned to users while experiencing the scenes. Further, Alexiou et al. [26] carried out a gaze-tracking experiment in the immersive environments adopting a 6 degrees of freedom task-driven approach.

Besides immersive VR, there have been studies where visual saliency has been examined

only in 2D viewing of VEs. In one such study, Oyekoya et al. [27] suggested a visual attention model benefiting from the human visual attention in VEs to animate the gaze of avatars realistically. In another, McDonnell et al. [28] carried out a user experiment to extract eye-catching body parts of human animations in crowded VEs.

Interestingly, the saliency of 360 VR images/videos has been studied much more extensively than the rendered interactable VEs. There have been many studies on predicting fixations or head trajectories for static or dynamic 360 VR images/videos or proposing new saliency models for them [29–31], even with the deep learning approaches [32–34]. There have also been efforts focusing on unveiling the underlying attention mechanism while viewing such 360 VR images/videos [35–37]. Yet, the depth cue has not been studied in the saliency of 360 VR images/videos in detail, albeit it is a crucial factor not only for immersive VR but also for all VR applications.

Although the efforts towards analyzing modern visual saliency prediction methods within rendered interactable VEs to acquire a thorough understanding of VAM in VR stand on a limited extent, saliency information has been utilized in various VR-targeted applications. A prominent use of saliency emerges in the enhancement of stereoscopic viewing experience by manipulating stereoscopic rendering parameters even though different techniques for adjustment of stereoscopic rendering parameters are proposed in the literature with the same purpose [38–40]. To this end, Celikcan et al. [41] achieved better visual comfort, depth perception, and image quality in interactable VEs against two alternative approaches by optimizing disparity based on extracted attention features. Similarly, Yoon et al. [42] manipulated stereo camera parameters under the guidance of saliency maps in desktop VR scenes, and their method outperformed the default settings with regard to visual comfort and depth perception. Xia et al. [43] configured depth of stereoscopic images in desktop VR utilizing optimum zero-disparity plane position extracted from RGB-D saliency maps, resulting in improved depth perception and visual comfort. Similarly, Shao et al. [44] generated discomfort maps using saliency maps to finetune the zero-disparity plane, resulting in a decrease in visual discomfort of stereoscopic images in desktop VR while preserving depth sensation.

The use of saliency for improving VR experience and/or imagery has not been limited to optimization of stereoscopic rendering parameters [45–47]. Nie et al. [48] reported a reduction of experienced motion sickness at the rate of %50 in immersive VR by blurring non-salient regions in the scenes. Shi et al. [49] corrected the shifted gaze points of the eye-trackers integrated to an HMD using 2D drift vectors estimated by extracting saliency features. Sung et al. [50] implemented a rendering algorithm for VR applications, in which selective anti-aliasing is applied with saliency map values. They reported that the outputs were not distinguishable from the full anti-aliasing outputs, and provided a reduction in rendering time. Finally, Poreddy et al. [51] proposed a robust and competitive 3D VR image quality evaluation method, where the attention maps form the basis of their study.

3. EVALUATED METHODS

Table 3.1 gives an overview of the saliency prediction methods evaluated in this work. These methods can be basically categorized as two groups in regard to if they run in real-time (first part) or non-real-time (second part). Although there are a lot of saliency detection methods proposed in the literature for each category, the methods that have high prediction performances among the most recent ones whose implementations are publicly available were chosen for this study. Furthermore, a special care was taken to ensure that the methods used were architecturally different in each part. In the following, the details regarding their extended taxonomy and the mechanisms they use for saliency prediction are given.

3.1. Real-Time Saliency Detection Methods

All methods examined in this section are heuristic-based image saliency detection approaches.

Paper	Method Name	Response Time	Prediction Approach	Targeted Domain	Input Type	Depth Usage
Zhang et al. [52]	Boolean Map Saliency (BMS)	Real-Time	Heuristic-based	Image Saliency	RGB	None
Yildirim et al. [53]	Fast, Accurate, and Size-Aware Salient Object Detection (FASA)	Real-Time	Heuristic-based	Image Saliency	RGB	None
Montabone et al. [54]	Fine-Grained Saliency (FGS)	Real-Time	Heuristic-based	Image Saliency	RGB	None
Zhang et al. [55]	Minimum Barrier Salient Object Detection (MB, MB+)	Real-Time	Heuristic-based	Image Saliency	RGB	None
Ju et al. [12]	Anisotropic Center-Surround Difference (ACSD)	Real-Time	Heuristic-based	Image Saliency	RGB-D	Directly
Feng et al. [13]	Local Background Enclosure for RGB-D Salient Object Detection (LBE)	Real-Time	Heuristic-based	Image Saliency	RGB-D	Directly
Zhu et al. [56]	Center-Dark Channel Prior for Salient Object Detection (CDCP)	Non-Real-Time	Heuristic-based	Image Saliency	RGB-D	Directly
Zhu et al. [57]	Multilayer Backpropagation Saliency Detection (MBSD)	Non-Real-Time	Heuristic-based	Image Saliency	RGB-D	Directly
Chen et al. [58]	Video Saliency Detection via Spatial-Temporal Fusion and Low-Rank Coherency Diffusion (SLC)	Non-Real-Time	Heuristic-based	Video Saliency	RGB	None
Cong et al. [59]	Depth-Transformed Video Saliency Detection via Spatial Temporal Fusion and Low-Rank Coherency Diffusion (DSLC)	Non-Real-Time	Heuristic-based	Video Saliency	RGB-D	Infused
Fan et al. [60]	Deep Depth-Depurator Network (D^3Net)	Non-Real-Time	Learning-based	Image Saliency	RGB-D	Directly
Fu et al. [61]	Joint Learning and Densely-Cooperative Fusion (JL-DCF)	Non-Real-Time	Learning-based	Image Saliency	RGB-D	Directly
Piao et al. [62]	Depth-Induced Multi-scale Recurrent Attention Network (DMRANet)	Non-Real-Time	Learning-based	Image Saliency	RGB-D	Directly

Table 3.1 Overview of the saliency prediction methods that are evaluated in this study.

3.1.1. RGB Saliency Detection Methods

Boolean Map Saliency (BMS) Zhang et al.'s BMS method [52] mainly adopts the surroundedness cue and exploits Boolean map theory for visual saliency [63]. The surroundedness cue assumes that objects are clearly distinguishable from their backgrounds. According to this assumption, a surrounded area possibly enclose an object. A frame is symbolised as a collection of Boolean maps, that are constructed by arbitrarily thresholded feature maps of the frame, in BMS . An initial attention map is computed considering the surroundedness cue for every Boolean map. Then, these maps are fused linearly and construct an intermediate saliency map. Finally, the output is computed after post-processing of the intermediate saliency map.

Fast, Accurate, and Size-Aware Salient Object Detection (FASA) In the method of Yildirim et al. [53], the pixel-wise saliency values on an image are fused with maps representing global contrast values. Fast nature of this method comes from image simplification by using a color quantization in color space of CIE L*a*b. After the simplification, the basic features of attention-grabbing elements are detected by using the variances of the quantized colors and spatial center. Saliency probabilities are calculated by using the features detected in the earlier step. On the other hand, global contrast values are found by means of the differences between the quantized colors. The output attention map is calculated as a consequence of linear interpolation of the global contrast values and the saliency probabilities.

Fine-Grained Saliency (FGS) This method is proposed and used in the study aiming to detect humans with a mobile platform in real-time by Montabone et al. [54]. The FGS method is mainly based upon center-surround differences cue. Because human detection is realized in real-time and the FGS method is exploited to help detection, visual saliency detection is also realized in real-time. Therefore, this makes the method valuable for interactive VR systems, where noticeable interruptions are intolerable. The features extracted

are determined by taking inspiration from the biological attention system [4, 64] and the feature extraction is made at the original input resolution with the help of an integral image, which is given in [65]. To this end, a Gaussian filter is utilized for smoothing the input at first. Thereafter, making use of the integral image, on- and off-center differences are computed.

Minimum Barrier Salient Object Detection (MB, MB+) In the same study, Zhang et al. [55] came up with two different real-time saliency detection approaches called MB and MB+. According to the cue named image boundary connectivity, there is a connection between the background and the borders of an image, and these methods are built on this opinion. For obtaining the image boundary connectivity of the pixels in an image, a modified minimum barrier distance (MBD) algorithm, whose speed is 100 times higher than the original implementation, is used. The outputs of this operation are the MBD maps of the corresponding image for different channels. These MBD maps are further fused by pixel-wise addition to get a single MBD map for the image. A number of post-processing steps are applied to the fused map to enhance the salient object segmentation. The output of this post-processing step is the finalized saliency map of MB. Apart from MB, the backgroundness cue is employed besides the image boundary connectivity cue to construct the outputs of MB+. According to the backgroundness cue, the image background is connected to the image boundary as well as similar to it.

3.1.2. RGB-D Saliency Detection Methods

Anisotropic Center-Surround Difference (ACSD) Ju et al. [12] implemented a method to detect visual saliency of images by employing depth cue. The method employs depth contrast, which is calculated as the ACSD value at the centroid of every super-pixel by means of depth maps. To calculate the ACSD value of a super-pixel, 8 different scanlines are determined for each super-pixel and the difference between the deepest point and central point across the scanline is determined as the ACSD of the corresponding scanline. Then, the sum of ACSD values of 8 different scanlines is accepted as the ACSD value of

the corresponding super-pixel. By using the scaled values of 0-255, saliency values are determined for each pixel. As a post-processing step, linear weighting is applied to get the regions close to the observer more salient. Because the objects closer to the center have a tendency to be more attention-grabbing [66, 67], a Gaussian filter is applied to the output attention map.

Local Background Enclosure for RGB-D Salient Object Detection (LBE) Feng et al. [13] came up with a new cue named local background enclosure (LBE) instead of depth contrast, unlike most studies to predict saliency of RGB-D images. They indicated that employing depth contrast may cause false positives because the depth of the background is very changeable in a frame. According to LBE, attention-grabbing objects are located in front of their enclosures. Namely, the salient objects in an image have a lower depth than their surroundings. To identify the object boundaries, LBE makes use of angular density and depth components. Super-pixel segmentation is applied to the image before calculating LBE. After calculating LBE, depth, spatial, and background priors are realized. In the end, the final saliency map is acquired through a refinement with Grabcut segmentation [68].

3.2. Non-Real-Time Saliency Detection Methods

3.2.1. Heuristic-Based Saliency Detection Methods

The heuristic-based methods under consideration here employ unsupervised techniques that exploit low-level cues to extract saliency information. These methods further fall into two main categories with respect to the domain they work on.

3.2.1.1. Image Saliency Detection Methods

The methods that employ depth information with traditional approaches process each image individually using various cues besides depth to locate the visually conspicuous parts of each image. Here we consider two RGB-D methods in particular. Both of them employ center

bias, which is shown to have a crucial role in visual saliency prediction of VEs [37]. They are also among the most recent approaches that handles RGB-D saliency in a traditional way and they present better prediction performance than the studies preceding them.

Center-Dark Channel Prior for Salient Object Detection (CDCP) In this method, Zhu et al. [56] employed a new cue named center-dark channel in addition to depth and color cues to strengthen the robustness of saliency detection. At first, the original images and depth maps are separated into a number of regions using K-means clustering. Saliency values of each region are calculated considering spatial region weights and inter-region Euclidean distances. Another weight, which is derived from center bias and depth information, is assigned to each region of depth and original images, and an initial saliency map is constructed by adding the corresponding weighted region-based image and depth saliency values. Also, single-layer cellular automata algorithm [69] is employed to optimize background-based map and a center attention map is acquired by using it. Based on the idea that the foreground and background have different transmissivities, a dark channel saliency map is created. Then, this map is merged in the center saliency map. Finally, a further enhanced fusion of the initial saliency, depth and, center-dark saliency maps is made to get the output saliency map.

Multilayer Backpropagation Saliency Detection (MBSD) Zhu et al. [57] suggested this RGB-D image saliency prediction approach employing the depth cue in 3 different layers. In the first layer, a preliminary attention map is constructed similarly to the initial saliency map in CDCP and then it is refined by the depth and center saliency maps. In the second layer, an extended map is acquired by multiplying the depth map and original image, and this replaces the original image. An intermediate attention map is obtained similarly to the initial attention map in CDCP, and this map is refined with the refined preliminary saliency map from the first layer and the depth information. In the last layer, the depth map is filtered to cut off the background noise and a reprocessed map is generated similarly to the second layer by using the filtered depth map instead of the original depth map. This reprocessed map

replaces the original image, and a saliency map is calculated similarly to the second layer. By refining this map with the refined preliminary and intermediate saliency maps from the previous layers, the output saliency map is constructed.

3.2.1.2. Video Saliency Detection Methods

The aim of video saliency prediction is localizing movement-related attention-grabbing objects in a frame sequence by evaluating them individually as well as collectively to uncover the temporal relationships among the interrelated frames. By doing this, these methods can use both spatial and temporal information in frames. VEs have similarities with traditional videos in this regard and the motion cue may play a critical role besides the spatial cues for saliency detection in VEs.

For our evaluation, we use an RGB video saliency prediction method that employs the fusion of motion and color cues. The method detects saliency in a batch-wise manner. Furthermore, we also use its depth infused version which applies a depth transformation model to its outputs and use this transformed model as an RGB-D video saliency prediction method.

Video Saliency Detection via Spatial-Temporal Fusion and Low-Rank Coherency Diffusion (SLC) In their method, Chen et al. [58] processed the equally divided frame batches of a video rather than dealing with it as a whole or frame-by-frame basis in order to mitigate error accumulations. To increase accuracy, motion saliency of each frame obtained from the spatial-temporal gradient map is further combined with the corresponding color saliency, and refined by intra-batch and inter-batch indicators, to boost motion saliency. They used a new scheme called low-rank coherency guided saliency diffusion and boosting instead of the traditional spatial-temporal smoothing to eliminate inaccurately detected regions better throughout a batch, while maintaining temporal smoothness. Also, the last frame of each batch is set as the first frame of the next one so that the diffusion of the batch-level temporal prior of the current frame to all frames is ensured.

Depth-Transformed Video Saliency Detection via Spatial-Temporal Fusion and Low-Rank Coherency Diffusion (DSLC) Although the heuristic-based RGB-D saliency detection has a successful record, all existing methods focus on developing the models from scratch. However, Cong et al. [59] showed the transferability of existing RGB saliency detection methods with a 3-stage depth transformation model. Exploiting the depth cue, the local geodesic and global compactness cues are calculated, and then the RGB-D saliency initialization is realized by combining the RGB saliency from an external method with these cues. Finally, depth-based refinement and optimization are carried out considering the quality of the depth map and depth constraints. As stated previously, the saliency maps from SLC are fed into the transformation model, and the final maps are evaluated as RGB-D video saliency detection results.

3.2.2. Learning-Based Saliency Detection Methods

As opposed to traditional methods, learning-based methods adopt data-driven techniques commonly using high-level features. Following its emergence, the Visual Geometry Group's VGG CNN architecture [70] has been quickly adopted for use with new deep learning-based (DL-based) RGB-D saliency prediction approaches due to reduction in training/testing times with smaller kernel size and improvement in precision with deeper layers. This prompted an outstanding acceleration in the advancement of DL-based RGB-D saliency prediction approaches.

In our evaluation, we handle three modern DL-based RGB-D saliency prediction approaches adopting VGG CNN. Even though they were trained on relatively small datasets, they excel in terms of both speed and prediction performances in comparison to the the former models, as the backbone. While they use the same model as the backbone, these methods are architecturally different from each other as detailed below.

Deep Depth-Depurator Network (D^3Net) D^3Net by Fan et al. [60] adapts the efficient pyramidal feature extraction mechanism of feature pyramid networks (FPNs) [71] to image

saliency detection with a feature learning module (FLM), which consists of three identical subnetworks that model RGB, depth, and RGB-D saliency separately. Early-fusion strategy is adopted to combine the depth and RGB information in the RGB-D subnetwork. FPNs are independent of the backbone, so a modified VGG-16 architecture is used in each subnetwork to simply extract the features in a pyramidal manner. Assuming the observation that the deeper layers maintain more semantic information, the sixth layer is added to the original VGG-16 architecture, and fine and coarse feature maps are merged with concatenation instead of addition, unlike the original FPN. In addition to the FLM, depth depurator unit (DDU), which filtrates the images that hold insignificant depth information in the test phase, is introduced. When there is a high level of similarity between outputs of the depth and RGB-D subnetworks, this points to a rich depth map and DDU determines the output of the RGB-D subnetwork as the predicted final attention map. Otherwise, the depth information is discarded, and the predicted final attention map becomes the output of the RGB subnetwork.

Joint Learning and Densely-Cooperative Fusion (JL-DCF) Another effective way of discovering the commonality between the depth and RGB cues in image saliency prediction is making use of the Siamese network as it enables an easy cross-modal information transmission while reducing the parameters via a shared backbone [61]. In the JL-DCF method, a joint learning (JL) component adopting a Siamese network with middle-fusion is proposed. This module is independent of the backbone and uses a modified VGG-16 architecture. Initially, the depth and RGB inputs are concatenated in the fourth dimension, instead of the third, and then fed into the network, where side-output features are generated from each layer. To ensure the dimensional consistency among the side-output features, each of them is compressed to an identical size through convolutional layers and rectified linear units (ReLUs). After this, the densely cooperative fusion (DCF) module steps in and combines the depth and RGB features in every side-output feature. Finally, the fused side-output features are aggregated by feature aggregation modules (FAs) such that the inputs of each FA are the FAs of deeper layers and the fused side-output feature of that layer.

Depth-Induced Multi-scale Recurrent Attention Network (DMRANet) Piao et al. [62] designed a two-stream model consisting of two identical subnetworks adopting the VGG-19 as the backbone for depth and RGB cues. First of all, side-output features of both networks from each layer are sent to the depth refinement block, where the depth feature traverses a couple of weight layers before getting merged in the RGB feature. The finalization of features in each layer is made via processing the combined feature in a residual unit, and then the features of the layers are combined through element-wise addition. Besides, for the purpose of learning a depth vector, the last side-output feature of the depth subnetwork is further processed. Thereafter, the fused feature and depth vector meet in the depth-induced multi-scale weighting module, where the fused feature is separated into multi-scale features through a couple of parallel convolutional layers and global pooling. Here, for weighting the multi-scale features, the depth vector is used , and the weighted features are fused again. The purpose of this operation is to locate salient objects accurately. Lastly, inspiring by the internal generative mechanism of HVS, a recurrent attention module including many convolutional and Convolutional Long Short-term Memory (ConvLSTM) blocks is employed to boost the method's performance by exploiting the previous memory to make the current decision.

4. METHODOLOGY

4.1. User Study Design

To compose the basis of the dataset, the fixation data was gathered from the users throughout a user study in desktop and VR viewing conditions separately via eye trackers. We give a detailed explanation of the user study in this section.

4.1.1. Subjects

Upon an online call, completely volunteer 13 female and 17 male participants attended the data collection without any financial or other incentive promotion. They had normal or corrected-to-normal vision and the average age and standard deviation were 27 and 4.07, respectively. Their previous VR experiences were asked via a 5-point Likert scale where 0 and 4 points correspond to “no experience” and ”highly experienced”, respectively. According to the questionnaire results, the 14 participants had no previous VR experience and the average experience was 0.9 ± 1.09.

4.1.2. Hardware

For the 2D viewing condition, a laptop equipped with an integrated Tobii Aware eye tracker, Intel® Core™ i7-7700HQ 2.8 GHz quad-core CPU, 17-inch display, Nvidia GeForce GTX 1070 was used.

For the 3D viewing condition, an HTC Vive HMD containing Tobii Pro eye trackers and a desktop computer equipped with Intel® Core™ i7-6700 3.4 GHz quad-core CPU, 16 GB RAM and, an Nvidia GeForce GTX 1050 were used.

4.1.3. Virtual Environments

Three different VEs constitute the experimental structure of the user study which is implemented by using the Unity game engine. The gaze data was collected throughout the sessions in these VEs for both viewing conditions. Fig. 4.1 gives a general idea about the VEs, and their characteristics are as follows.

The first VE (henceforth, the abstract VE) resembles a small labyrinth having three connected narrow corridors including plain 3D masses spreading around. The VE is shown for 195 seconds along a camera path through the corridors. The second VE (henceforth, the indoor VE) is a classroom, in which stable and dynamic animated humans exist among tables, chairs, desktop computers, etc. This VE is shown for 54 seconds along a path traversing the classroom. The last VE (henceforth, the outdoor VE) is a street, where trees, houses, rubbish bins, mailboxes, etc. extend across both sides of it. While the user experiences various animated humans and cars for 226 seconds, the path follows a straight line till the end of the street, then turns back and moves towards the beginning.

To comprehensively evaluate the gaze behavior of desktop and VR viewing conditions under various settings, these VEs introduce a wide range of content with various features. For example, the abstract VE has more basic objects with high color contrasts so this presents a more distinguishable environment when compared to the other VEs. On the other hand, these VEs offer different levels of depth-of-field, and this characteristic is a key factor for a comparative analysis in such a study which prioritizes the investigation of depth in VR. Beside this, the abstract VE includes only static content while the other two have both dynamic and static content, and this is another important feature to observe its effects on video saliency in which the motion cue plays a critical role.

4.1.4. Procedure

At the beginning of the experiment, before taking the signed consent from the participants, information about the experimental process was given to them, and they asked their questions

(a) Abstract VE (b) Indoor VE (c) Outdoor VE

Figure 4.1 Sample frames from the VEs.

freely if they needed to. However, the purpose of the study was kept secret to prevent biases during the sessions. Subsequently, Monoyer chart, Ishihara, and random dot stereogram tests were applied to participants to assess their visual acuity, color blindness, and stereo blindness, respectively. The ones who could not pass any of the tests were not accepted to participate in the data collection.

For the 2D viewing condition, the gaze data was collected for each frame via a Tobbi Aware eye tracker integrated into the laptop, while the participants were watching the VEs in an uprightly immobilized sitting position facing the screen of the laptop from 50 cm away. The lightness level of the experiment environment was kept around 50 lux during the sessions. In the 3D viewing condition, the gaze data was collected for each frame via a Tobii Pro eye tracker integrated into the HTC Vive HMD, while the participants were experiencing the VEs with the HMD in a sitting position again. At the beginning of the sessions, calibrations were carried out for the users to adjust the eye tracker and HMD-eye tracker pair for 2D and 3D viewing conditions, respectively.

All VEs were shown to the participants in a free-viewing paradigm. For both viewing

conditions, the outdoor VEs were shown after the calibration phases, and the abstract and indoor VEs followed it, respectively. To ensure the full consistency of the gaze data, the VEs used in both viewing conditions were completely identical without any change in camera parameters, speeds, paths, etc., and camera rotations were disabled.

4.2. Dataset

By using the collected gaze data obtained from the user study, a dataset that contains RGB images, depth maps, 2D ground truths (2D GTs), and 3D ground truths (3D GTs) of sequential 2933 abstract, 808 indoor, and 3237 outdoor frames was created. The RGB image and depth map pair of each frame are obtained during a demo session in Unity. The collected gaze data is the fixation coordinates of the participants on each frame for the two different viewing conditions. To create the 2D GTs, blank maps full of zeros are created for each frame in the VEs at first. Then, the Gaussian filters are dropped centering each fixation location. Element-wise additions are applied to the overlapping pixels of the intersecting filters. For the 3D GTs, the fixation coordinates are converted to the 2D screen coordinates such that they point to the same locations of the corresponding frames in the 2D viewing condition, and then a similar approach is followed to create the 3D GTs. The dataset is available on https://graphics.cs.hacettepe.edu.tr/vr_sal/

4.3. Experiments

4.3.1. Real-Time Saliency Detection Methods

For the real-time saliency detection methods evaluated in this study, the saliency maps of each method are obtained through an online demo session of each VE in Unity. The RGB image of each frame is given as input for the RGB saliency detection methods BMS, FASA, FGS, MB, and MB+ and the saliency maps are generated as outputs. For the RGB-D saliency prediction methods ACSD and LBE, beside the RGB frame, its depth map is obtained and given as input and the saliency map of the frame is computed as output. To prevent

interruptions and provide comfortable viewing for the accuracy of the data collection by keeping high frame rates, the saliency computations are separated from the user study, i.e., the data are collected during online sessions without any saliency computations. However, the saliency maps are generated throughout offline sessions on the same VEs by making necessary method calls for each frame.

During the online demo sessions to get saliency maps of the real-time saliency detection methods, a desktop machine equipped with Intel® Core™ i7-6700 3.4 GHz quad-core CPU, 16 GB RAM and, an Nvidia GeForce GTX 1050 was used.

4.3.2. Non-Real-Time Saliency Detection Methods

For the traditional image saliency prediction approaches CDCP and MBSD, the depth map and RGB image pairs of the frames are given as inputs, and their saliency maps are generated as outputs. Differently, the saliency maps of the video saliency prediction method SLC are generated by giving only the RGB images as inputs. In addition to the RGB-D images of the frames, the RGB saliency maps obtained by SLC are the inputs of the depth-transformed video saliency detection method DSLC.

Mirroring the split ratio in D^3Net, the dataset is divided into three subsets as 4655 frames (66.7%) for training, 1162 frames (16.65%) for validation, and 1161 frames (16.63%) for testing samples by choosing them from the VEs randomly while maintaining the same split ratio for each VE. However, the validation subset is included in the training subset in JL-DCF and DMRANet methods, in which no validation set is used (i.e., 5817 frames (83.36%) for training and 1161 frames (16.63%) for testing samples). In the original implementations of the methods, data augmentations are applied to the training data because of the limited scale of the datasets. With similar concern, in this study, the training subsets are augmented by flipping, cropping, rotating, or mirroring the images, sticking to the original implementations.

With the augmented dataset, two different models are trained for 2D and 3D viewing conditions by using the 2D and 3D ground truths respectively for each learning-based saliency detection method without changing their original implementation. Consequently, by feeding the depth map and RGB image pairs in the test subset as inputs into the trained 2D and 3D models, two different saliency map sets for desktop and VR viewing conditions are obtained for each method, in contrast to the heuristic-based saliency detection methods, where only one saliency map set is sufficient per method.

Throughout the experimental process, a desktop computer equipped with Intel Xeon E5-2609V4 1.7 GHz octa-core CPU, 32 GB RAM and, an NVidia GeForce GTX Titan X was used.

5. RESULTS

To achieve a broad understanding of the methods' performances, their evaluation is to be carried out using several metrics that interpret saliency from varying perspectives. For this reason, we chose a set of six metrics: two implementations of area under the curve (AUC-Borji, AUC-Judd), Kullback-Leibler divergence (KL), normalized scanpath saliency (NSS), Pearson's linear coefficient (CC), and similarity (SIM). The evaluation scores were assessed by comparing the generated saliency maps and their corresponding 2D and 3D ground truths under these metrics.

AUC implementations and NSS estimate saliency considering the exact fixation points in salient regions, while CC, KL, and SIM handle saliency map-ground truth pairs as continuous distributions. More specifically, AUC implementations are the area under the curve of true positives against false positives calculated by applying different threshold values to saliency maps. NSS averages the normalized saliency at exact fixation locations. SIM is a simple comparison between two normalized maps as it is the sum of the smallest values at every pixel of two maps. CC simply exhibits the linear correlation of two maps. KL treats two maps as probabilistic distributions and measures the dissimilarity between them when saliency map is used to estimate ground truth. It should be noted that using the same metric set in the evaluation of the methods with the same dataset in both parts allows us to make fair comparisons between the methods used in the first and second parts.

5.1. Viewing Behaviors in 2D and 3D Viewing Conditions

Before giving the aforementioned metric scores and computational performance results of the saliency prediction approaches investigated in this study, we examine the viewing behaviors of the users in desktop and VR viewing conditions and the differences between them in this section.

To get a general idea about the differences between the viewing behaviors of the users in desktop and VR viewing conditions, combined fixations of all frames by using the fixations

of all users in each frame are given as heatmaps in Figure 5.1 for each viewing condition, and for each VE. The higher value of a pixel on the heatmap means that more users have fixations on that pixel than the others. The maps of a viewing condition are combined by pixel-wise summation of all fixation maps of each VE. Then the heatmaps are scaled to 0-255 for illustrative concerns.

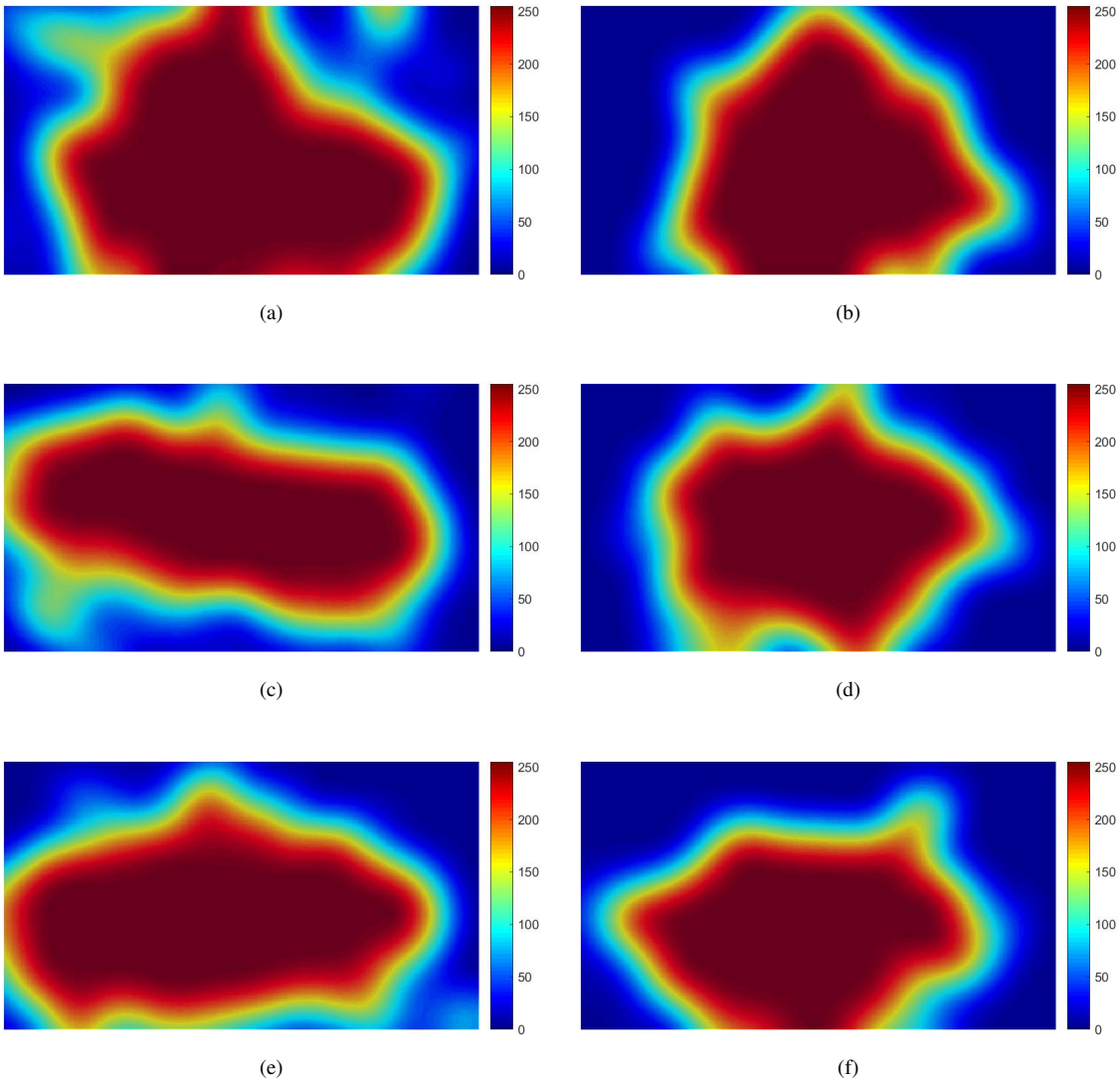

Figure 5.1 Combined GTs in heatmap forms of all frames from (a,b) abstract VE, (c,d) indoor VE, (e,f) outdoor VE in 2D (left column) and 3D (right column) viewing conditions. The maps are scaled to 0-255.

The CC scores between combined fixation maps of desktop viewing and VR viewing conditions for abstract, indoor, and outdoor VEs are calculated as 0.71, 0.56, and 0.69,

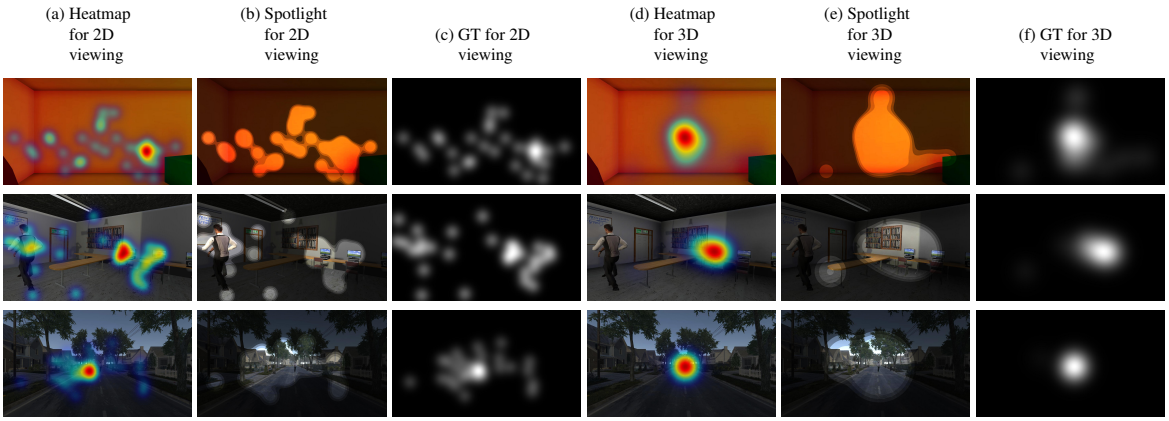

Figure 5.2 The GTs of the images which have the most different viewing behavior between desktop and VR displaying situations of the abstract (first row), indoor (second row), and outdoor (third row) VEs.

respectively. These results point out that the viewing behaviors of users in desktop and VR viewing conditions are mostly similar. However, this similarity is disposed to increase in proportion to the depth of the VE. The results also hint that users experience VEs with a center bias but it is more explicit in 3D viewing condition.

In Figures 5.2 and 5.3, the frames, which have the most different and the least different viewing behaviors of the users between desktop and VR viewing conditions, for every VE are given. The first three columns of the images present the results of 2D viewing conditions while the last three columns are the results of 3D viewing conditions. In each three columns, the first and second columns represent the GTs of the corresponding frames, which are given in the third column, as the overlaid heatmap on the image and spotlight interpretations, respectively. In the spotlight forms, the more visible regions correspond to the denser fixations on those locations, while the darker regions represent the locations that have fewer fixations.

Root-mean-square error values between 2D and 3D GTs of the frames are used to identify the frames in these figures. Hence, the pairs that have the maximum root-mean-square errors are given in Figure 5.2 and the pairs that have the minimum root-mean-square errors are given in Figure 5.3. Regarding the frames in Figure 5.2, the maximum values are calculated as 5.37, 5.69, and 5.51 for abstract, indoor, and outdoor VEs, respectively. Regarding the

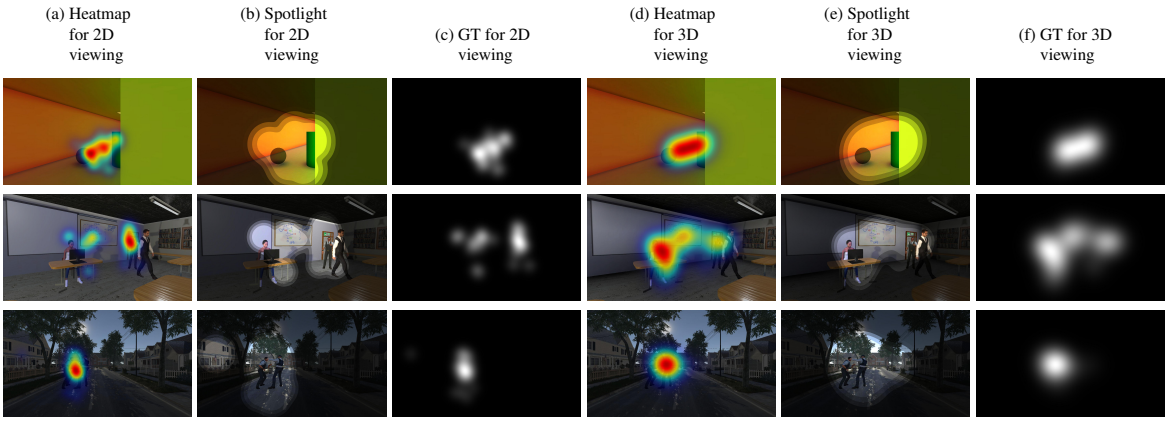

Figure 5.3 The GTs of the images which have the least different viewing behavior between desktop and VR displaying situations of the abstract (first row), indoor (second row), and outdoor (third row) VEs.

frames in Figure 5.3, the minimum values are calculated as 2.53, 3.15, and 2.91 for abstract, indoor, and outdoor VEs, respectively. Also, the average values are calculated as 3.88, 4.38, and 4.05; the median values are calculated as 3.87, 4.33, and 4.06, in the same order.

5.2. Results of Real-Time Saliency Detection Methods

5.2.1. Metric Scores

Table 5.1 exhibits the average metric scores of the real-time saliency detection methods evaluated in this study for desktop and VR viewing conditions. The scores are demonstrated as both the VE-based averages of the frames and the fused averages of all VEs.

In most metric scores, the BMS method draws attention with its better results than the other methods for desktop viewing condition. MB and MB+ methods follow BMS with their scores. By contrast with this general observation, the MB method reaches higher results than the other methods for the abstract VE in SIM, CC, and NSS. Unlike desktop viewing, MB and MB+ methods produced higher results than other methods in 3D viewing condition for the abstract VE and the indoor VE. They also showed the best prediction performances in terms of overall mean scores. However, the BMS method holds its place as the best saliency

predictor among the other methods in 3D viewing conditions for outdoor VE, which offers the biggest average scene depth in comparison with the other two VEs.

	Abstract VE						Indoor VE						Outdoor VE						Overall					
(2D)	AUC - Borji	AUC - Judd	KL	NSS	CC	SIM	AUC - Borji	AUC - Judd	KL	NSS	CC	SIM	AUC - Borji	AUC - Judd	KL	NSS	CC	SIM	AUC - Borji	AUC - Judd	KL	NSS	CC	SIM
BMS	0.54	**0.84**	1.13	0.80	0.49	0.43	0.51	0.74	**1.06**	0.47	**0.35**	**0.37**	0.53	0.67	**1.21**	**0.48**	**0.32**	**0.28**	0.53	0.75	**1.13**	**0.58**	**0.39**	**0.36**
FASA	0.52	0.50	2.33	0.10	0.14	0.23	0.53	0.62	2.11	0.22	0.11	0.28	0.53	0.61	2.09	0.18	0.08	0.20	0.53	0.58	2.18	0.16	0.11	0.24
FG	0.52	0.76	1.62	0.50	0.27	0.32	0.49	0.52	1.93	-0.02	0.01	0.23	0.50	0.48	2.51	-0.07	-0.04	0.17	0.50	0.59	2.02	0.14	0.08	0.24
MB	0.54	0.84	2.11	**0.82**	**0.52**	**0.44**	0.51	**0.78**	1.21	**0.50**	0.29	0.36	0.50	**0.68**	1.90	0.26	0.11	0.24	0.52	**0.77**	1.74	0.53	0.31	0.35
MB+	**0.60**	0.78	**1.04**	0.78	0.47	0.36	**0.54**	0.69	1.06	0.37	0.23	0.31	0.50	0.52	1.55	0.00	0.01	0.20	**0.54**	0.67	1.22	0.38	0.23	0.29
ACSD	0.51	0.42	6.26	-0.07	-0.04	0.17	0.52	0.51	4.31	0.16	0.12	0.26	**0.54**	0.56	6.49	0.19	0.03	0.21	0.52	0.50	5.69	0.10	0.03	0.21
LBE	0.52	0.44	3.56	-0.05	-0.03	0.19	0.52	0.43	2.97	0.03	0.06	0.24	0.48	0.53	3.03	-0.03	-0.05	0.18	0.51	0.47	3.19	-0.02	-0.01	0.20

(3D)	AUC - Borji	AUC - Judd	KL	NSS	CC	SIM	AUC - Borji	AUC - Judd	KL	NSS	CC	SIM	AUC - Borji	AUC - Judd	KL	NSS	CC	SIM	AUC - Borji	AUC - Judd	KL	NSS	CC	SIM
BMS	0.53	**0.83**	1.22	0.59	0.46	0.45	0.51	0.70	0.96	0.29	**0.43**	0.41	0.52	0.68	**1.04**	**0.46**	**0.40**	**0.31**	0.52	0.74	1.07	0.45	**0.43**	0.39
FASA	0.51	0.48	2.09	0.02	0.15	0.26	0.50	0.54	2.52	0.05	0.09	0.30	0.53	0.60	2.05	0.16	0.09	0.22	0.51	0.54	2.22	0.08	0.11	0.26
FG	0.52	0.78	1.34	0.44	0.30	0.36	0.50	0.56	1.63	0.04	0.06	0.27	0.51	0.52	2.27	0.06	-0.02	0.19	0.51	0.62	1.75	0.18	0.11	0.27
MB	0.53	0.83	2.33	**0.60**	**0.53**	**0.48**	0.51	**0.79**	1.09	**0.40**	0.40	**0.43**	0.50	**0.73**	1.60	0.36	0.16	0.27	0.51	**0.78**	1.68	**0.45**	0.36	**0.39**
MB+	**0.57**	0.75	**0.96**	0.56	0.48	0.38	**0.53**	0.67	**0.91**	0.27	0.30	0.36	0.51	0.56	1.31	0.11	0.05	0.22	**0.54**	0.66	**1.06**	0.31	0.28	0.32
ACSD	0.52	0.47	5.26	0.05	-0.02	0.20	0.52	0.50	4.04	0.11	0.11	0.26	**0.55**	0.57	6.84	0.21	-0.02	0.20	0.53	0.51	5.38	0.12	0.02	0.22
LBE	0.51	0.44	2.89	-0.05	-0.04	0.20	0.51	0.43	2.79	0.00	0.04	0.24	0.50	0.59	2.97	0.05	-0.10	0.18	0.51	0.48	2.88	0.00	-0.03	0.21

Colored cells indicate the best scores in each metric.

Table 5.1 Average scores of the real-time saliency detection methods for desktop and VR displaying situations. The scores shown below are truncated to two decimal places for illustrative purposes.

5.2.2. Frame Rates

Due to the different designs they have, the time complexities of the saliency detection methods are different. This results in various frame rates in an interactive VE, when different saliency detection methods are used. Therefore, we evaluate the real-time saliency prediction approaches with regard to frame rate performances when they are used online in our VEs to reveal their convenience for real-time rendering. As mentioned earlier, the user study is conducted through online sessions and the saliency computations are made during the offline sessions. However, no frame rate recording is made either during the user study or during the saliency computations. To this end, we conducted a different test to record the frame rates of each saliency method in our VEs. The test is conducted online in 2D and 3D viewing conditions for each VE by recording the frame rates of each real-time saliency detection method as 42 sessions in total. In each session, the corresponding method is called in every 30 frames and the frame rate of each frame is computed.

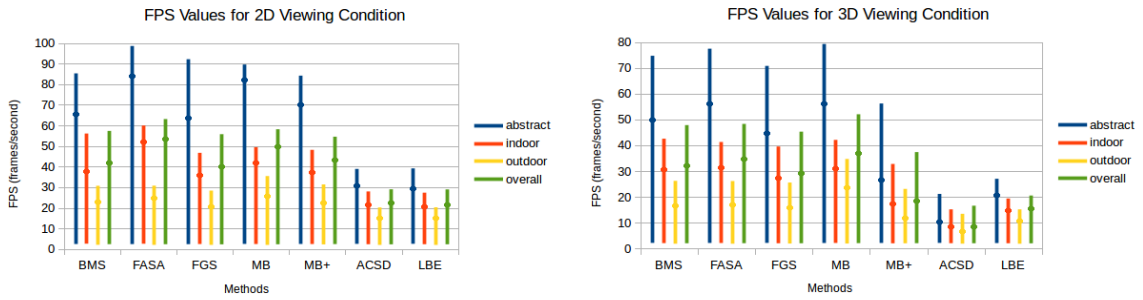

Figure 5.4 Recorded frame rates of the real-time saliency detection methods in the VEs with 2D and 3D viewing conditions. The values are represented in frame per second (FPS). Vertical lines present the minimum and maximum FPS values recorded during the sessions while dots give the mean FPS value in the session.

The frame rate results obtained from the test are given in Figure 5.4. The mean frame rates of the abstract VE in both viewing conditions are higher than the other two VEs. This is because the abstract VE has fewer and simpler models than the other two VEs and this results in the existence of fewer polygons in the abstract VE. Regarding 2D viewing condition, the FASA method grabs attention as the leader with a value of 53.68 overall mean frame rate. MB and MB+ are the follower of FASA with 49.98 and 43.31 overall mean FPS values, respectively. As opposed to 2D viewing condition, the MB method achieves the best overall mean frame rate with a value of 36.96 in 3D viewing conditions. It is followed by FASA and BMS with overall mean FPS values of 34.98 and 32.32, respectively.

5.3. Results of Non-Real-Time Saliency Detection Methods

5.3.1. Metric Scores

The mean metric scores are given in Table 5.2 for desktop and VR viewing conditions. The per-VE scores are calculated for both viewing conditions separately by averaging the scores of the frames of each VE and presented in the first three column groups of the table. The last column group presents the mean scores of all frames without discriminating among VEs.

In both viewing conditions, DMRANet achieves the best scores in the location-based metrics evaluating the similarity (AUC-Borji, AUC-Judd, NSS) and distribution-based

metric evaluating the dissimilarity (KL), while D^3Net achieves the best scores in the distribution-based metrics evaluating the similarity (CC, SIM). Although JL-DCF does not achieve the best score in any metric, its results are quite close to the other best scores, especially in AUC-Judd, CC, and SIM. The learning-based saliency prediction approaches surpass the heuristic-based ones in most metrics. Also, it can be induced that there is a general enhancement of the metric results for both displaying situations in SLC $\rightarrow$ DSLC transitions via the depth transformation model.

	Abstract VE						Indoor VE						Outdoor VE						Overall					
(2D)	**AUC - Borji**	**AUC - Judd**	**KL**	**NSS**	**CC**	**SIM**	**AUC - Borji**	**AUC - Judd**	**KL**	**NSS**	**CC**	**SIM**	**AUC - Borji**	**AUC - Judd**	**KL**	**NSS**	**CC**	**SIM**	**AUC - Borji**	**AUC - Judd**	**KL**	**NSS**	**CC**	**SIM**
CDCP	0.63	0.85	1.95	0.90	0.56	0.48	0.56	0.80	2.75	0.54	0.33	0.38	0.51	0.77	2.62	0.27	0.10	0.23	0.56	0.81	2.35	0.57	0.32	0.35
MBSD	0.59	0.71	13.50	0.74	0.55	0.46	0.53	0.65	15.38	0.34	0.20	0.22	0.50	0.52	22.31	0.07	0.00	0.04	0.54	0.61	17.80	0.38	0.25	0.23
SLC	0.55	0.71	9.91	0.45	0.27	0.29	0.51	0.60	12.69	0.15	0.10	0.21	0.48	0.44	18.35	-0.13	-0.07	0.04	0.51	0.58	14.15	0.14	0.09	0.17
DSLC	0.57(↑)	0.79(↑)	1.27(↑)	0.75(↑)	0.47(↑)	0.41(↑)	0.52(↑)	0.57	1.78(↑)	0.22(↑)	0.15(↑)	0.27(↑)	0.49(↑)	0.40	2.62(↑)	-0.11(↑)	-0.03	0.15(↑)	0.53(↑)	0.58(↑)	1.95(↑)	0.28(↑)	0.20(↑)	0.28(↑)
D^3Net	0.52	**0.94(*)**	3.12	0.88	**0.97(*)**	**0.89(*)**	0.52	**0.94(*)**	3.21	**0.76(*)**	**0.96(*)**	**0.88(*)**	**0.52(*)**	**0.93(*)**	3.70	**0.79(*)**	**0.98(*)**	**0.89(*)**	0.52	**0.94(*)**	3.40	**0.83(*)**	**0.97(*)**	**0.89(*)**
JL-DCF	0.53	**0.97(*)**	1.49	**0.92(*)**	**0.97(*)**	**0.88(*)**	0.52	**0.98(*)**	**1.28(*)**	**0.80(*)**	**0.96(*)**	**0.87(*)**	**0.52(*)**	**0.97(*)**	**1.53(*)**	**0.84(*)**	**0.97(*)**	**0.88(*)**	0.52	**0.97(*)**	**1.48(*)**	**0.87(*)**	**0.97(*)**	**0.88(*)**
DMRANet	**0.82(*)**	**0.99(*)**	**0.26(*)**	**1.53(*)**	**0.56(*)**	**0.52(*)**	**0.79(*)**	**0.99(*)**	**0.22(*)**	**1.28(*)**	**0.56(*)**	**0.53(*)**	**0.82(*)**	**0.99(*)**	**0.23(*)**	**1.48(*)**	**0.52(*)**	**0.50(*)**	**0.82(*)**	**0.99(*)**	**0.24(*)**	**1.48(*)**	**0.54(*)**	**0.52(*)**
(3D)	**AUC - Borji**	**AUC - Judd**	**KL**	**NSS**	**CC**	**SIM**	**AUC - Borji**	**AUC - Judd**	**KL**	**NSS**	**CC**	**SIM**	**AUC - Borji**	**AUC - Judd**	**KL**	**NSS**	**CC**	**SIM**	**AUC - Borji**	**AUC - Judd**	**KL**	**NSS**	**CC**	**SIM**
CDCP	0.60	0.86	2.16	0.71	0.66	0.57	0.55	0.82	2.86	0.43	0.54	0.51	0.52	0.84	2.06	0.42	0.18	0.29	0.56	0.85	2.20	0.54	0.42	0.43
MBSD	0.57	0.70	14.6	0.54	0.64	0.52	0.53	0.66	15.95	0.28	0.35	0.30	0.51	0.56	20.97	0.21	0.03	0.06	0.54	0.63	17.71	0.36	0.33	0.28
SLC	0.53	0.69	11.2	0.33	0.29	0.31	0.50	0.56	14.14	0.06	0.05	0.19	0.50	0.50	16.65	0.00	-0.06	0.06	0.51	0.58	14.07	0.14	0.10	0.18
DSLC	0.54(↑)	0.73(↑)	1.62(↑)	0.52(↑)	0.47(↑)	0.43(↑)	0.51(↑)	0.54	1.64(↑)	0.15(↑)	0.18(↑)	0.31(↑)	0.50	0.43	2.37(↑)	-0.03	-0.02	0.17(↑)	0.52(↑)	0.57	1.97(↑)	0.22(↑)	0.20(↑)	0.30(↑)
D^3Net	0.53	**0.95(*)**	3.05	0.65	**0.99(*)**	**0.95(*)**	0.52	**0.94(*)**	3.26	**0.48(*)**	**0.99(*)**	**0.93(*)**	**0.52(*)**	**0.94(*)**	3.51	**0.65(*)**	**0.99(*)**	**0.95(*)**	0.52	**0.94(*)**	3.28	**0.63(*)**	**0.99(*)**	**0.95(*)**
JL-DCF	0.53	**0.99(*)**	**1.18(*)**	0.68	**0.99(*)**	**0.94(*)**	0.52	**0.98(*)**	**1.25(*)**	**0.51(*)**	**0.99(*)**	**0.93(*)**	**0.53(*)**	**0.99(*)**	**1.30(*)**	**0.68(*)**	**0.99(*)**	**0.94(*)**	0.53	**0.99(*)**	**1.25(*)**	**0.66(*)**	**0.99(*)**	**0.94(*)**
DMRANet	**0.80(*)**	**0.99(*)**	**0.09(*)**	**1.29(*)**	0.49	0.47	**0.74(*)**	**0.99(*)**	**0.15(*)**	**1.00(*)**	0.47	0.47	**0.80(*)**	**0.99(*)**	**0.15(*)**	**1.33(*)**	**0.48(*)**	**0.45(*)**	**0.79(*)**	**0.99(*)**	**0.12(*)**	**1.27(*)**	**0.48(*)**	**0.46(*)**

Colored cells indicate the best scores in each metric.
(*) indicates the scores of learning-based saliency detection methods which outperform all the heuristic-based saliency detection methods in that metric.
(↑) indicates improvement in the metric score from SLC to DSLC via depth-transformation.

Table 5.2 Average scores of the non-real-time saliency detection methods for desktop and VR displaying situations. The scores shown below are truncated to two decimal places for illustrative purposes.

Fig. 5.5 shows the overall score distributions of the methods represented by strip-plots again for both viewing conditions. It is seen that the scores of the heuristic-based methods scatter in wider ranges more sparsely, while the scores of the learning-based methods are realized in tighter ranges more densely.

Finally, Fig. 5.6 presents a sample of visual results for the depth map and RGB image pairs from the dataset along with their corresponding ground truths in 2D and 3D.

5.3.2. Computation Times

Regarding the training times of the learning-based saliency detection methods, DMRA's training took 254 hours, 28 minutes, 38 seconds which represents a very costly process.

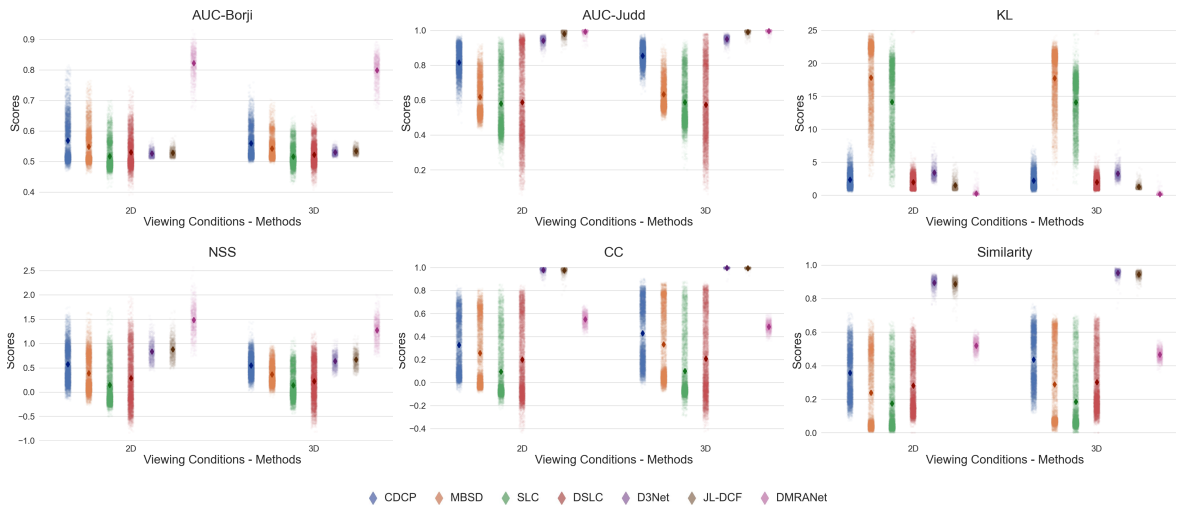

Figure 5.5 Scores obtained by the evaluated methods are presented as strip-plots that give the range and distribution of scores, s.t., strip becomes more transparent where attained scores are sparser. The vertical axis in each plot is specific to the range achieved by the respective metric, while the scores are grouped for the 2D and 3D viewing conditions on the left and the right, respectively. Diamonds mark the corresponding means.

It is followed by JL-DCF with a training duration of 60 hours, 14 minutes, 38 seconds, and the least time-consuming training process occurred for D^{3}Net with 10 hours 15 minutes 1 second. These are the averages of the training times of the models trained for desktop and VR viewing conditions of each model, for which there are no differences between the inputs.

In Fig. 5.7, the average per-frame saliency map generation times of the methods are presented. As there is a remarkable difference in the execution times between the two groups of methods, the results are presented in two charts with different time scales to provide a clearer insight.

Within the heuristic-based saliency detection methods, the most time-consuming one is SLC, and it is followed by MBSD, CDCP, and DSLC, respectively, with substantial reductions. While DSLC executes faster than the other three, it should be noted that this measure does not account for the time to calculate the input RGB attention maps of DSLC beforehand.

On the other side, saliency map generation is considerably faster in the DL-based saliency detection methods despite their costly training processes. D^{3}Net has a domination in terms of speed against other methods by a significant margin, while the other two DL-based

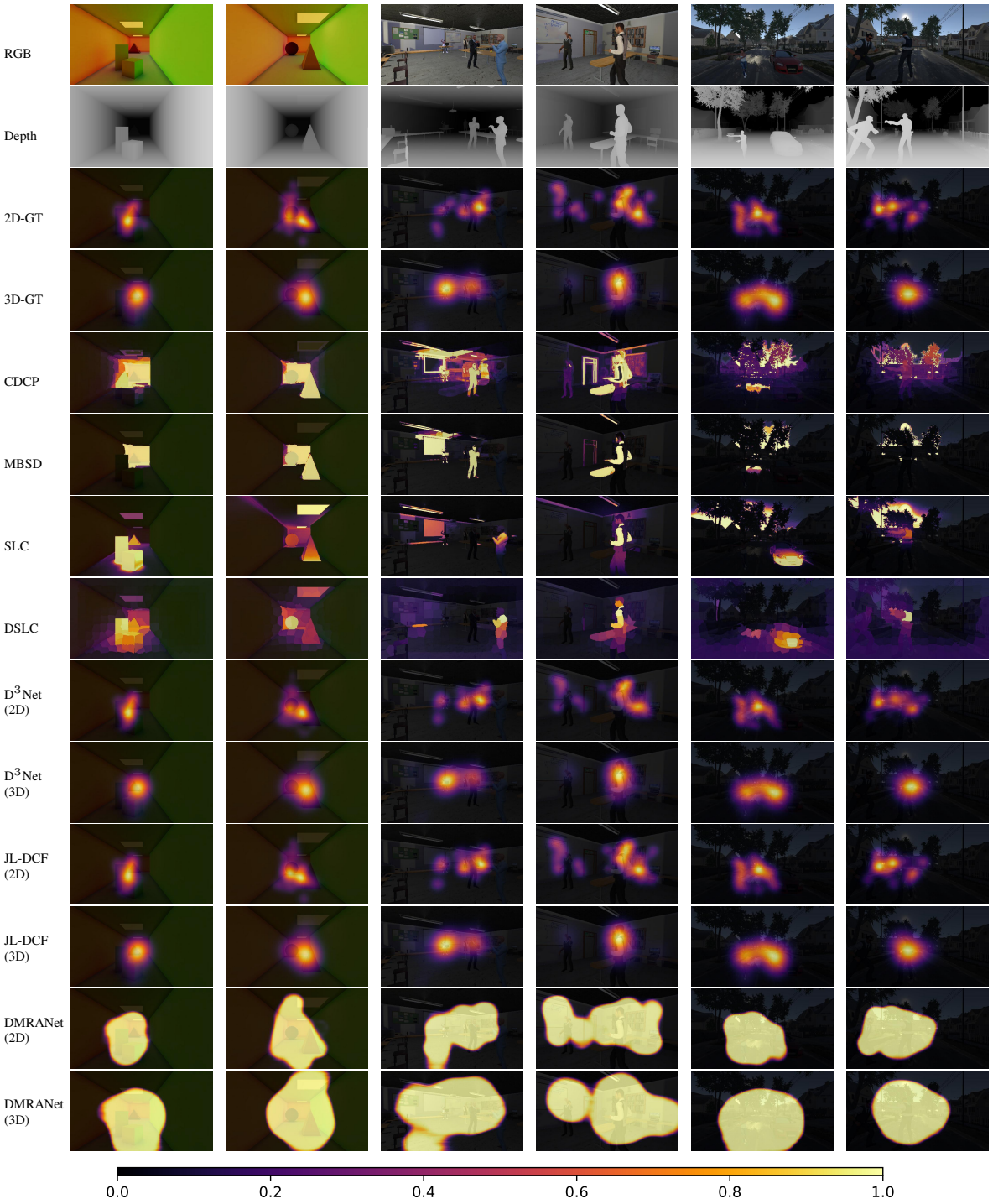

Figure 5.6 Saliency maps (in the latter 10 rows) of sample frames (in the 1st row) from the dataset along with their corresponding depth maps, 2D and 3D ground truths (in the 2nd, 3rd and 4th rows, respectively). The first, the second and the third pair of frames belong to abstract, indoor and outdoor VEs, respectively.

methods exhibit slower performances than D^3Net but are still considerably faster than the heuristic-based saliency detection methods.

For each learning-based method, the training and map generation times of the 2D and 3D models are combined because the 2D and 3D models of each method share the same network architecture, and, as expected, a meaningful difference between their execution times was not observed. The images with the resolution 638x359 are used as inputs and outputs during the training and saliency map generation phases.

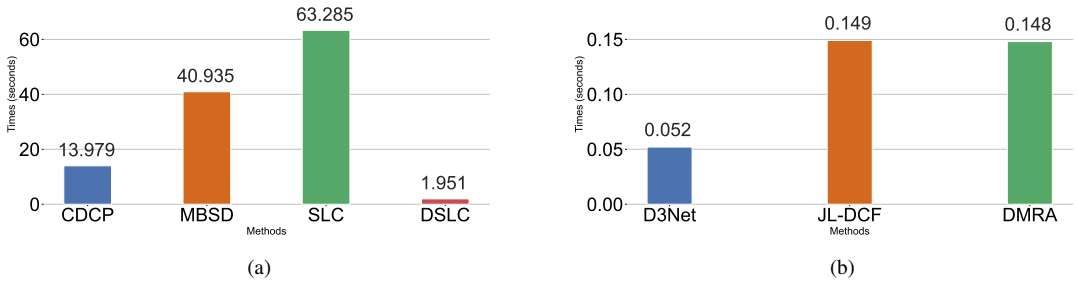

Figure 5.7 Average per frame saliency map generation times in seconds of (a) heuristic-based methods and (b) learning-based methods.

6. DISCUSSION

6.1. Viewing Behaviors

Our comparison of the user behaviors between desktop and VR viewing conditions showed that the behaviors of users are most likely to be different between the viewing conditions when the depth of the frame is higher. The gaze points in these frames are scattered to the distant parts of the frame for 2D viewing while they tend to heap together on a closer region to the camera for 3D viewing. In the exact opposite situation, i.e., for the frames where the depth-of-field is lower, the gaze points tend to heap together at a closer region to the camera for both viewing conditions. The non-salient contents such as plants, houses, etc. are usually located at the points closer to the camera in outdoor VE, which offers higher scene depth in comparison with the other VEs. In this VE, the gaze locations are clustered to the distant points where the attention-grabbing elements are located.

6.2. Metric Scores

Interestingly, the prediction performances of the heuristic-based RGB-D saliency detection approaches decrease drastically by comparison with the results in their original papers where they are proposed. This is likely for the reason that there are important characteristic differences between the dataset used in this work and the the ones used in their respective studies. The latter ones may include images that are more appropriate to extract saliency information exploiting low-level cues, especially the depth, i.e., the salient objects may be clearly distinguishable from their surroundings and the background of the images with closer distances to the camera. However, the dataset used in this study contains more homogeneous content in this aspect. Our findings from the per-VE average metric scores of the non-real-time saliency detection methods support this rationale. That is, the metric scores of the heuristic-based saliency prediction approaches are lower in the outdoor VE, in which the objects are not as easily distinguishable and spread sparsely farther from the camera along with the VE with a relatively high depth-of-field. On the other hand, the other VEs, especially

the abstract VE, have objects or animations that are more easily distinguishable from their surroundings in a narrower depth range close to the camera. Accordingly, the average metric scores increase for the indoor VE and even more so for the abstract VE. But, it is not possible to make such observations for the results of the learning-based saliency prediction approaches. They have higher scores that are close to each other for all VEs. This shows that these methods are able to model even complex contents' saliency successfully under the supervision of existing RGB and depth data. Hence, this indicates that the depth cue might be a critical factor in the saliency detection of VEs by the learning-based algorithms in both viewing conditions even though it may not yield improvement in the prediction accomplishment of the heuristic-based approaches.

6.2.1. Real-Time Methods

Regarding the results of the real-time saliency detection methods employing depth cue evaluated in this study, the average KL and CC scores are lower than the other metric scores for the outdoor VE. On the contrary, the average scores of these metrics are higher than the other metric scores for the indoor VE, which has relatively low depth-of-field and where the salient scene content is located close to the camera. Despite this situation, it cannot be inferred that the depth cue may be the reason by itself for a better saliency prediction performance of the various scene contents because the real-time RGB-D saliency detection approaches do not have a clear dominance in the overall metric scores.

According to the results of our study, the real-time RGB saliency prediction approaches achieved better results than the real-time saliency prediction approaches employing depth information. Such an outcome highlights the significance of 2D cues, even if they are used in 3D environments. If we look at the scores presented in Table 5.1, we can see that the BMS, MB, and MB+ methods outperform the other methods. These methods mainly employ boundary connectivity and surroundedness cues, which fundamentally use the information of the relationship between objects and their backgrounds with different regards. Therefore, we draw a conclusion that both cues are crucial for the real-time saliency detection of both

2D and 3D VEs. Although the prediction performances of these three methods are higher for both viewing conditions, their general superiority is more pronounced in 3D viewing conditions, where the VEs are experienced with an HMD. Furthermore, the metric scores of the methods, which achieve the best prediction performances, are better for VR viewing conditions than the desktop viewing conditions except for NSS.

6.2.2. Non-Real-Time Methods

Regarding the non-real-time saliency detection methods evaluated in this study, for three different VEs, under two different viewing conditions and six different metrics evaluating the saliency from different perspectives, none of the heuristic-based saliency prediction approaches can succeed the best results although they outperform learning-based saliency detection methods in a few cases. For AUC-Borji, CDCP and MBSD perform better than D^3Net and JL-DCF in both viewing conditions for abstract and indoor VEs, while DSLC has higher results than these two methods in both viewing conditions only for abstract VE and SLC performs better than these two methods in 2D viewing condition only for abstract VE. In the KL metric, where lower is better, CDCP and DSLC outperform D^3Net in both viewing conditions for all VEs, while DSLC has a better result than JL-DCF in 2D viewing condition only for abstract VE. For NSS, CDCP has higher results than D^3Net in both viewing conditions for abstract VE, while it outperforms JL-DCF in 3D viewing condition only for abstract VE. Regarding CC and SIM, CDCP outperforms DMRANet in 3D viewing for abstract and indoor VEs, while MBSD has higher results than DMRANet in 3D viewing only for abstract VE. Nonetheless, the learning-based methods surpass the heuristic-based methods in most cases by considerable margins. Thus, in visual saliency prediction of VEs under both desktop and VR viewing conditions, the results demonstrate the superiority of the data-driven techniques exploiting the depth cue amplified with the modern deep learning architectures against the traditional approaches using the depth. While there is no other work that makes such an evaluation in VR, a recent study examined different saliency detection methods in augmented reality (AR) with HMDs and also demonstrated the success of learning-based methods against the heuristic-based methods [72].

In the part of our study in which the viewing behaviors of the users in 2D and 3D viewing conditions are investigated, it was revealed in the collected ground truth data that the center bias is more pronounced in 3D viewing condition and this made in impact that caused meaningful differences between the average metric scores of 2D and 3D viewing. However, the results gathered with the offline methods in this work do not indicate such an impact between the average scores of 2D and 3D viewings. Although the differences in scores are not significant between 2D and 3D in general, certain cases stand out. Considering the CC and SIM results, the methods generally achieved better results for 3D viewing condition. However, it can be seen that DMRANet's results are higher for 2D viewing for these metrics. Therefore, given the continuous similarities in distributions of predicted maps and ground truths, it can be inferred that DMRANet may model the saliency of 2D VEs better than 3D VEs. KL scores, measures of continuous distributional dissimilarities, tend to increase towards 3D viewing condition except for DSLC. Unlike the general situation, NSS scores are higher in 2D viewing condition. Hence, considering the average saliency predicted by the methods at fixation locations, it can be seen that the methods may offer better models for 2D VEs.

When compared to the scores presented in the first part of our study, obtained by the real-time traditional RGB-D image saliency prediction approaches (ACSD [12], LBE [13]) that use the depth cue in addition to 2D cues with the same set of metrics on the same dataset, it is seen that the non-real-time traditional RGB-D image saliency prediction approaches evaluated in this study (CDCP, MBSD) achieved higher accuracy in general while performing much slower. This implies that employing the depth cue in addition to the 2D cues can prove to be quite beneficial for visual saliency prediction using heuristic-based methods in immersive VEs when frame generation rate is not a concern.

Regarding the video saliency, it is seen that SLC, a heuristic-based RGB video saliency method, does not achieve better results than the traditional RGB-D image saliency approaches in this study. On the other hand, it is also seen that the prediction performance is increased via the depth transformation from SLC to DSLC. This points out that the depth information in addition to spatiotemporal information may have positive effects on

the saliency prediction of immersive VEs by the heuristic-based video saliency detection methods.

6.3. Run-Time Analysis

6.3.1. Real-Time Methods

In addition to the saliency prediction performances of the real-time saliency prediction approaches, our frame rate analysis shows that the real-time saliency prediction approaches employing the depth cue may not be appropriate choices during the real-time experiences of the complex VEs (particularly in 3D viewing conditions), whose graphic qualities are high. The main reason for this inference is that it is impossible to pre-prepare the depth maps of the frames before experiencing environments interactively in real-time, i.e., the depth maps of the frames must be extracted instantly and this cause extra overheads in addition to actual costs of the saliency methods used. Because of the limits of HVS, the minimum mean FPS must be higher than 30 for all types of viewing. Furthermore, it is suggested that the mean FPS should be higher than 60 in short 3D viewing experiences while it should be higher than 90 FPS in longer 3D viewing experiences. The average FPS values of the ACSD and LBE methods seem impossible to reach 60 FPS even if they are boosted with different rendering optimizations. The only exception for this situation might be using the latest advanced graphic hardware or reduction of the saliency prediction time significantly.

6.3.2. Non-Real-Time Methods

The training process of DMRANet is notably more costly than the other DL-based methods. At first glance, the reason behind this might seem to be that DMRANet has two extra modules bringing about a heavy computational cost with their complicated structures. Yet, if that were the case, it would also affect the saliency map generation times. Contrarily, the saliency map generation times of DMRANet and JL-DCF are almost the same, which means that the architectural structures of these two methods prompt similar generation costs while D^3Net

has a simpler yet still effective architecture. The actual reason for the differences in their training times is revealed in the amount of augmented training data used since DMRANet augments the training data to a much larger extent than others. This also indicates the importance of data augmentation in CNN-based saliency detection models in VEs with relatively small datasets due to the domination of DMRANet in the metric scores.

Regardless of the fact that the execution times of the evaluated non-real-time heuristic-based saliency detection methods are far below the aforementioned real-time VR requirements, the saliency map generation times of the learning-based saliency prediction approaches are promising. Particularly, D^3Net reaches approximately 20 frames per second, and this can be improved by adopting smaller input/output resolutions, reducing saliency prediction frequency during VR experience or using more advanced hardware.

6.4. Limitations and Future Work

There would be inevitable differences between the viewing behaviors of the users that experience a VE along a predetermined path and those that navigate freely in that VE. However, it was not possible to investigate such differences within the scope of this study as the dataset we used was collected with the VE experiences of users along fixed view paths only. Hence, this is an important research question that should be addressed in future studies.

The cognitive awarenesses of the users with low and high VR experiences are most likely to be different while they navigate through such environments. This might cause significant differences between the metric scores and viewing behaviors of those users. Because finding users with a high level of VR experience is not easy, the average VR experience of the users who participated in this study is quite low. Therefore, our results might not be generalizable to users with high levels of VR experience. The effects of the higher VR experience levels should be examined in future studies comprehensively. Furthermore, dividing users into groups according to their VR experience levels and analyzing the results of each group separately would reveal more robust insights about the visual attention mechanisms of humans in virtually created environments. Similarly, the demographic differences among

the users might have impacts on both user behaviors and metric scores. Thus, such impacts should be investigated in further studies.

The saliency prediction methods evaluated in this study are implemented to extract saliency information from images and videos taken from the real world. However, the dataset used in this study consists of frames from virtually created environments and these methods are most likely to be insufficient to model the saliency of such environments with full potential. Therefore, the saliency prediction methods that aim to solve the visual attention problem of VE should be proposed in the future.

7. CONCLUSION

In this study, we give a detailed analysis of seven real-time image saliency detection approaches in VEs with 2D and 3D viewing conditions. The real-time saliency detection methods investigated in the study exploit various saliency cues to handle the problem from different perspectives. These methods are examined as two groups. Five of them adopt only 2D saliency cues and two of them also use the depth cue. Throughout the data collection with the user study, the users experienced three unique VEs, which offer different types of content and scene depth ranges. The reason for the different depth-of-fields offered by the VEs is that the depth has a special significance because it is an essential piece of realistic stereoscopic rendering in VEs. Therefore, the viewing behaviors of the users in the VEs that has different depth-of-fields should be examined in detail. For fair comparisons of the user behaviors in desktop and VR viewing conditions, the data was collected from each user during a VR session with an HMD as well as a desktop session.

It is shown in the previous studies that a center bias occurs while experiencing VEs in both desktop and VR [37]. Our results confirm this finding. However, our results also show that the center bias tends to be more noticeable in 3D viewing conditions than in 2D viewing conditions. Two methods, which adopt the boundary connectivity feature, mostly achieve the best metric scores for VR viewing conditions. On the other hand, the BMS method, which employs the surroundedness feature, mostly achieves the best metric scores for 2D viewing conditions. Both cues employed by the leading methods handle the relation between the objects and their backgrounds. Although ACSD achieves the most superior scores in the AUC-Borji metric for both viewing conditions, the methods employing the depth information have worse prediction performances than the other methods in general. In conclusion, the results show that the depth cue is not as important as expected in real-time saliency detection of VEs, even though the depth is unignorably crucial in VEs, especially experienced in 3D with HMDs.

We gave the results of a comprehensive analysis of visual saliency in VR displayed by

modern HMDs by comparing various heuristic-based and learning-based saliency prediction approaches in different virtual environments. Our main purpose was to reveal the role of real-timeliness and depth information in the saliency detection of virtually created scenes. Using a dataset comprising fixations in 2D and 3D viewings of a set of three different VEs, we evaluated three RGB-D image saliency prediction approaches, which employ deep learning processes, sharing the same backbone but having different architectures and two traditional RGB-D image saliency prediction approaches comparatively. We also investigated the performance of video saliency in immersive VEs with and without depth information on the same dataset.

The results showed that the learning-based approaches can achieve much higher scores than the heuristic-based approaches if the depth cue is employed. Also, the learning-based approaches can model the saliency of VEs successfully regardless of the VE contents. On the other hand, heuristic-based video saliency detection without employing depth is seen to be a poor choice. Despite the training costs of the learning-based saliency prediction approaches, the saliency map generation times are faster compared to the heuristic-based approaches.

All in all, RGB-D image saliency prediction approaches, which employ learning processes, stand out to be very promising in visual saliency prediction of the VEs in both desktop and immersive VR with their high scores in all metrics and higher computational speed rates. Although the current generation speeds of the learning-based approaches are far below the real-time VR requirements, which must be maintained to provide visually comfortable VR experiences, there is ample room for further improvement in average computation times.

REFERENCES

[1] Inam Ullah, Muwei Jian, Sumaira Hussain, Jie Guo, Hui Yu, Xing Wang, and Yilong Yin. A brief survey of visual saliency detection. *Multimedia Tools and Applications*, 79:34605–34645, **2020**.

[2] Erkut Erdem and Aykut Erdem. Visual saliency estimation by nonlinearly integrating features using region covariances. *Journal of Vision*, 13(4):11–11, **2013**. ISSN 1534-7362.

[3] Jonathan Harel, Christof Koch, and Pietro Perona. Graph-based visual saliency. In B. Schölkopf, J. Platt, and T. Hoffman, editors, *Advances in Neural Information Processing Systems*, volume 19. MIT Press, **2007**.

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

[5] Alexander Kroner, Mario Senden, Kurt Driessens, and Rainer Goebel. Contextual encoder–decoder network for visual saliency prediction. *Neural Networks*, 129:261–270, **2020**. ISSN 0893-6080.

[6] Matthias Kummerer, Thomas S. A. Wallis, Leon A. Gatys, and Matthias Bethge. Understanding low- and high-level contributions to fixation prediction. In *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*. **2017**.

[7] Junting Pan, Cristian Canton Ferrer, Kevin McGuinness, Noel E. O'Connor, Jordi Torres, Elisa Sayrol, and Xavier Giro-i Nieto. SalGAN: Visual Saliency Prediction with Generative Adversarial Networks. *arXiv e-prints*, page arXiv:1701.01081, **2017**.

[8] Roberto Valenti, Nicu Sebe, and Theo Gevers. Image saliency by isocentric curvedness and color. In *2009 IEEE 12th International Conference on Computer Vision*, pages 2185–2192. **2009**.

[9] Wenguan Wang, Jianbing Shen, Fang Guo, Ming-Ming Cheng, and Ali Borji. Revisiting video saliency: A large-scale benchmark and a new model. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. **2018**.

[10] Huazhu Fu, Xiaochun Cao, and Zhuowen Tu. Cluster-based co-saliency detection. *IEEE Transactions on Image Processing*, 22(10):3766–3778, **2013**.

[11] Wei Liu, Shengcai Liao, Weiqiang Ren, Weidong Hu, and Yinan Yu. High-level semantic feature detection: A new perspective for pedestrian detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 5187–5196. **2019**.

[12] Ran Ju, Ling Ge, Wenjing Geng, Tongwei Ren, and Gangshan Wu. Depth saliency based on anisotropic center-surround difference. *2014 IEEE International Conference on Image Processing (ICIP)*, pages 1115–1119, **2014**.

[13] David Feng, Nick Barnes, Shaodi You, and Chris McCarthy. Local background enclosure for rgb-d salient object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2343–2350. **2016**.

[14] Junwei Han, Hao Chen, Nian Liu, Chenggang Yan, and Xuelong Li. Cnns-based rgb-d saliency detection via cross-view transfer and multiview fusion. *IEEE Transactions on Cybernetics*, 48(11):3171–3183, **2018**.

[15] Jia-Xing Zhao, Yang Cao, Deng-Ping Fan, Ming-Ming Cheng, Xuan-Yi Li, and Le Zhang. Contrast prior and fluid pyramid integration for rgbd salient object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*. **2019**.

[16] Abel J Lungu, Wout Swinkels, Luc Claesen, Puxun Tu, Jan Egger, and Xiaojun Chen. A review on the applications of virtual reality, augmented reality and mixed reality in surgical simulation: an extension to different kinds of surgery. *Expert Review of Medical Devices*, 18(1):47–62, **2021**. doi:10.1080/17434440.2021.1860750.

[17] Laura Freina and Michela Ott. A literature review on immersive virtual reality in education: state of the art and perspectives. In *The international scientific conference elearning and software for education*, volume 1, pages 10–1007. **2015**.

[18] Joseph J LaViola Jr. A discussion of cybersickness in virtual environments. *ACM Sigchi Bulletin*, 32(1):47–56, **2000**.

[19] Berk Cebeci, Ufuk Celikcan, and Tolga K. Capin. A comprehensive study of the affective and physiological responses induced by dynamic virtual reality environments. *Computer Animation and Virtual Worlds*, 30(3-4):e1893, **2019**. doi:https://doi.org/10.1002/cav.1893. E1893 cav.1893.

[20] Dilara Albayrak, Mehmet Bahadir Askin, Tolga K. Capin, and Ufuk Celikcan. Visual saliency prediction in dynamic virtual reality environments experienced with head-mounted displays: An exploratory study. In *2019 International Conference on Cyberworlds (CW)*, pages 61–68. **2019**.

[21] Ufuk Celikcan, Mehmet Bahadir Askin, Dilara Albayrak, and Tolga K. Capin. Deep into visual saliency for immersive vr environments rendered in real-time. *Computers & Graphics*, 88:70–82, **2020**. ISSN 0097-8493.

[22] Mehmet Bahadir Askin and Ufuk Celikcan. Learning based versus heuristic based: A comparative analysis of visual saliency prediction in immersive virtual reality. *Computer Animation and Virtual Worlds*, 33(6):e2106, **2022**. doi:https://doi.org/10.1002/cav.2106.

[23] Gerd Marmitt and Andrew Duchowski. Modeling visual attention in vr: Measuring the accuracy of predicted scanpaths. In *Eurographics 2002 - Short Presentations*. Eurographics Association, **2002**.

[24] Zhiming Hu. Gaze analysis and prediction in virtual reality. In *2020 IEEE Conference on Virtual Reality and 3D User Interfaces Abstracts and Workshops (VRW)*, pages 543–544. **2020**. doi:10.1109/VRW50115.2020.00123.

[25] Radiah Rivu, Ville Mäkelä, Mariam Hassib, Yomna Abdelrahman, and Florian Alt. Exploring how saliency affects attention in virtual reality. In *Human-Computer Interaction – INTERACT 2021*, pages 147–155. Springer International Publishing, Cham, **2021**.

[26] Evangelos Alexiou, Peisen Xu, and Touradj Ebrahimi. Towards modelling of visual saliency in point clouds for immersive applications. In *2019 IEEE International Conference on Image Processing (ICIP)*, pages 4325–4329. **2019**.

[27] Oyewole Oyekoya, William Steptoe, and Anthony Steed. A saliency-based method of simulating visual attention in virtual scenes. In *Proceedings of the 16th ACM Symposium on Virtual Reality Software and Technology*, pages 199–206. ACM, **2009**.

[28] Rachel McDonnell, Michéal Larkin, Benjamín Hernández, Isaac Rudomin, and Carol O'Sullivan. Eye-catching crowds: Saliency based selective variation. *ACM Trans. Graph.*, 28(3), **2009**. ISSN 0730-0301.

[29] A. Deniz Aladagli, Erhan Ekmekcioglu, Dmitri Jarnikov, and Ahmet Kondoz. Predicting head trajectories in 360° virtual reality videos. In *2017 International Conference on 3D Immersion (IC3D)*, pages 1–6. **2017**.

[30] Ana De Abreu, Cagri Ozcinar, and Aljosa Smolic. Look around you: Saliency maps for omnidirectional images in vr applications. In *2017 Ninth International Conference on Quality of Multimedia Experience (QoMEX)*, pages 1–6. **2017**. doi:10.1109/QoMEX.2017.7965634.

[31] Ruofei Du and Amitabh Varshney. Saliency computation for virtual cinematography in 360° videos. *IEEE Computer Graphics and Applications*, 41(4):99–106, **2021**.

[32] Fang-Yi Chao, Lu Zhang, Wassim Hamidouche, and Olivier Deforges. Salgan360: Visual saliency prediction on 360 degree images with generative adversarial networks. In *2018 IEEE International Conference on Multimedia Expo Workshops (ICMEW)*, pages 01–04. **2018**.

[33] Ching-Ling Fan, Jean Lee, Wen-Chih Lo, Chun-Ying Huang, Kuan-Ta Chen, and Cheng-Hsin Hsu. Fixation prediction for 360° video streaming in head-mounted virtual reality. In *Proceedings of the 27th Workshop on Network and Operating Systems Support for Digital Audio and Video*, page 67–72. Association for Computing Machinery, New York, NY, USA, **2017**.

[34] Rafael Monroy, Sebastian Lutz, Tejo Chalasani, and Aljosa Smolic. Salnet360: Saliency maps for omni-directional images with cnn. *Signal Processing: Image Communication*, 69:26–34, **2018**.

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

[36] Yashas Rai, Patrick Le Callet, and Philippe Guillotel. Which saliency weighting for omni directional image quality assessment? In *2017 Ninth International Conference on Quality of Multimedia Experience (QoMEX)*, pages 1–6. **2017**.

[37] Vincent Sitzmann, Ana Serrano, Amy Pavel, Maneesh Agrawala, Diego Gutierrez, Belen Masia, and Gordon Wetzstein. Saliency in vr: How do people explore virtual environments? *IEEE Transactions on Visualization and Computer Graphics*, 24(4):1633–1642, **2018**.

[38] George Alex Koulieris, George Drettakis, Douglas Cunningham, and Katerina Mania. Gaze prediction using machine learning for dynamic stereo manipulation

in games. In *2016 IEEE Virtual Reality (VR)*, pages 113–120. **2016**. doi:10.1109/VR.2016.7504694.

[39] Robert Konrad, Anastasios Angelopoulos, and Gordon Wetzstein. Gaze-contingent ocular parallax rendering for virtual reality. *ACM Trans. Graph.*, 39(2), **2020**. ISSN 0730-0301. doi:10.1145/3361330.

[40] Emre Avan, Tolga K. Capin, Hasmet Gurcay, and Ufuk Celikcan. Enhancing vr experience with rbf interpolation based dynamic tuning of stereoscopic rendering. *Computers & Graphics*, 102:390–401, **2022**. ISSN 0097-8493. doi:https://doi.org/10.1016/j.cag.2021.09.016.

[41] Ufuk Celikcan, Gokcen Cimen, E Bengu Kevinc, and Tolga Capin. Attention-aware disparity control in interactive environments. *The Visual Computer*, 29(6-8):685–694, **2013**.

[42] Yeo-Jin Yoon, Jaechun No, and Soo-Mi Choi. Saliency-guided stereo camera control for comfortable vr explorations. *IEICE Transactions on Information and Systems*, E100.D:2245–2248, **2017**.

[43] Zhenping Xia, Cheng Cheng, and Xiaohua Li. Visual comfort enhancement study based on visual attention detection for stereoscopic displays. *Journal of the Society for Information Display*, 24:633–640, **2016**.

[44] Feng Shao, Zhutuan Li, Qiuping Jiang, Gangyi Jiang, Mei Yu, and Zongju Peng. Visual discomfort relaxation for stereoscopic 3d images by adjusting zero-disparity plane for projection. *Displays*, 39:125–132, **2015**.

[45] Che-Han Chang, Chia-Kai Liang, and Yung-Yu Chuang. Content-aware display adaptation and interactive editing for stereoscopic images. *IEEE Transactions on Multimedia*, 13(4):589–601, **2011**.

[46] Sang-Hyun Cho and Hang-Bong Kang. A new visual comfort-based stereoscopic image retargeting method. *International Symposium on Visual Computing (ISVC)*, pages 290–300, **2013**.

[47] Hong-Shang Lin, Shuen-Huei Guan, Chu-Tien Lee, and Ming Ouhyoung. Stereoscopic 3d experience optimization using cropping and warping. In *SIGGRAPH Asia 2011 Sketches on - SA '11*, page 1. ACM, ACM Press, New York, **2011**.

[48] Guangyu Nie, Yue Liu, and Yongtian Wang. [poster] prevention of visually induced motion sickness based on dynamic real-time content-aware non-salient area blurring. In *2017 IEEE International Symposium on Mixed and Augmented Reality (ISMAR-Adjunct)*, pages 75–78. **2017**.

[49] Peiteng Shi, Markus Billeter, and Elmar Eisemann. Salientgaze: Saliency-based gaze correction in virtual reality. *Computers & Graphics*, 91:83–94, **2020**. ISSN 0097-8493.

[50] Mankyu Sung and SungHwan Choi. Selective anti-aliasing for virtual reality based on saliency map. In *2017 International Symposium on Ubiquitous Virtual Reality (ISUVR)*, pages 16–19. **2017**.

[51] Ajay Kumar Reddy Poreddy, Peter A. Kara, Balasubramanyam Appina, and Aniko Simon. A no-reference 3D virtual reality image quality assessment algorithm based on saliency statistics. In *Optics and Photonics for Information Processing XV*, volume 11841, pages 160 – 169. SPIE, **2021**.

[52] Jianming Zhang and Stan Sclaroff. Exploiting surroundedness for saliency detection: a boolean map approach. *IEEE transactions on pattern analysis and machine intelligence*, 38(5):889–902, **2015**.

[53] Gökhan Yildirim and Sabine Süsstrunk. Fasa: fast, accurate, and size-aware salient object detection. In *Asian Conference on Computer Vision*, pages 514–528. Springer, **2014**.

[54] Sebastian Montabone and Alvaro Soto. Human detection using a mobile platform and novel features derived from a visual saliency mechanism. *Image Vision Comput.*, 28(3):391–402, **2010**. ISSN 0262-8856.

[55] Jianming Zhang, Stan Sclaroff, Zhe L. Lin, Xiaohui Shen, Brian L. Price, and Radomír Mech. Minimum barrier salient object detection at 80 fps. *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 1404–1412, **2015**.

[56] Chunbiao Zhu, Wenhao Zhang, Thomas H. Li, Shan Liu, and Ge Li. Exploiting the value of the center-dark channel prior for salient object detection. *ACM Trans. Intell. Syst. Technol.*, 10(3):32:1–32:20, **2019**. ISSN 2157-6904.

[57] Chunbiao Zhu and Ge Li. A multilayer backpropagation saliency detection algorithm and its applications. *Multimedia Tools and Applications*, 77:25181–25197, **2018**. ISSN 1573-7721.

[58] Chenglizhao Chen, Shuai Li, Yongguang Wang, Hong Qin, and Aimin Hao. Video saliency detection via spatial-temporal fusion and low-rank coherency diffusion. *IEEE Transactions on Image Processing*, 26(7):3156–3170, **2017**.

[59] Runmin Cong, Jianjun Lei, Huazhu Fu, Junhui Hou, Qingming Huang, and Sam Kwong. Going from rgb to rgbd saliency: A depth-guided transformation model. *IEEE Transactions on Cybernetics*, 50(8):3627–3639, **2020**.

[60] Deng-Ping Fan, Zheng Lin, Zhao Zhang, Menglong Zhu, and Ming-Ming Cheng. Rethinking rgb-d salient object detection: Models, data sets, and large-scale benchmarks. *IEEE Transactions on Neural Networks and Learning Systems*, 32(5):2075–2089, **2021**.

[61] Keren Fu, Deng-Ping Fan, Ge-Peng Ji, Qijun Zhao, Jianbing Shen, and Ce Zhu. Siamese network for rgb-d salient object detection and beyond. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, **2021**.

[62] Yongri Piao, Wei Ji, Jingjing Li, Miao Zhang, and Huchuan Lu. Depth-induced multi-scale recurrent attention network for saliency detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*. **2019**.

[63] Liqiang Huang and Harold Pashler. A boolean map theory of visual attention. *Psychological review*, 114 3:599–631, **2007**.

[64] Simone Frintrop. *VOCUS: A visual attention system for object detection and goal-directed search*, volume 3899. Springer, **2006**.

[65] Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In *Proceedings of the 2001 IEEE computer society conference on computer vision and pattern recognition. CVPR 2001*, volume 1, pages I–I. IEEE, **2001**.

[66] Ran Margolin, Ayellet Tal, and Lihi Zelnik-Manor. What makes a patch distinct? In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1139–1146. **2013**.

[67] Tongwei Ren, Ran Ju, Yan Liu, and Gangshan Wu. How important is location in saliency detection. In *Proceedings of International Conference on Internet Multimedia Computing and Service*, pages 10–13. **2014**.

[68] Paria Mehrani and Olga Veksler. Saliency segmentation based on learning and graph cut refinement. In *BMVC*, pages 1–12. **2010**.

[69] Yao Qin, Huchuan Lu, Yiqun Xu, and He Wang. Saliency detection via cellular automata. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. **2015**.

[70] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. In *International Conference on Learning Representations*. **2015**.

[71] Tsung-Yi Lin, Piotr Dollar, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. **2017**.

[72] Huiyu Duan, Wei Shen, Xiongkuo Min, Danyang Tu, Jing Li, and Guangtao Zhai. Saliency in augmented reality, **2022**.