Modelling Light Field Visual Attention: A Saliency Field Approach

by

Ailbhe Gill, B.Sc

Dissertation

Presented to the

University of Dublin, Trinity College

in fulfillment

of the requirements

for the Degree of

Doctor of Philosophy in Computer Science

University of Dublin, Trinity College

July 2022
Declaration

I, the undersigned, declare that this work has not previously been submitted as an exercise for a degree at this, or any other University, and that unless otherwise stated, is my own work.

Ailbhe Gill

July 1, 2022
Permission to Lend and/or Copy

I, the undersigned, agree that Trinity College Library may lend or copy this thesis upon request.

______________________________
Ailbhe Gill

July 1, 2022
Acknowledgments

Sincere thanks to my supervisor Prof. Aljosa Smolic for your invaluable guidance over the last four years and the enthusiasm you bring to the workplace. Thanks as well to the post-docs: Dr. Emin Zerman for your expert insights and encouragement, and to Dr. Mikael Le Pendu and Dr. Martin Alain for your knowledgeable input and for bringing me up to speed with your software. A special mention to Dr. Weston Aenchbacher for organising the light field group meetings, and Dr. Cagri Ozcinar for your guidance in the early stages of my research development. I’d like to thank Cian Judd for being so generous with your time and showing me how to operate the Eyelink 1000 Plus. Thanks to Gail Weadick, for always being so welcoming and ensuring the lab ran smoothly. Credit also goes to my fellow PhD students, Yang Chen, Pierre Matysiak, Koustav Ghosal, Rafael Monroy, Sebastian Lutz, Colm O’Fearghail, Matthew Moynihan, Tejo Chalasani and David Hardman, a central support system in this process, who made this experience all the more enjoyable and valuable, and from whom I have learnt so much research-wise and beyond. Thanks to my parents, you have given me every opportunity to pursue my passions and taught me to make the most of opportunities that present themselves. No doubt this has played a major role in how I’ve come this far. Finally, thanks to David for always encouraging me to try new things and helping me get through the more challenging periods of my PhD. Your support over the last few years has been immense and I am extremely grateful.

AILBHE GILL

University of Dublin, Trinity College
July 2022
Abstract

Light field imaging is becoming more accessible, hence understanding how people perceive and interact with it will be of immense value. Although visual attention has been explored for traditional 2D images, we extend this line of research to light fields. Since light field cameras capture information from all spatio-angular dimensions, the data captured can be rendered by adjusting the plane of focus, changing the aperture or shifting the viewing angle. To analyse the effect of the focus parameter on visual attention, we generate a database of focally varying light field renderings which can be used to gather saliency data using a state-of-the-art Fourier Disparity Layer renderer. Then, we create a corresponding eye fixation dataset gathered from 21 participants which is the first of its kind for light field visual attention ground truth. Our analysis of saliency maps generated from the eye fixation data reveal that light field saliency is of a higher dimension and encompasses that of 2D saliency. We demonstrate that light field renderings encode additional information compared to regular images, which we then exploit to build a 4D saliency field on which we can perform operations similar to the light field itself to obtain saliency maps of any light field rendering.

To generate the saliency field, we integrate light field refocusing algorithms with a state-of-the-art deep learning model to create a hybrid data-driven approach to saliency prediction of light field renderings. In our first model, we use a shift-and-sum refocusing algorithm and in the second, we employ the Fourier disparity layer method. We evaluate the performance of our saliency prediction models against a baseline and show their efficacy both qualitatively and quantitatively for a variety of metrics. In this work, we concentrate mainly on validating our prediction for changes in the focus cue - an operation intrinsic to light fields and a challenge for visual attention prediction. However, our models can also generate saliency maps of renderings where the angle and aperture have been adjusted.
Finally, to show the capabilities of our saliency field model as an automated ranking mechanism, we demonstrate its plausible application in optimal focal plane selection for gaze contingent blur software. In addition, we discuss other possible uses from AR/VR headset design to compression and quality assessment. We foresee this work stimulating further investigation into how the intrinsic properties of light fields influence visual attention.
Contents

Acknowledgments iv
Abstract v
List of Tables x
List of Figures xi
Acronyms xiv

Chapter 1 Introduction 1
  1.1 Motivation for Light Field Visual Attention Research 2
  1.2 Research Questions 3
  1.3 Thesis Scope 4
  1.4 Thesis Outline 5
  1.5 Main Contribution and Publications 6

Chapter 2 Background 8
  2.1 Light Fields 8
    2.1.1 Light Field Acquisition 9
    2.1.2 Light Field Image Renderings 13
    2.1.3 Light Field and Multi-View Displays 17
  2.2 Visual Saliency 22
    2.2.1 Saliency Estimation Models 22
    2.2.2 Visual Attention Prediction in Images 24
    2.2.3 State-of-the-Art in Visual Attention Prediction 26
  2.3 Saliency Prediction of Light Field Data 31
    2.3.1 Traditional Methods for Light Field Salient Object Detection 32
    2.3.2 Deep Learning Methods for Light Field Saliency Prediction 41
    2.3.3 Datasets Available 46
2.4 Evaluation Metrics ........................................ 47

Chapter 3 Light Field Visual Attention Dataset ........................................ 49
3.1 Introduction ........................................ 49
3.2 Related Work ........................................ 50
  3.2.1 Eye Movements .................................... 50
  3.2.2 Eyelink 1000 Plus .................................. 51
3.3 Light Field Eye Gaze Data Collection ........................................ 52
  3.3.1 Choice of Light Field Data and Rendering Methods .................. 52
  3.3.2 Experimental Setup .................................. 53
  3.3.3 Participants & Methodology ................................ 54
3.4 Light Field Eye Gaze Data Processing ........................................ 55
  3.4.1 Saliency Map Generation ................................ 55
  3.4.2 Fixation Map Generation ................................ 56
  3.4.3 Scanpath Generation .................................. 56
3.5 Light Field Eye Gaze Data Analysis ........................................ 56
  3.5.1 Qualitative Analysis of Scanpaths and Saliency Maps ............... 56
  3.5.2 Quantitative Analysis of the Distribution of Fixations ............. 59
  3.5.3 Analysis of the Influence of the Focus Cue ........................ 61
  3.5.4 Limitations of the Datasets Used and Collected ..................... 67
3.6 Conclusion ........................................ 68

Chapter 4 Focus Guided Saliency Estimation ........................................ 70
4.1 Introduction ........................................ 70
4.2 Background & Related Work ........................................ 72
  4.2.1 Shift-and-Sum Light Field Refocusing ................................ 72
4.3 Proposed Method ........................................ 73
  4.3.1 Focus Guided Saliency .................................. 73
  4.3.2 Integration into Rendering .................................. 75
4.4 Experimental Results ........................................ 75
  4.4.1 Dataset ........................................ 75
  4.4.2 Saliency Estimation Method ................................ 76
  4.4.3 Quantitative Analysis .................................. 76
  4.4.4 Qualitative Analysis .................................. 77
  4.4.5 Computation Time Analysis ................................ 79
  4.4.6 Downsampling Results .................................. 81
4.5 Conclusion ........................................ 83
# Chapter 5  Saliency Prediction Using Fourier Disparity Layers

5.1 Introduction ......................................................... 84
5.2 Background & Related Work ..................................... 85
   5.2.1 Fourier Disparity Layer Method ............................. 85
5.3 Method Overview .................................................. 85
5.4 Construction of Saliency Layers ................................. 87
   5.4.1 FDL notations and construction ............................. 87
   5.4.2 Extraction of a Disparity Confidence Volume for Disparity Transfer ... 88
5.5 Refocused Rendering from Saliency Layers ..................... 90
5.6 Experimental Results .............................................. 91
   5.6.1 Database ..................................................... 91
   5.6.2 Quantitative Analysis ........................................ 91
   5.6.3 Qualitative Analysis ......................................... 92
   5.6.4 Computation Time Analysis ................................. 94
5.7 Conclusion ........................................................ 94

# Chapter 6  Optimal Focal Plane Selection: A Downstream Application Using Light Field Saliency Estimation

6.1 Introduction ......................................................... 97
6.2 Background and Related Work ................................... 98
   6.2.1 Use of Defocus Blur in Multi-View Displays ................. 98
   6.2.2 Vergence-Accommodation Conflict .......................... 98
   6.2.3 Eye-Tracking and Dynamic Rendering for Virtual Reality .... 98
6.3 Method Overview .................................................. 103
6.4 Experimental Results .............................................. 107
   6.4.1 Qualitative Results .......................................... 107
   6.4.2 Quantitative Results ......................................... 113
6.5 Conclusion ........................................................ 114

# Chapter 7  Conclusion

7.1 Summary .......................................................... 116
7.2 Re-Addressing Research Questions ............................... 117
7.3 Future Work ....................................................... 120

# Bibliography

ix
# List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Summary of the light field salient object detection datasets</td>
<td>46</td>
</tr>
<tr>
<td>3.1</td>
<td>Entropy analysis results of the five light field rendering cases</td>
<td>60</td>
</tr>
<tr>
<td>3.2</td>
<td>$R^2$ values of the plots of recorded focal planes against the rendered focal planes for the focal-sweep renderings for each light field</td>
<td>66</td>
</tr>
<tr>
<td>4.1</td>
<td>Analysis of the FGSE method’s parameters$^\dagger$</td>
<td>76</td>
</tr>
<tr>
<td>4.2</td>
<td>Metric results$^\dagger$ for the FGSE method compared with the baseline shift &amp; sum saliency estimation (SSSE) without focus guidance.</td>
<td>77</td>
</tr>
<tr>
<td>4.3</td>
<td>Computation times for the FGSE method</td>
<td>80</td>
</tr>
<tr>
<td>4.4</td>
<td>Metric results for the FGSE method compared with FGSE method using only the centre SAI and using the centre SAI and four SAI at corners</td>
<td>82</td>
</tr>
<tr>
<td>4.5</td>
<td>Metric results$^\dagger$ for the FGSE method compared with FGSE method using only the centre SAI and using the centre SAI and four SAI at corners, and the baseline shift &amp; sum saliency estimation (SSSE) without focus guidance</td>
<td>82</td>
</tr>
<tr>
<td>5.1</td>
<td>Metric results of our FDLSE model vs. the FGSE model both evaluated against the ground truth data</td>
<td>91</td>
</tr>
<tr>
<td>5.2</td>
<td>Computation times for the FDLSE model</td>
<td>95</td>
</tr>
<tr>
<td>6.1</td>
<td>Comparison of the saliency weighted PSNR results of two combined slices images, one generated from equidistant slices of the focal stack and the other from our method’s selected slices</td>
<td>114</td>
</tr>
</tbody>
</table>
List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Two-plane parameterisation of a light field</td>
<td>9</td>
</tr>
<tr>
<td>2.2</td>
<td>Camera array light field acquisition</td>
<td>10</td>
</tr>
<tr>
<td>2.3</td>
<td>Camera on a gantry light field acquisition</td>
<td>11</td>
</tr>
<tr>
<td>2.4</td>
<td>Image formation in a light field camera compared to a traditional camera</td>
<td>11</td>
</tr>
<tr>
<td>2.5</td>
<td>Consumer-level and commercial light field plenoptic cameras and add-ons</td>
<td>12</td>
</tr>
<tr>
<td>2.6</td>
<td>Rendering the light field as a 2D array of perspective views</td>
<td>13</td>
</tr>
<tr>
<td>2.7</td>
<td>Visualisation of capturing a micro-lens image</td>
<td>14</td>
</tr>
<tr>
<td>2.8</td>
<td>Light field micro-lens image array, zoomed-in portion and micro-lens image</td>
<td>14</td>
</tr>
<tr>
<td>2.9</td>
<td>Epipolar plane images (EPI) representation of a light field</td>
<td>15</td>
</tr>
<tr>
<td>2.10</td>
<td>Light field focal stack</td>
<td>16</td>
</tr>
<tr>
<td>2.11</td>
<td>Reparameterising the light field by moving the sensors plane from $F$ to $F'$</td>
<td>16</td>
</tr>
<tr>
<td>2.12</td>
<td>Integral imaging when the eye accommodates close vs. far</td>
<td>18</td>
</tr>
<tr>
<td>2.13</td>
<td>Autostereoscopic 360-degree light field display [2]</td>
<td>18</td>
</tr>
<tr>
<td>2.15</td>
<td>Comparison between attenuation-based and polarization-based light field display</td>
<td>19</td>
</tr>
<tr>
<td>2.16</td>
<td>Multi-plane display using beam splitters [4][5]</td>
<td>20</td>
</tr>
<tr>
<td>2.17</td>
<td>Multi-plane display via time-multiplexing [6]</td>
<td>21</td>
</tr>
<tr>
<td>2.18</td>
<td>Linear-blending vs. optimal blending for multi-plane display [5]</td>
<td>21</td>
</tr>
<tr>
<td>2.19</td>
<td>Saliency map and corresponding colour density plot of a light field rendering</td>
<td>22</td>
</tr>
<tr>
<td>2.20</td>
<td>Visual saliency estimation categories</td>
<td>23</td>
</tr>
<tr>
<td>2.21</td>
<td>DeepGaze II Network [7]</td>
<td>26</td>
</tr>
<tr>
<td>2.22</td>
<td>EML Network [8]</td>
<td>27</td>
</tr>
<tr>
<td>2.23</td>
<td>UNISAL Network [9]</td>
<td>28</td>
</tr>
<tr>
<td>2.27</td>
<td>WSC Network [12]</td>
<td>34</td>
</tr>
<tr>
<td>2.28</td>
<td>DILF Network [12]</td>
<td>34</td>
</tr>
</tbody>
</table>
2.29 Saliency detection using relative location estimation

2.30 A two-stage Bayesian integration framework (BIF) [13]

2.31 MA Network [14]

2.32 SDDF Network [15]

2.33 SGDC Network [16]

2.34 RDFD Network [17]

2.35 DCA Network [18]

2.36 Architectures of existing deep learning light field salient object detection models

3.1 The Eyelink 1000 Plus experimental setup

3.2 Illustration of the effect of head movement and eye rotation on the position of the corneal reflection and pupil centre of a user’s eye on an image, as captured by the Eyelink sensor [19].

3.3 Light field rendering cases visualisation

3.4 Scanpaths of light field renderings

3.5 Light field renderings overlayed with their corresponding colour saliency maps

3.6 Light field renderings overlayed with their corresponding colour saliency maps generated from all participants and split over time into five 2-second segments.

3.7 Illustration of how $R^2$, the coefficient of determination, can capture the influence of the focus cue on eye fixation.

3.8 Plots of the recorded focal plane of fixation vs. the rendered focal plane for the focal-sweep renderings of the Tarot-Small, Table and Couch light fields.

3.9 Plots of the recorded focal plane of fixation vs. the rendered focal plane for the focal-sweep renderings of the Boardgames, Medieval and Pens light fields.

3.10 Density plots of the $R^2$ values using the fixation data.

4.1 Visualisation of the saliency estimation result of a state-of-the-art VA model DeepGaze II on an all-in-focus rendering and the result of our FGSE model on a refocused rendering of a light field.

4.2 Pipeline of the FGSE method for focus guided light field saliency estimation.

4.3 The qualitative results of our focus guided model (FGSE).

4.4 The output of our FGSE model using three different $\sigma_D$ values for the Region-2 rendering of the Boardgames light field.

4.5 The qualitative results of our focus guided model FGSE with downsampled multi-view array input.

5.1 Overview diagram of the proposed FDLSE method for light field saliency estimation using Fourier disparity layers.
5.2 Visualisation of the FDL layers ......................................................... 88
5.3 The qualitative results of our FDLSE model ....................................... 93

6.1 Demonstration of the vergence-accommodation conflict for near and far scenarios .................................................. 99
6.2 Illustration of vergence and accommodation distances matched by adjusting the virtual image distance [20] .......................................................... 99
6.3 Gaze-contingent focus [20] ................................................................. 100
6.4 Varifocal display with an eye-tracker [21] ........................................... 101
6.5 Accommodation-invariant near eye display [22] .................................. 101
6.6 Multi-focal display with dense focal stacks [21] ................................102
6.7 Overview diagram of the proposed method for selecting optimal focal slices for a downstream gaze contingent blur application ............................................. 104
6.8 Probability density function and cumulative distribution function for the Dino light field ........................................................................... 105
6.9 Plots for six light field examples showing how the cumulative distribution function (c.d.f) is used to select focal stack slices according to saliency ........... 106
6.10 Side-by-side comparison, for the Lego Knights and Platonic light fields, of the six focal stack slices selected using our method to those sampled with equidistant spacing ................................................ 108
6.11 Side-by-side comparison, for the Table and Treasure light fields, of the six focal stack slices selected using our method to those sampled with equidistant spacing 109
6.12 Side-by-side comparison, for the Pens and Dishes light fields, of the six focal stack slices selected using our method to those sampled with equidistant spacing 110
6.13 Side-by-side comparison of the ‘combined images’ generated from slices selected by our method to those generated from equidistant slices ............... 111
6.14 The focus depth map generated for six slices selected using our method to those generated with equidistant spacing, for each light field .................... 112
Acronyms

AUC  Area Under the Curve.

CC  Pearson’s Correlation Coefficient.

CNN  Convolutional Neural Network.

DCV  Disparity Cost Volume.

DOF  Degrees of Freedom.

DSLR  Digital Single-Lens Reflex.

FDL  Fourier Disparity Layer.

FDLSE  Fourier Disparity Layer Saliency Estimation.

FGSE  Focus Guided Saliency Estimation.

fps  Frames per Second.

GAN  Generative Adversarial Network.

GT  Ground Truth.

HMD  Head Mounted Display.

KLD  Kullback-Leibler Divergence.

LF  Light Field.

NSS  Normalised Scanpath Saliency.

PSNR  Peak Signal to Noise Ratio.
ReLU  Rectified Linear Unit.

RGB  Red Green Blue.

SAI  Sub-Aperture Image.

SIM  Similarity.

SOD  Salient Object Detection.

SSSE  Shift and Sum Saliency Estimation (Without Focus Guidance).

VA  Visual Attention.
Chapter 1

Introduction

New developments in capture [23] and display technologies [24, 25] has propelled research into an advanced form of visual media representation - the light field. In contrast to traditional imaging systems, which capture a 3D scene by projecting it onto a 2D surface, light fields [26, 27] encode both the angular and spatial coordinates, as well as the intensity information of light rays travelling within a 3D-space.

The effective design of visual computing systems depends heavily on the anticipation of human attention. While visual saliency, a term which generally describes the perceived important regions of a scene which stand out in the scene context, is well investigated for conventional 2D images and video, it is nevertheless a very active research area for emerging immersive media. In particular, visual saliency of light fields has only recently become a focus of research. Additionally visual attention (i.e. where people look when viewing a visual scene) a sub-field of visual saliency, has never been previously investigated for light fields to our knowledge. We have treated visual attention as a person-specific measurement in relation to an image. Specifically, for a given image and person the visual attention of that person in relation to that image at a given time is the set of spatial coordinates which the fovea of the eye is centred on for a period of time. For a more formal and in-depth overview of the details of how visual attention is defined, we refer the reader to sections 2.2.1, 2.2.2 and 3.2.1.

As they may be rendered and consumed in various ways, a primary challenge that arises is the definition of what visual attention of light field content should be. The work presented in this thesis aims to address this problem by firstly, analysing the eye fixation patterns of humans viewing light field renderings and secondly, using this analysis to model light field saliency as a 4D saliency field analogous to the light field itself. Furthermore, to demonstrate the effectiveness of this saliency field approach we built two models for visual attention prediction with the saliency field at their cores and analysed the results.

As a 4D media type, light fields can be viewed using multi-view technology such as vol-
umetric light field displays and head-mounted-displays, for example. Additionally, they can be rendered in 2D in different ways by adjusting the angle, focus and aperture parameters to refocus and present virtual views. In this work, we have explored the potential use of visual attention prediction of light fields in the design of these devices.

1.1 Motivation for Light Field Visual Attention Research

As humans are the ultimate consumers of multimedia systems, visual attention is invaluable for many applications and aspects of the multimedia content delivery chain. This is especially true for applications that involve human-computer interaction such as gaze prediction, eye tracker calibration, stereoscopic disparity manipulation among others. However, the current state-of-the-art in light field saliency prediction models are all object-based models. Information related to eye fixations cannot be inferred from object-based saliency models, as the main task for these models is to segment the dominant objects of a scene and so, they are not suitable for the aforementioned tasks. In general, for salient object detection models, ground truth data is collected using human segmentation of images and does not verify that objects chosen as salient regions are in fact the areas where eyes will fixate. Likewise, they don’t account for attention in the pre-attentive stage, when objects are not yet recognised. They further fail to capture all aspects of visual interest, for example how salient non-object regions such as text command attention.

Therefore, to allow the development of these applications, creating a foundation for the understanding and estimation of light field visual attention is crucial. For this purpose, in this thesis, we explore visual attention and eye fixation prediction for light fields. In addition, we account for the shortcomings in light field visual saliency research and instead address all spatial locations that attract visual attention, be they regions, objects or points of interest. Eye fixation data is collected using eye-tracking devices and thus, provides a more meaningful form of ground truth for visual attention compared to binary maps, since it represents the statistical distribution of fixation data. Saliency maps [28], which are continuous density maps that represent the probability of fixation at every point in an image, can be computed from this data. They can be analysed to understand how the visual perception of light fields differs from that of traditionally captured photographs and conventional images with red, green and blue (RGB) colour channels. To this end, we observe how attention is affected, with an emphasis on refocusing, which is a distinctive feature of light field technology. We then use this as a base component in our models’ architecture.

Light fields’ (LF) higher dimensionality relative to other media types introduces computational complexity. This presents challenges for applications such as real-time rendering and LF streaming when resources are limited. For example, Visual Attention (VA) prediction can
be used to compress the LFs by highly compressing regions that aren’t relevant according to VA, reducing the computational and memory bottlenecks for these applications. The values of regions of the predicted LF saliency field can thus be used as an automated ranking mechanism to make rendering decisions for a range of tasks such as compression and quality assessment for light field rendering. These tasks require that every point in the light field has an associated importance ranking. A 4D continuous representation is necessary as the light field is a 4D continuous space and a continuous valued ranking would be more versatile than a discrete one. Since the output of visual attention models are values sampled from the continuous probability space they present a viable option provided they can be extended to 4D space.

1.2 Research Questions

Our research questions were chiefly motivated by the following:

1. Current research into visual saliency prediction for light fields has a very strong focus on salient object detection [29]. We have hypothesised that when using renderings from light fields, objects in the rendering only account for a part of the resulting saliency map for that image in relation to visual attention. Therefore, we set out our research questions to investigate this.

2. The previous salient object detection methods for light field saliency prediction were all extensions of saliency work for 2D images, using a limited number of various light field aspects such as focal stacks, depth maps, multi-view images, etc. [29] to boost performance. However, since light field renderings are fundamentally different to traditional renderings, we viewed these works as under-utilising the 4D light field specific aspects. This motivated us to ask if new models could be developed to more strongly reflect the intrinsic properties of light fields, with the aim of achieving high performance and offering greater interpretability.

3. Some real-world applications would greatly benefit from the ability to create saliency maps of light field renderings, which can then be used for a plethora of downstream applications such as ranking and selection mechanisms for light field renderings. Prior to the commencement of this work, this was not readily available as salient object detection models typically only offered this in a binary sense, since the object(s) are only defined as salient (1) or not (0) [11].
Based on this knowledge, we developed the following research questions:

Q1. How can we quantitatively and qualitatively estimate how phenomena specific to light fields affect visual attention?

Q2. How can we build a model to predict the visual attention of light fields which uses intrinsic properties specific to light fields, thereby providing interpretability as well as high prediction accuracy for a suitable evaluation methodology?

Q3. How can the models proposed in Q2 be used to generate saliency maps of light field renderings, which in turn can be applied to downstream applications such as ranking and selection mechanisms for light field renderings?

1.3 Thesis Scope

The main focus of this thesis was to explore the intersection of light fields and visual attention and thus, design a model for light field visual attention estimation and show its value for use in a practical application. Ultimately, in deciding the scope of this thesis, we wanted to be able to establish whether the visual attention of light fields is different to normal 2D images and to investigate whether a model which incorporates light field specifics aspects (refocusing), can be built to predict the saliency of light field renderings in a more integrated way than a state-of-the-art image visual attention model.

In Chapter 2 we provide a general overview of the broader areas of light fields and visual saliency. The background goes as far as giving readers a fundamental understanding of light fields, from how they are derived and parameterised to examples of light field acquisition, rendering and display. We chose examples based on their relevance to the work in this thesis as well as their prevalence in the literature. Similarly, to help readers with the basic concepts of visual saliency, we cover the definitions of visual saliency and visual attention as well as the three main ways to categorise saliency estimation models. To give an understanding of the existing research in this area that is also applicable to this thesis, we outline both the traditional and deep learning style architectures, and summarise the top four models in image visual attention prediction and the traditional and state-of-the-art in light field visual saliency. For completeness, we detail and categorise all the existing light field visual saliency estimation models, to our knowledge, as well as the corresponding light field saliency datasets. While newer metrics for visual attention have been explored, we cover only the five most commonly used visual attention evaluation metrics in this thesis, since these are used in the MIT/Tübingen Saliency Benchmark.
To collect data for our light field visual attention dataset, we used only one eye-tracker, the Eyelink 1000 Plus [19]. We chose to include 20 light fields and three light field renderings according to the criteria outlined in Chapter 3. We limited the data to four acquisition types with the light fields represented as a multi-view array. However, other light field capture or representation types such as spherical representations, for example, would likely influence the attention of users and therefore, the model architecture would need to be constructed to reflect this different representation. In this thesis, we also focus predominantly on the influence of a tunable focus cue, something which is characteristic to light fields. This has impacted the structure of our models and therefore, to predict the saliency of other novel light field renderings the architecture may need to be adjusted. Chapter 3 also provides readers with a basic knowledge of the three main types of fixational eye movements. Although other methods of visualising saliency data exist, in our analysis, we examine three ways of generating saliency visualisations. Since some of the renderings varied over time, we analysed the saliency data we collected both over the full data collection period and over shorter equal time segments. This was done from both a quantitative and qualitative perspective.

For our light field visual attention model, we considered two popular light field refocus rendering algorithms and one image saliency estimator, DeepGaze II [7], to act as a basis for our models’ architecture. To demonstrate the benefit of our model in practise, we outline its use in one downstream application, namely optimal focal plane selection, to alleviate some of the problems caused by the vergence-accommodation conflict in head mounted displays. We give an overview of the vergence-accommodation conflict as well as the main areas of research that aim to resolve this conflict using eye-tracking and dynamic rendering. We focus on developing adaptive depth of field rendering software as this did not require access to specialised devices.

1.4 Thesis Outline

The thesis is structured into seven chapters. In Chapter 2 we give an overview of light fields and visual saliency, with details such as light field terminology, categories of visual saliency, saliency estimation and visual attention prediction of images all outlined. We then detail the different light field capture and display technologies as well as the current state-of-the-art methods for visual attention prediction in images and light field saliency prediction. Datasets which have been used for these methods are outlined in terms of scale, capture device used, spatial resolution, focal stack size used as well as data types provided. Finally, evaluation metrics which can be used for light field saliency estimation are also outlined. This chapter serves as necessary background information for the work presented in the rest of the thesis.

Chapter 3 mainly contains the methodology and implementation details of a subjective experiment we carried out to gather gaze data in relation to light fields. The methodology and
The rationale used in previous studies is discussed, providing a foundation for our methodology. The details of our recording equipment and experimental set-up are all provided so that the experiment can be re-produced. We then give details of how we specifically chose the light fields used in our study to give a more diverse set of data in relation to previous experiments. Quantitative analysis of saliency maps and scanpaths resulting from the study are provided, with particular emphasis on the analysis of the influence of the focus cue and distribution of fixations. This chapter principally addresses Q1.

Chapter 4 is responsible for providing the details of our focus guided saliency estimation model (FGSE), which predicts the visual attention of any light field rendering. We proposed this model as a novel way to generate saliency maps for light fields which doesn’t rely on salient object estimation. The model uses both the multi-view array and depth map information to generate a saliency map for any rendering of the light field. The technical details for the model are provided, as well as our results using the dataset we gathered in Chapter 3.

Chapter 5 contains the details of our Fourier Disparity Layer Saliency Estimation (FDLSE) model, which is an alternative to the FGSE approach described in Chapter 4. This model represents a more concise approach to the prediction of visual attention, since it doesn’t require the full view array or pre-estimated disparity maps. This mirrors Chapter 4, where the technical details of the model are provided, as well as our experimental results.

Chapter 6 gives the details of an application of the dataset and saliency models outlined in previous chapters: how to rank and select refocus renderings for use in gaze contingent blur of light field data to resolve the vergence-accommodation problem. The intuition and technical details of this problem are first outlined. We then describe our proposed solution to this problem, which involves using the FDLSE saliency model to choose suitable focal slices to use. Quantitative and qualitative results are provided.

Finally, in the conclusion we re-address the research questions set out in Chapter 1. We summarise the work done as part of this thesis and give an outline of future research which could be done in this area, building on the groundwork laid by our work.

1.5 Main Contribution and Publications

We present below the main contributions of this work along with the corresponding recent publications.

- In Chapter 3 we planned the methodology and executed a subjective experiment which led to the creation of the first database of temporal eye fixation data for focally varying renderings of 20 lights fields.
  [31] A. Gill, E. Zerman, C. Ozcinar, and A. Smolic, “A study on visual perception of light

- In Chapter 4 we designed, implemented and evaluated a novel visual attention prediction model which took into account specific aspects of light fields.


- In Chapter 5 we designed, implemented and evaluated another visual attention prediction model for light fields using a more unified Fourier disparity layer framework.


- In Chapter 6 we investigated the application of our saliency field model, FDLSE, to select optimal focal planes, based on estimated saliency, for the gaze contingent blur rendering of light fields. We anticipate that this research will be published in a journal on image processing and human visual perception, such as IEEE Transactions on Image Processing, IEEE Transactions on Multimedia, or ACM Transactions of Applied Perception.
Chapter 2

Background

This chapter outlines the requisite background knowledge for the subsequent methodology chapters. An introduction to light fields, refocusing, visual saliency and display methods is provided. Methods for saliency prediction specifically for light fields are then outlined and reviewed. Existing datasets are discussed and their different features are highlighted. Finally, evaluation metrics commonly used for saliency are explained.

2.1 Light Fields

Light field (LF) imaging technology lies in the intersection of the fields of computer vision and computer graphics and aims to capture the amount of light travelling in every direction through every point in space. The light field is derived from the plenoptic function $P(\theta, \phi, \lambda, t, x, y, z)$ that assigns a radiance value to rays propagating in all directions in 3D space. In formal terms, $P(\theta, \phi, \lambda, t, x, y, z)$ represents the spectral radiance per unit time, where $(\theta, \phi)$ denotes an incident direction, $\lambda$ the light ray’s wavelength, $t$ denotes time and $(x, y, z)$ a spatial position.

The plenoptic function takes into consideration all possible variations of light at all points in space, directions, wavelengths and points in time. For light fields, however, the wavelength of each ray of light is recorded in each of the colour channels independently. Additionally, light fields are most commonly considered to be static (although research is being extended to create video light fields). Therefore, light fields are assumed to be monochromatic and time invariant. There is also the assumption that the light field is measured in “free space” and that the light ray radiance remains constant along the rays in the region outside the measured scene. Taking these assumptions into account, light fields can then be represented as a 4D function $L(u, v, s, t)$. The most common representation of a light field is by the coordinates of their intersec-
tions with two parallel planes placed at arbitrary positions proposed by Levoy and Hanrahan in 1996 [26].

Figure 2.1: Two-plane parameterisation of a light field

Two planes, the focal plane \((s, t)\) and the camera plane \((u, v)\), contain the spatial and angular information of the light rays respectively. A light ray in the system intersects the \(uv\) plane at \((u, v)\) and the \(st\) plane at \((s, t)\) giving \(L(u, v, s, t)\). This is known as 2-plane parameterisation of a light field and can be seen in Fig. 2.1.

2.1.1 Light Field Acquisition

Camera Array

One way to capture a light field is to arrange multiple cameras into a 2D array. Each camera captures a sub-aperture image and the 2D array of captures forms a multi-view array. Light field captures using camera arrays enable applications for which a large parallax between the different views is beneficial such as tracking through occlusions [37], multi-object detection [38] and reconstructing occluding surfaces [39]. A downside to these systems is that they are often very large which makes them much less portable.

The Stanford Multi-Camera array [40] was an early camera array system built from 128 VGA video cameras controlled by four computers. With this setup, it was possible to arrange the cameras in a number of configurations. One arrangement of 100 cameras an inch apart, as seen in Fig. 2.2a, allowed the system to function as a multiple-center-of-projection camera and thus, capture a light field. Another arrangement, with the cameras 9-inches apart, in an arc, and facing the centre of a room, allowed them to capture a video light field which could subsequently be used for view interpolation or shape estimation.

Sabater et al. [41] presented a multi-view light field video capture and processing pipeline. They used a 4x4 fully synchronised camera array consisting of 16 digital single-lens reflex
(DSLR) cameras with a 12mm lens (50° × 37° field of view) which capture video at 30 fps with 2048 × 1088 resolution. There camera rig can be seen in Fig. 2.2b.

Most recently, Herfet et al. [42] introduced Saarland University’s light field camera array, shown in Fig. 2.2c, which involved 64 full-HD (1920x1200 resolution) cameras capable of capturing video at 41 frames per second (fps). These cameras can be arranged in a variety of ways, at different distances apart, and the sampling phase of the cameras can be set, with an exactness of 10 µs, so that the cameras trigger at different times. A subsequent work [36], explored the use of an 8x8 setup of this camera array in media productions. For their test scenario, capturing a cellist’s performance, all cameras were shuttered synchronously, using identical frame rates and sampling phases. The light field video data captured with this setup allows for novel post-production simulated lens effects such as tilt-shifts and refocusing, and can be used to generate high quality, dynamic depth maps and geometric reconstructions of the scene.

**Gantry-Controlled Camera**

For static scenes, the light field perspective views can also be captured by translating a single gantry-controlled camera across a plane.

The Stanford research group [40] have built three gantries. The purpose of the first was for 3D scanning in the Digital Michelangelo project [43]. The setup, as in Fig. 2.3a, involved mounting a digital camera instead of a 3D scanner which allowed four degrees of freedom for movement i.e. (x,y) translation, nod and shake. The motions were repeatable and accurate to within less than a millimeter. The second was the Lego Mindstorms gantry, shown in Fig. 2.3b, which allowed for two translational motions. The accuracy was close to that of the first light field gantry, while still remaining portable and having a simple construction. To capture spherical light fields, they designed a third spherical gantry with four motorised motions as shown in
Fig. 2.3c. It consisted of an inner arm which held the camera and an outer arm which grasped a light source or video projector. The spherical gantry captured light emanating in all directions from a central subject.

The Disney Research group’s gantry was composed of a DSLR camera with a 50 mm lens and a high 21 megapixel resolution mounted onto a motorized 1.5 metre long linear stage [44]. As the camera was computer operated, they could space the captures with high accuracy. A typical capture session involved taking 100 images of the scene with a 10mm uniform spacing. For each image, they allowed two minutes for capture (including moving and stopping) to avoid motion blur and to obtain higher image resolution.

These gantries are considered to be cheaper and simpler to set up than camera arrays. They also have an advantage over plenoptic cameras in terms of higher resolution.

**Plenoptic camera**

In contrast to conventional 2D image cameras, light field plenoptic cameras have an array of micro-lenses added between the main lens and the sensor. This is illustrated in Fig. 2.4

![Plenoptic camera diagram](Figure 2.4: a) Light field camera and b) Traditional camera)
This allows for the angular \((u, v)\) information of light rays to be recorded in addition to their spatial position \((s, t)\) and intensity. Without considering the main field lens, from its array position, each micro-lens records the perspective view it observes of the scene. Thus, a light field is captured with a uv resolution related to the number of micro lenses and an st resolution according to the number of pixels behind each micro-lens. However, the field lens transposes the light field which causes the st resolution to depend on the number of micro-lenses and vice versa for the uv plane. Using the field lens is preferred since only one lens (this field lens) needs to be corrected for aberrations.

An important aspect of the plenoptic camera is its capability to perform synthetic aperture photography commonly known as refocusing. This allows for a photograph taken to be refocused post-capture, more details will be given below in Section 2.1.2.

To obtain perspective views (sub-aperture images) from the raw light field data captured, post-processing is needed in the form of decoding, calibration and rectification [45]. Furthermore, colour correction [46] techniques have recently been developed for the raw camera output, to improve the colour consistency across the full array of perspective views (multi-view array). The terms multi-view and sub-aperture will be defined in more detail below in Section 2.1.2

![Figure 2.5: Consumer-level and commercial light field plenoptic cameras and add-ons](image)

In 2006, Ren Ng founded what eventually became Lytro Inc., based on his PhD research on digital light field photography [50][51]. The full thesis can be found in the footnote below. This company introduced the first consumer-marketed plenoptic camera Lytro [52] in 2010, as seen in Fig. 2.5a and introduced a professional-level camera the Lytro Illum in 2014, presented in Fig. 2.5b. The Lytro Illum is a professional-grade light field camera with a 40,000 microlens array, 4 inch touchscreen, removable battery and SD card slot. The raw data captured by the Lytro camera is output in the form of LFP files and in the case of Lytro Illum LFR files. These files can be processed using Lytro desktop software [47] or LFToolbox [48] to obtain light field images.

Although the company ceased operations in 2018 its products are still being used in research as

recently as 2019 [53].

In 2010, another company Raytrix [48] introduced the first commercial light field camera, the R11 plenoptic camera. This was subsequently updated to the R29 camera which captures images at 7.25 megapixel resolution at 10 fps, shown here in Fig. 2.5c. This is a modified Lumenera camera with a 40,000 microlens array, each lens having a diameter of 200 μm, which gives a resolution of 3 megapixels. It can also capture full resolution video at 3.5 fps.

More recently, the K-lens was introduced by Manakov et al. [49] which can be attached to a DSLR camera allowing it to acquire a light field capture. This device is shown in Fig. 2.5d.

Compared to the other light field acquisition systems the light field camera is the most portable. However, there are disadvantages to these cameras: the range of viewpoints captured only spans a small depth (inches to microns) and they capture much smaller spatial resolutions.

2.1.2 Light Field Image Renderings

In this section, we describe the most common image renderings of light fields along with figures and diagrams for visualisation.

**Sub-Aperture Images and the Multi-View Array**

Light fields are often captured and visualised as a multi-view array, a 2D array of \( U \times V \) sheared perspective views of the scene [26]. Each view, also called a sub-aperture image (SAI), represents a 2D slice of the 4D light field at a fixed point \((u, v)\). Each sub-aperture image has \( S \times T \) spatial resolution. Fig. 2.6 shows a sample of a light field multi-view array for visualisation.

(a) Rays arriving at a fixed point on the camera plane from all points on the focal plane
(b) Light field multi-view array (sub-aperture images sampled from the Platonic light field multi-view array [54] for visualisation)

Figure 2.6: Rendering the light field as a 2D array of perspective views
Micro-Lens Images

This is the image recorded from a single microlens in the microlens array, on the plenoptic camera sensor. Fig. 2.7 demonstrates how the micro-lens image can be captured by a plenoptic camera. Micro-lens images can be generated from the light field by sampling the spatial \((s, t)\) dimensions. A single micro-lens image is produced by keeping the angular dimension \((u, v)\) fixed, as follows: rays leaving a point on a subject are focused by the main lens onto a microlens of the micro-lens array and subsequently arrive on the sensor of the plenoptic camera as a micro-lens image.

Figure 2.7: Visualisation of capturing a micro-lens image

![Micro-lens image array and sensor](image)

Figure 2.8: Light field micro-lens image array, zoomed-in portion and micro-lens image

Fig. 2.8 shows both a micro-lens image array and micro-lens image. Micro-lens images look like reflectance maps as seen in the zoomed-in portion of the micro-lens image array. This occurs because the subject has been placed astride the focal plane, observe Fig. 2.4(a). This
makes sets of rays leaving a micro-lens which are then captured on the photosensor, similar in character to sets of rays leaving points on the subject [26].

**Epipolar Images**

![Epipolar Images](image)

Figure 2.9: Epipolar plane images (EPI) representation for the Platonic light field [54]

Another common representation of LFs are Epipolar Plane Images (EPI) [55], which are 2D slices of the 4D light field obtained by fixing one spatial and one angular dimension (su- or vt-planes). An example of two epipolar images can be seen in Fig. 2.9.

**Light Field Refocusing and the Focal Stack**

Light field refocusing is the task of rendering a light field such that a certain focal plane appears sharp or-in-focus, while the rest appears blurry or out-of-focus. The challenge of refocusing led to a novel way of representing light fields, commonly known as the focal stack. The focal stack consists of a sequence of refocus renderings. In this context, the refocus renderings are called “focal slices” and can be thought of as slices of the light field focused at consecutive depths.

Ng. et al’s paper [50], first outlined a digital refocusing technique for light fields. Since this paper defines refocusing with the plenoptic camera in mind, they describe the two-plane parameterisation of the light field in terms of the lens plane (u,v) and the sensors plane (x,y). To obtain a refocused image $I_{\alpha F}$ “in-focus” at image distance $F'$ it is necessary to first reparameterise to the light field $L_{\alpha F}$ in the refocus plane located at $F'$. A diagram is given in Fig 2.11.
This reparameterisation to the light field \( L_{\alpha F} \) in the refocus plane, can be written as:

\[
L_{\alpha F}(u, v, x, y) = L_F(u, v, (u + \frac{x - u}{\alpha}), (v + \frac{y - v}{\alpha})) = L_F(u, v, u(1 - \frac{1}{\alpha}) + \frac{x}{\alpha}, v(1 - \frac{1}{\alpha}) + \frac{y}{\alpha})
\]

where \( F' = \alpha F \), \( F \) is the focal distance of the main lens, \( L_F \) is the light field in the plane of the sensor and \( \alpha = \frac{F'}{F} \) is a factor that controls the image distance. The sharp image \( I_{\alpha F} \) can be generated by integrating rays in the refocus plane. Hence, the intensity at each pixel \((x, y)\) of the sharp image can be calculated as

\[
I_{\alpha F}(x, y) = \frac{1}{\alpha^2 F^2} \iint L_{\alpha F}(u, v, x, y) \, du \, dv.
\]

Different algorithms can be used to approximate \( I_{\alpha F} \) such as the shift-and-sum algorithm described in [50], or by using the Fourier transform [51, 56] or other algorithms [57, 58, 59]. Two such algorithms will be discussed further in Chapters 4 and 5.
2.1.3 Light Field and Multi-View Displays

Light fields, with respect to display, are closely related to holograms which are defined as wavefronts created by the interference of coherent light. The aim of the light field display is to present 3D content in a way that provides a naturalistic viewing experience. They often fall under two use cases: for multiple users such as multi-projector and multi-layer displays, or for personal use as head-mounted displays (HMDs) and near-eye displays. Common displays of 3D content are virtual reality (VR) and augmented reality (AR) displays. Typically these are head-mounted stereoscopic displays. Stereoscopic displays are image-based displays that render a pair of views, one for each eye, with a disparity between them to facilitate stereo parallax [60]. This binocular disparity drives fusional vergence which gives the impression that virtual objects are at varying depths. However, these displays render images at a fixed display distance and fail to simulate accommodative blur, thus, a viewer accommodates to a fixed distance for all virtual objects displayed. This gives rise to the vergence-accommodation conflict [60] which we discuss in more detail in Chapter 6. In recent years, there have been advances in light field display technology as a means to solve this conflict. Thus, light field displays can be categorised according to their approach in tackling this problem [61, 60] into multi-scopic displays which use a ray-based integral imaging, and multi-plane displays which use an image-based multiple focal plane approach. In this section, we outline these two approaches and describe some notable designs.

Multi-Scopic Light Field Displays

Multi-scopic displays integrate rays from multiple views projecting them onto each eye and are thus categorised as ray-based displays. This generates what appears to be a continuous light field. They employ a technique known as integral imaging. When using this technique, multiple light rays corresponding to the same point in a virtual scene are generated. This is equivalent to displaying multiple viewpoints of the same scene with a translational offset. Light rays from a point on that scene are guided to the eye in a way that emulates a cone of light fanning out from that point. These rays intersect within the eye at different depths. This allows for the eye to accommodate and bring observed points into focus, blurring the rest.

The autostereoscopic 360-degree Light Field Display [2] is one such display which renders interactive graphics at 5,000 images per second, updating 20 times per second. The display involves a high speed video projector, a motion-control motor, and a spinning mirror covered by a holographic diffuser. The mirror is tilted at 45° to reflect rays of light from the projector to all possible viewing positions 360° around the device. The “auto” in the name refers to the glasses-free viewing. When viewing the display from any point, their multiple-center-of-projection rendering technique displays images with correct perspective and parallax. An image
Figure 2.12: Integral imaging (a) Eye accommodates close: Rays emanating from the blue square intersect at the retina such that it appears in-focus. Rays from the green circle intersect at a point in front of the retina resulting in a circle of confusion marked as c which emulates retinal blur. (b) Eye accommodates far: Rays emanating from the green circle intersect at the retina appearing sharp and rays from the blue square intersect at a point behind the retina appearing blurred [60].

(a) A 3D object shown on the display  
(b) Visualisation of the intersection \( M \) of a vertically diffused ray of light with the circular locus of viewpoints \( V_1, V_2 \) etc.

Figure 2.13: Autostereoscopic 360-degree light field display [2]

and visualisation of the device can be seen in Fig. 2.13.

In 2013, Nvidia produced a multi-scopic near-eye light field display [3] prototype which makes use of a micro-lens array to split images produced by an OLED screen into individual light rays. This generates a light field directly in front of a user’s eyes, allowing them to accommodate their eye to any depth.

A Polarization Field Display [62] was presented by Lanman et al. which displays the light field using multi-layered liquid crystal displays (LCDs). This is an automultiscopic display
which consists of stacked LCD panels enclosed by one pair of crossed linear polarizers. Each panel is considered a spatially-controllable polarization rotator. To dynamically display the light field in real-time, tomographic algorithms are employed to find optimal spatially-varying polarization state rotations to apply to each panel. To achieve colour in the display, they use field sequential colour illumination with monochromatic LCDs. This is different to the preceding attenuation-based light field display, which contains polarizers between each layer and acts as a programmable transparency stack. This polarization field design showed improved brightness, resolution and extended depth-of-field over attenuation-based light field displays. A comparison between the two can be seen in Fig. 2.15.
Multi-Plane Displays

Volumetric displays ensure eye vergence, accommodation and full parallax. Light sources called voxels are placed throughout a volume of space which are transparent in an off-state and either luminous or opaque in an on-state. However, the voxels of light lead to limitations in resolution and a glow is generated from additive light. Multi-plane displays are a variation on volumetric displays with a fixed viewpoint.

![Multi-plane display using beam splitters](Image)

Figure 2.16: Multi-plane display using beam splitters [4] (Image taken from [5])

Within multi-plane displays, multiple image planes are stacked, each focused at a certain depth. Different parts of an image are rendered to each plane simultaneously. For each plane, the pixels of the plane that correspond to objects/regions at that plane’s depth in the virtual scene are rendered, the other pixels of the plane are left transparent. Similar to volumetric displays light arrives to the eye from sources placed at the correct focal distance and so multi-plane displays achieve near-correct vergence and accommodation cues. Unlike the multi-scopic display, it only works for a single viewpoint at a time.

These virtual image planes can be approximated using spatial or temporal multiplexing. The spatial multiplexing approach uses beam splitter to superimpose images of different parts of a monitor screen additively along the axis of vision. Akeley et al. [4] designed a prototype 3-plane spatial multiplexing system with a high resolution display, as illustrated in Fig 2.16.

For time multiplexing, the virtual distance of the monitor can be adjusted with high speed switchable lenses. The image is time-multiplexed in-sync with the lenses to give the illusion of multiple focal planes in front of the viewer. A time-multiplexed multi-plane display was developed by Love et al. [6] and is shown here in Fig 2.17. Furthermore, near-eye multi-plane displays have been developed in recent work for both virtual and augmented reality [63, 64].

To allow for continuous depth in between the display planes, depth-weighted interpolation
can be used. Although linear interpolation has long been proposed [6], recently Narain et al. has put forward a non-linear optimized blending method that considers both the content itself and the visual system response [5]. This optimized blending is visualised in Fig. 2.18. Although there is a major limitation on viewpoints, multi-plane displays have been found to provide
a comfortable viewing experience [6]. In addition, the majority of multi-plane displays are capable of high resolution, utilizing the full resolution of the monitor used. Thus, image quality is only restricted by computational resources.

2.2 Visual Saliency

Visual saliency is the subjective term describing perceived pertinent regions or elements of a scene which stand out in the scene context. It has been studied for both images [65] and videos [66] on planar surfaces. Saliency can be in relation to the entire scene, objects within a scene etc. We provide an overview of the related literature and an overview of the various ways saliency is typically described in the following sections.

![Saliency map and corresponding colour density plot for the Dino light field](image)

Figure 2.19: Saliency map and corresponding colour density plot for the Dino light field [54]

2.2.1 Saliency Estimation Models

Saliency estimation is a fundamental task in image processing and computer vision, aiming to predict regions of a scene that standout in the scene context. There are many different definitions of visual saliency estimation, each being more or less relevant depending on the desired task [67]. The three main ways of categorising saliency estimation models are as follows:

1. **Bottom-Up vs. Top-Down models:** A saliency model is referred to as bottom-up if it is stimulus-driven. It aims to predict what characteristics of a scene influence visual attention. Top-down models on the other hand are task-driven. They investigate the effect that cues, such as experience, current goals, expectation and reward, have on where people look when viewing a scene.

2. **Static vs. Dynamic models:** Static models rely solely on the spatial aspects of a still scene. Dynamic models aim to capture the spatio-temporal aspects of visual attention.
For example, how to represent the effect changing position of stimuli over time has on attention.

3. **Pixel-Based vs. Object-Based**: Pixel-based saliency models attend to the spatial locations that attract attention be they regions or points of interest. Object-based saliency focuses on localising dominant object instances of a scene. This involves a level of semantic understanding of a scene to confirm both the existence and location of objects.

![Visual Saliency Estimation Categories](image)

As described in the third categorisation above, visual saliency estimation models can be subdivided into pixel-based and object-based models. These two sub-fields of saliency estimation are known as: visual attention prediction (also known as eye fixation prediction) and salient object detection.

Visual attention (VA) is colloquially defined as where people look when shown an image, and in more definite terms represents, for a given person, visual stimulus, and time duration, the set of spatial coordinates which the fovea of the person’s eye fixates on. Thus, for visual attention prediction, the aim is to estimate a saliency map which represents the likelihood of each pixel to attract the human eye. These models are trained and evaluated on their ability to predict visual attention by using eye-tracking data as ground truth. From this data ground truth data, fixation maps can be generated which are binary maps with ones at fixation coordinates and zeros elsewhere. Subsequently, ground truth saliency maps can be created by convolving fixation maps with a Gaussian filter. Thus, these saliency maps are viewed as probability density maps of eye fixations. This is visualised in Fig 2.19.

Salient object detection (SOD) is the task of detecting and segmenting the visually prominent (salient) object(s) of a scene. For SOD, the goal is to generate a binary map with the segmented object(s) regions/ pixels assigned a value of “one” and all other regions a “zero”.

23
These two sub-fields of saliency have uses in different applications. SOD is commonly used for multimedia analysis. The SOD saliency maps with estimated segments of salient objects, can be used in object detection and recognition, or semantic segmentation to classify the salient objects of a scene [68]. Visual attention prediction on the other hand can be used, for example, in compression [69], quality assessment [70] and content creation [71].

For the research detailed in this thesis, we were concerned with all regions that draw a viewer’s gaze. Therefore, visual attention prediction was more suitable for our research and so this is the definition we use for our saliency estimation models in Chapter 4 and Chapter 5.

2.2.2 Visual Attention Prediction in Images

Humans have been found to fixate on regions with greater edge density and local contrast [72]. Low level features such as intensity, orientation and color contrast have been found to guide visual attention [73] as well as high level features like faces [74, 75]. Viewers in visual attention experiments tend to move their gaze to targets near where they are currently looking. There is also a tendency to look at centrally located targets in their field of view [76]. Typical approaches when modelling eye fixation prediction include:

Contrast Based Models:

The majority of visual saliency models for gaze prediction use center-surround contrast and image statistics to identify salient regions that are complex or rare in appearance. Itti et al’s model [73] formed the basis of later models. An image is subsampled into a Gaussian pyramid. Each pyramid level \( \sigma \) is decomposed into channels for image features \( l \): colour (\( C \)), intensity (\( I \)) and orientation (\( O \)). These are used to construct normalised center-surround feature maps \( f_l = N(\sum_{i=2}^{4} \sum_{c=s+c}^{s+c+3} f_{l,c,s}) \), \( \forall l \in L_I \cup L_C \cup L_O \) where \( L_I = \{ I \} \), \( L_C = \{ RG, BY \} \), \( L_O = \{ 0^\circ, 45^\circ, 90^\circ, 135^\circ \} \). These maps are then summed and normalised to form conspicuity maps:

\[
C_I = f_I, C_C = N(\sum_{l=I} f_{C,l}), C_O = N(\sum_{l=O} f_{O,l}).
\]

These are linearly combined to form the saliency maps \( S = \frac{1}{3} \sum_{k \in \{I,C,O\}} C_k \). Contrast based models can be further grouped into the following: information theoretic, Bayesian, and graphical models [67].

- **Information Theoretic:** In these models, saliency serves to maximise information sampled from our environment. To do this, the most informative regions of a scene are selected and the rest is discarded. For one such model, AIM (Attention based on Information Maximisation) [77], saliency of a local image region is defined as the information the region conveys relative to its surroundings. Information of a visual feature is calculated as \( I(f) = -\log p(f) \), to be inversely proportional to the probability of observing it. In
In this case, attention is thought to be attracted by minority features. Other approaches such as maximising entropy and measuring ‘self-resemblance’ have been explored.

- **Bayesian models:** Using Bayes Rule, prior knowledge such as context is combined with sensory information like target features. Torralba et. al [78] combine local features with a global feature that summarises the probability density of the presence of the target object in the scene. Zhang et. al [79] formulated the combination of bottom-up knowledge of features or ‘self-information’ with top-down knowledge of the target’s appearance and location. Itti et. al [80] based their model on surprising stimuli. ‘Surprise’ arises in space when an observation at one image location affects the observers beliefs derived from neighbouring locations. Temporally, it arises when observing an image at a certain point in time affects beliefs from past points.

- **Graphical models:** These models exploit the structure of graph algorithms to compute saliency. Eye movements are treated as a time series, as visual attention is a sequential process. Graph-Based Visual Saliency (GBVS) [81], for example, uses Markov chains as activation maps and incorporates a centre prior. A scale-space pyramid is derived from image features $I$, $C$ and $O$ as mentioned previously. Next, a fully connected graph over all grid locations of each feature map is built. Here nodes and edges represent states and transition probabilities respectively. Edges are directed and are assigned weights proportional to the similarity of the features and the spatial distance between the nodes they join. The weights of outbound edges are normalized to 1. The equilibrium distribution of the resulting Markov chain is adopted as the activation maps. Nodes that differ greatly from surrounding nodes are assigned higher values. Activation maps are then normalised and combined to form a saliency map. Other models have incorporated Hidden Markov Models, Dynamic Bayesian Networks and Conditional Random Fields to extend a static saliency model to a dynamic one.

**Learning Based Models:**

These models use machine learning and deep learning to learn eye fixation patterns. The top performing visual attention models for 2D-images are tested on the MIT/Tübingen Saliency Benchmark [30]. We use DeepGaze II [7] in this thesis as it was a top performing model during the time of writing of our work and had publicly released code. However, research in this field is continually progressing and improvements on this model have been made. The current state-of-the-art deep learning saliency prediction model is DeepGaze IIE [10] which improves on DeepGaze II by leveraging an ensemble of backbone encoders compared to the single VGG-19 [82] backbone employed by DeepGaze II.
2.2.3 State-of-the-Art in Visual Attention Prediction

In this section, we will give an overview of the best performing models in order of publication.

- **DeepGaze II** [7]

  ![DeepGaze II Network](image)

  As an initial step in this network, shown in Fig. 2.21, an input image is subsampled and passed through the normalised VGG-19 [82] very deep convolutional network in which all features have been rescaled to yield feature maps over the ImageNet [83] dataset. To obtain a sufficient spatial resolution for accurate prediction, these feature maps are passed through convolutional layers which have been selected by random search and are upsampled by a factor of 8. These maps are combined to form a single 3D tensor which is then inputted into a ‘readout’ network. This network can only represent a point-wise non-linearity in the VGG features so it can only learn correlations between existing features across channels not across pixels. This ensures it can’t learn new spatial features.

  The final output is convolved with a Gaussian filter to yield a saliency map $S$. As fixations tend to the centre when viewing an image, center bias is modelled and added as a prior to $S$. A softmax function is then used to convert this into a probability distribution [84].

- **EML-NET** [8] An Expandable Multi-Layer Network

  This model, shown in Fig. 2.22, uses convolutional neural networks that have been pre-trained for the task of image classification. The DenseNet-161 model [85] is pre-trained on the PLACE365 [86] dataset and the NasNet-Large model [87] is pre-trained on ImageNet [83]. The model consists of an encoding stage and decoding stage.

  In the encoding stage, the fully connected layer of the CNNs is replaced with convolutional layers to predict saliency maps. These convolutional layers map an input image to feature space. The output prediction is reduced to a single feature map by applying a $1 \times 1$
convolutional layer or conv1x1 (kernel size x no. of filter maps). The output map is then resized to that of the input by using bilinear up-sampling.

They train a decoder to combine learned features from the two CNN models. The combined features come from features extracted from multiple layers of both models, four layers from the DenseNet model and three from NasNet. Features are combined by compressing each of the selected layers into one feature map by applying conv1x1 and a rectified linear unit (ReLU). These are then up-sampled and another conv1x1 is applied to predict saliency maps [8].

- **UNISAL [9] Unified Image and Video Saliency Modeling**

This model was the first attempt, to our knowledge, to join image and video saliency estimation into a single framework. This framework is shown in Fig. 2.23. They evaluated their model on the following saliency datasets: DHF1K [88], Hollywood-2 and UCF-Sports [89] video datasets, and the MIT300 [65] and SALICON [90] image datasets. At the heart of this model is domain specific learning which is a type of domain adaptation that allows a learning system to process data from different domains by separating shared/domain-invariant parameters and private/domain-specific parameters [91].

To be able to apply domain-adaptive learning to this task, they first had to identify sources of domain shift between image and video saliency data and between various video saliency datasets. The various domain adaptation techniques used were specifically: domain-adaptive batch normalisation, domain-adaptive priors, domain-adaptive fusion, and domain-
To analyse sources of domain shift, they first randomly sampled 256 images from each dataset and ran the lightweight MobileNet V2 network (MNet V2) \[92\] on them to obtain average pooled features. Next, they applied batch normalisation, a z-normalisation on each training batch, which is used to diminish the internal covariate shift of neural network activations and at the same time, keeps a running estimate of the distribution of each batch. This allowed them to analyse the distributions of data among and between datasets. They then visualised the distributions obtained for all samples (domain-invariant) and for respective dataset samples (domain-adaptive). Upon comparing the results, a strong domain shift, evident among the different datasets upon domain-invariant normalisation, was observed to be alleviated by the domain-adaptive normalisation. Thus, the domain-adaptive normalisation became a part of their model’s architecture.

Equivalently, since the Hollywood-2 and UCF-Sports datasets had greater centre bias than DHF1K and much greater than SALICON, they decided to learn domain-adaptive priors i.e. individual Gaussian prior maps one for each dataset.

For similar image features across the training datasets, they hypothesised that the visual attention on them might vary among different datasets. For example, for the Hollywood-2 and UCF Sports datasets, viewers were asked to identify the main action shown during data collection (top-down saliency) whereas for both DHF1K and SALICON a free-viewing collection strategy (bottom-up saliency) was carried out. Therefore, they merged the output of the MNet V2 model into a single map using a conv1x1 fusion layer which was then upsampled via bilinear interpolation. They trained the fusion layer until convergence for two schemes: a domain-invariant scheme using one set of weights and a domain-adaptive scheme utilising different weights for each dataset. Upon using the domain-adaptive scheme, validation loss was found to be lower for all datasets and so this scheme was also adopted as a part of their model.
Similarly, they speculated that the size of the blur filter applied to generate the ground truth saliency maps was likely to have varied for different datasets. Upon measuring the sharpness for each dataset, this speculation was backed up by the heterogeneous distributions visible across datasets. Thus, they resolved to blur the output of their network with a different learned smoothing kernel for every dataset.

They integrated the domain-adaptive components into UNISAL, a simple (relative to other state-of-the-art models) encoder-BypassRNN-decoder network which is trained jointly on image and video saliency data. For their backbone encoder network, they selected MobileNet-V2 (MNet V2). Additionally, they proposed a BypassRNN which they define as a residual neural network (RNN) whose output is added to its input features via a residual connection that is automatically omitted (bypassed) for static batches, during training and inference. Therefore, the RNN only models the residual changes in saliency that are a result of temporal features. Furthermore, for each dataset they varied the resolution of images/frames used as input and for the temporal data and they kept only every 5th or 4th frame, depending on the dataset, to assimilate the frame rate across the two video datasets.

Retrospective experiments that they carried out confirmed the merit of the domain-adaptive components. To further analyse the contribution of the domain shift modeling, they performed an ablation study on the DHF1K and SALICON validation sets and found all of the domain-adaptive modules improves the model’s performance significantly.

With the exception of DeepGaze IIE, the UNISAL model performs as well as the other top ranked models on the image saliency datasets. Similarly, it reaches state-of-the-art performance on the video saliency datasets. In addition, compared to the smallest competitive deep learning model they achieved a 5 to 20-fold size reduction as well as a faster runtime.

• DeepGaze IIE [10]

This model mainly consists of a backbone CNN pre-trained on ImageNet classification and a readout network, illustrated in Fig. 2.24. The backbone CNN extracts features from an input image which are then fed into a readout network consisting of conv1x1 blocks, a layernorm and a softplus function. The output is blurred, weighted with centre-bias and passed though a softmax function to obtain a fixation probability density (saliency map). This readout network as well as the blur and centre bias weighting are the only parts of the model to undergo training and the backbone CNN’s weights are kept fixed throughout. Firstly, the model is pre-trained on the SALICON [90] dataset, and subsequently a 10-fold cross validation is used to evaluate every configuration of the model on the MIT1003 [65].
dataset and thus fine-tune the model.

![DeepGaze IIE: Testing Backbone Networks](image)

Figure 2.24: DeepGaze IIE: Testing Backbone Networks [10]

With the aim to achieve state-of-the-art performance, they ran numerous experimental tests and compared the results of different state-of-the-art ImageNet backbone models namely, AlexNet [83], VGG-11 and VGG-19 [82], ResNet-50 and ResNet-101 [93], ShapeNet [94], EfficientNet-B5 [95] and DenseNet [85]. As for the DeepGaze II model described previously, they use the information gain metric in their backbone testing phase. For each backbone, they implemented both a layer search and instance search experiment. From the layers extracted, they investigate which ones induced highest performance. This was followed by an instance search in which multiple initialisations of this highest performing configuration were repeated to give a reliable metric for the final performance. They observed that using around 3 to 4 layers from the final and semi-final layer spaces was ideal.

To optimise their model’s performance, they leveraged both inter-model and intra-model complementarity, visualised in Fig. 2.25. Inter-model complementarity was sought by combining multiple ImageNet backbone models into ensembles, adding one at a time until they achieve top performance. This was reached at a mixture of four models namely, ShapeNet-C, EfficientNet-B5, ResNext-50 and DenseNet-201. Adding a fifth model, ResNet-50, decreased the performance. This quadruple mixture was named DSREx3. Furthermore, they exploited intra-model complementarity by averaging several instances of the models so that each could make more informed decisions. The performance peaked at 3 instances per model.

In order to check whether a model made overconfident or underconfident predictions, they used confidence calibration when testing backbones, as in Fig. 2.25. This was achieved by sorting and then splitting predicted fixation density pixels into multiple bins according to their probability. Each bin had identical probability mass. By counting the number
of ground truth fixations in each, they determined if a model was calibrated well, since there should be the same number of fixations in each bin. An overconfident model would assign a higher probability to a region with fewer than expected fixations and other regions would have more fixations than predicted.

When evaluating confidence calibration of the backbones on the MIT1003 dataset, they found that individual models appeared to be well calibrated while ensemble models tended to be slightly underconfident. However, for the unseen datasets PASCAL-S [96] and Toronto [97], evaluation revealed the ensemble models to be better calibrated and individual models to be strongly overconfident.

Thus, the DSREx3 ensemble was chosen and renamed DeepGaze IIE. This model was found to have gained a 15% increase in performance over their previous DeepGaze II model and to rank highest in the benchmark on all metrics.

### 2.3 Saliency Prediction of Light Field Data

Light field saliency is a relatively new field of research with the first saliency prediction model for light fields presented by Li et al. in 2014 [11]. Previous to the work carried out in this thesis, light field saliency estimation has only focused on object-based methods namely, light
field salient object detection [98, 29]. These works represent saliency ground truth as binary maps obtained by manually segmenting objects that stand out in all-in-focus renderings of the light fields and human-labelling those segments as 1 and all other regions as 0. They focus on the localisation of instances of dominant objects, not taking into account tracked human gaze. In its current form, saliency estimation of LFs is essentially salient object segmentation using a single 2D-image. That is, the saliency estimation is mostly based on an all-in-focus rendered image of the LF, and the SAIs and depth map are used to incorporate additional information to improve the prediction performance. Only one type of light field rendering is considered, effectively ignoring the 4D nature of light fields. There has been only one model that uses multiple refocused renderings as a 4D input but their goal was still to output a single 2D map with the salient object segmented [99].

Though salient object detection is a continually growing field of research, as mentioned previously visual attention prediction has not been investigated to our knowledge. Saliency estimation for different novel renderings of the light field and under the visual attention sub-field of saliency was still a largely unexplored problem. Since both sub-fields are strongly related, it is our view that the methodologies and techniques used in salient object detection models can inform the field of light field visual attention prediction. Thus, in this section, we will give a brief overview of the changes and developments in light field salient object detection. Furthermore, we will detail the current state-of-the-art methods in light field saliency prediction and the datasets available.

### 2.3.1 Traditional Methods for Light Field Salient Object Detection

These methods formulate hand-crafted features to detect and segment salient objects in the form of SOD type saliency maps. The features used include colour contrast, location, background likelihood as well as light field specific features such as focusness, depth and light field flow. Most employ an algorithm like mean-shift or K-means to segment the input images into super-pixels. Many use post-refinement steps to promote object completeness and improve accuracy at the object boundary.

- **LFS [11] Light Field Saliency:**

  This pioneering model investigated the use of light field data to extract focusness and depth cues for SOD. The aim of this work was to use light field input instead of a regular RGB image and depth map to make better SOD predictions (on RGB images).

  This work formulates a region-based focusness (map) prior which gives a measure of the focusness of pixels in the focal slice input images. To determine the background slice, they first integrated the focusness measure of pixels along the x and y axes to generate
two focusness distributions. They computed a background likelihood score by scaling these distributions by a U-shaped suppression filter which is based on the assumption that salient objects are likely to lie at the image centre. The focal slice with the highest score was chosen. Multiple foreground slices were selected which have low background likelihood and high objectness score which is a measure of the completeness of a focal slice’s in-focus region. Using an all-in-focus image they calculated a colour contrast cue, a location cue using background slice focusness, and a foreground cue based on the focusness of foreground slices. To generate a saliency map, they combined these final cues using objectness as a weighting.

- **WSC** [12] Weighted Sparse Coding Framework:

  This dictionary based framework builds saliency and non-saliency dictionaries from stacked feature vectors. For 2D data, these features are color contrast and texture. When 3D (depth map) and light field (focal stack) input are used, depth information and (for 4D only) focusness priors are extracted. After applying these feature vectors to all the pixels of the image, they constructed two feature matrices, averaging and color histogram, for all superpixels. They selected a primitive saliency dictionary by first attempting to reconstruct the reference image with a non-saliency dictionary. The patches with high reconstruction error were used for the initialised saliency dictionary. The weighted sparse framework repeatedly refined the dictionary by removing the outliers and testing on the remaining superpixels until a final saliency map was output.
**DILF** Saliency Detection with a Deeper Investigation of the Light Field:

Unlike the previous models, DILF takes the depth map as input. They integrate a depth contrast map generated from the depth map and colour contrast map from the all-in-focus image. A background slice, which is constructed similarly to LFS above, is used as a weighting on the combined contrast map to eliminate background distraction. To enhance
the final result an optimization algorithm [101] is applied.

• **RL [102] Saliency Detection with Relative Location Measure:**

For a light field \(L(s, t, u, v)\), a micro-lens image \(I_{s,t}(u, v)\) is calculated for a point \((s,t)\) on a common boundary between two SAIs. First, they calculate *Edge diffusion* which is the proportion of the differences in colour between the centre view image \(I_{s,t}(u, v)\) and its neighbouring micro-lens images \(I_{s+1,t}(u, v)\) and \(I_{s-1,t}(u, v)\). The direction of the reflecting colour patterns indicates if the point is behind the plane of focus or not.

![Figure 2.29: Saliency detection using relative location estimation. From left to right: input image, selected pseudo salient regions, final saliency map [102].](image)

This is followed by the *relative location estimation* where they cluster pixels based on their colour similarity and proximity in the image plane to form superpixels. This divides the central view image into regions. The relative location \(\text{Loc}(r)\) of a region \(r\) depends on the average relative location of its boundary points.

A ranking function then assigns regions a saliency score based on relative location. Selected regions are called ‘pseudo salient regions’. These should be located in the foreground and should not adjoin with other selected regions. Similarly ‘pseudo background regions’ are regions in background selected based on relative location and should not be adjoining regions. Saliency maps are computed by an optimisation method with three terms. The first encourages foreground regions to have a higher value, the second term encourages background regions to have a lower value and the third encourages adjacent regions with similar colour to have similar saliency values.

• **BIF [13] A Two-Stage Bayesian Integration Framework for LF SOD:**

This approach integrates three saliency maps: colour contrast saliency \(S^B_C\) of all-in-focus images, depth contrast saliency \(S^B_D\) of depth map and focusness map \(S^B_F\) of foreground
slice. These are then weighted by background probability to highlight informative objects. A Bayesian framework is then used to fuse these maps in two-stages. Firstly the colour contrast saliency map is fused with the focusness map. $S^B_C$ is treated as prior when the probability $P(S^B_C|F_1)$ is computed and $S^B_B$ treated as prior when the probability $P(S^B_C|F_2)$ is computed. Bayes formula is used to compute the two corresponding posterior probabilities and integrate them into a saliency map. Similarly this saliency map is fused with the depth map $S^B_D$ by the same process.

• **MA [14] A Multi-cue Approach:**

For this model, saliency prior maps are generated from multiple light field features: colour space, location, depth, flow and structure, based on the feature distinctiveness between superpixels. They define feature distinctiveness as the pairwise distance that measures the feature difference between two superpixels/regions in the corresponding feature (cue) space. Colour and location priors are calculated from the all-in-focus image and the depth prior is inherited from the depth map. The location prior enhances superpixels in two ways: when they have spatial similarity, and when they are closer to the image centre. This paper introduces the concept of light field flow, where focusing flow is derived from the changes in focus along the focal stack, and viewing flow is extracted from the multi-view array whose images vary according to viewing angle. A random search based scheme is used to weight the various saliency cues and merge them together. Finally, to improve the overall prediction accuracy, they employ a structure cue, which takes into
account the saliency of neighbouring superpixels, to help more uniformly highlight the whole salient object. This is similar to the objectness cue for LFS.

- **SDDF [15]** Saliency Detection Based on Depth Feature:
The crux of this model is a depth feature extracted from the focal stack. To obtain this depth feature, they use a gradient operator to measure the sharpness of regions (focusness) in the focal slices and categorise the foreground slices and background slice.

To compute a coarse saliency map, the background slice is employed as a background prior to delineate foreground regions from background regions in the all-in-focus image. To improve the prediction accuracy, they further manipulated the coarse saliency map using two contrast features: colour and texture. For the texture feature, Local Binary Pattern histograms (LBP) are utilised on each superpixel to express the texture disparity between foreground and background regions. To output the final saliency map, these contrast features were linearly combined along with a background pixel suppression parameter.

- **SGDC [16] Salience Guided Depth Calibration:**
  This work derives a contrast enhanced SOD model with the intent to automatically choose a depth initialization for a compressive light field 3D display. Similar to the models discussed previously, they extract a colour cue, depth cue and focusness derived background cue from superpixels segmented from the all-in-focus image, depth map and focal slices, respectively. The final results were optimized by an optimization algorithm [101] as was used in the DILF model. This work also formulates how their SOD algorithm can be used to initialise and thus perceptually optimise a multi-layer light field display.

- **RDFD [17] Region-based Depth Feature Descriptor:**
  While other approaches rely on accurate pre-estimated depth maps, the RDFD model extracts blur and thus depth information from the focal stack images using the sparsity of its dark channel as a way to quantify the degree of blur. This is based on the observation in previous research [103] that the more blurred an image is, the less dark pixels present within it. They compute a difference image for each slice by subtracting the dark channel prior of the all-in-focus image from that of each slice. Then, for each superpixel, the
RDFD is calculated as $1 - \frac{||\Delta m(r)||_0}{T_r}$ where $\Delta m(r)$ is the difference image for a superpixel $r$ and $T_r$ is the total no. pixels belonging to $r$. This can then be used to find the depth layer of a superpixel (i.e. the slice) at which the superpixel is sharpest.

Next, a 3D spatial distribution prior is calculated (3D-SDP). To group superpixels into clusters based on their focused depth layer and 2D location cue, they employ K-means clustering. These clusters are further refined into centroids. Since foreground regions are likely to located at closer depths than far ones, the closest depth layer is chosen as the foreground. To enable foreground and background separation, they use a Gaussian kernel to create a gradient-like depth distribution for each superpixel. To render a more accurate spatial distribution prior centred at the object centre rather than the all-in-focus image centre, they use the coordinates of the foreground centroid and the coordinates of pixels in the all-in-focus image to obtain an object-biased prior. For the construction of a depth saliency map, they multiply a pair-wise distance between the RDFDs of superpixels by an exponential pairwise distance between the superpixels’ centroids.

To smoothen the depth saliency map, they apply a modified single-layer cellular automata (SCA) which allows the neighborhood of a superpixel some influence over its saliency value as defined by an impact factor. This is referred to as the optimised depth saliency map. Separately, they refine a background-based colour saliency map in two steps. Firstly, they estimate a background prior. They use the pre-optimised depth saliency map to calculate a background likelihood score which can determine background superpixels and thus, assign background regions of the all-in-focus image. Secondly, the an object-biased prior consisting of a Gaussian filter is used to further delineate between foreground and background.

To output a final saliency map they merge the optimised depth saliency map and the
background-based colour saliency map via multi-layer cellular automata (MCA) optimisation [104].

- **DCA [18] Depth-Induced Cellular Automata:** This model first computes a focusness prior as calculated in previous works. They define a background likelihood score similar to LFS, WSC, DILF and BIF however instead of using a U-shaped filter they use an inverse object-biased Gaussian filter (the inverse of the object-biased prior proposed in RDFD). This also determines the background focal slice which is used as a background prior. Then they segment the all-in-focus image into superpixels using a simple linear iterative clustering (SLIC) algorithm which is used as an alternative to the k-means algorithm often used in previous models. The focusness, background and location priors of superpixels are combined to form an object-guided depth prior.

For the selection of background seeds, they set a threshold on the object-guided depth map and superpixels with values below this become background seeds. Superpixels at the image boundary become background seeds if they are below another lower threshold. This threshold is relatively lower than the first to ensure only non-salient superpixels are selected. By enhancing the background seeds with the colour and location information of their respective superpixels, they create a contrast saliency map. To form a depth-induced saliency map, the contrast saliency map is multiplied by the object-guided depth map. To ensure spatial coherence, they propose a Depth-induced Cellular Automata (DCA) optimisation model. This involves three modifications to the SCA of the RDFD SOD model above: instead of joining all boundary superpixels, they only join those set as background seeds, they add an additional threshold on the superpixels that count as neighbours and
they modify the impact factor (i.e. the influence of neighbours) to account for the colour distance of superpixels as well as depth distance.

Similar to BIF, they employ a Bayesian fusion strategy to merge the object guided depth map with the optimised DCA output map. To further refine and improve spatial consistency, they employ a a pixel-wise saliency refinement model called CRF [105] on the fused map and thus calculate the final saliency map.

2.3.2 Deep Learning Methods for Light Field Saliency Prediction

The state-of-the-art LF saliency prediction models all employ deep-learning methods for salient object detection. In their light field saliency review and benchmark paper, Fu et al. [29] evaluated the state-of-the-art models and found that ERNet [106] achieved top performance across the datasets and metrics with MoLF [107] and LFNet [108] following in second and third place. Since their review in 2020, two new deep learning models have been developed with novel architectures DLG [109] which utilises graph neural networks and MGAN [110] which employs a multi-generator adversarial network.

The VGG-19 [82] network was adopted as a backbone architecture for all the networks except MTC-Net [111], MAC [1] and MGAN [110]. To adapt the VGG-19 network to better suit the task of SOD, the models drop the last pooling layer and the fully-connected layers, thus preserving five convolutional blocks which they then modify for use as an encoder. Furthermore, Fu et al. grouped the light field saliency deep learning methods, available to them at the time, into five categories. We will use the same categorisation scheme here for consistency and with the addition of the most recently published models for completeness.

- **Late Fusion:** These involve a two-stream fusion approach. They aim to obtain saliency maps or features using separate deep learning networks for the first input, light field captures as either focal stack or multi-view images and the second input, an all-in-focus or centre-view image and then combine them into a final output saliency map.

  The DLLF model [112] uses two CNN streams: A focal stack stream uses a recurrent attention network to gradually integrate weighted features extracted from each focal slice and refine their spatial information. These combined features are fed to convolutional layers to obtain a saliency map. This is fused with a saliency map generated from a second all-in-focus stream to generate a final saliency map. To make the network more robust against noise, they generate adversarial data by adding noise to training images.

  On the other hand, the MTCNet model [111] consists of a two module *multi-task collaborative network*. Firstly, a saliency-aware feature aggregation module (SAFA) extracts low-level features, focal plane edge information and heuristic semantic priors from the
centre-view SAI. Secondly, to obtain depth-oriented saliency features, a multi-view inspired depth saliency feature extraction (MVI-DSF) module is employed on multi-view images. To combine features and compute the final saliency map, a feature-enhanced salient object generator is used.

- **Middle Fusion:** This type of model also extracts features from the focal stack and all-in-focus images in two separate streams through the encoder. However, in contrast to Late Fusion, more hierarchical and intermediate features are fused through the decoder to ultimately output a saliency map.

For MoLF [107], two memory oriented modules, a spatial fusion module (Mo-SFM) and feature integration module (Mo-FIM) are used in the decoder. The Mo-SFM module weights light field features as a sequence of inputs corresponding to consecutive time steps via an attention mechanism and adaptively refines spatial relationships between these features using a ConvLSTM [113]. To capture global contextual information of the fused feature maps, a global perception module (GPM) is used on top of Mo-SFM. Mo-FIM consists of a scene context integration module (SCIM) and a ConvLSTM. To update the feature map, the SCIM uses memory information from upper layers to learn a
channel attention map. Thus, more relevant channels of the feature map can be emphasised and unimportant ones suppressed. A ConvLSTM progressively fuses each updated feature map with the last fused input and influences low-level feature selection. To compute a final saliency map a transition convolutional layer and an up-sample operation are performed.

Similarly, LFNet \[108\] was designed with two modules in the decoder, a light field refinement module (LFRM) and integration module (LFIM). LFIM learns complementary information between the features of both streams as follows: the feature maps from the focal stack stream are given as input to a refinement unit to learn the residuals. These residuals are fused with the all-in-focus features to further refine the saliency map. The LFIM employs an attention block which adaptively weights and integrates focal slice features (such as focusness) with the saliency map and a ConvLSTM which exploits the implicit spatial structural information of focal slices to emphasise more relevant light field features.

The most recent model DLG \[109\] performs the fusion of focal stack features using graph networks. Graph networks allow for strong context propagation from neighbouring nodes. These contexts are combined in focal stack images under the guidance of the all-focus image. The reciprocative guidance scheme consists of iterative guidance between the two feature types. Thus, they promote one another at multiple steps.

- **Knowledge-distillation:** With the aim to improve predictions and also keep computational complexity and memory usage low, a two stream teacher-student network was proposed. To boost the performance of the all-in-focus stream, the features and predictions of the focal stack stream are used to supervise it.

The ERNet model \[106\] is constructed using this architecture. The teacher network operates on the focal slices and involves two modules: a multi-focusness recruiting module (MFRM) which recruits rich saliency features and a multi-focusness screening module (MFSM) to screen for useful features and emphasise important ones.

To take account of the intrinsic differences between the all-in-focus image and the focal slices, a knowledge distillation process is used to supervise the training of the student network instead of transferring the knowledge extracted from the teacher to the student network directly. Throughout this distillation process, the teacher is pre-trained and the parameters are not changed. The process involves a multi-focusness distillation which promotes multi-focusness consistency between the two streams (focal stack and all-in-focus) by reducing the KLD loss between the penultimate layer of the all-in-focus stream and the MFRM recruited features. Then, a screened-focusness distillation allows the
student network to learn information which is complementary to both the screened focus-
ness knowledge output of the MFSM and the appearance information of the all-in-focus
stream. This is achieved by backpropagating a linear combination of KLD and cross
entropy losses throughout the full network.

A great benefit of this model is that, for cases where there are limited resources, the
student network alone can be used to generate a saliency map from just an RGB image
input but still benefit from the pre-trained teacher.

• **Reconstruction-based:** These reformulate SOD into two sub-tasks: reconstruction of
light field data synthesised from a single image input, light field driven SOD to complete
the saliency prediction.

The only model so far with this architecture is DLSD [114]. The reconstruction module
predicts depth maps along the horizontal and vertical directions only, to reduce redun-
dancy, with two independent depth-trained CNNs (one for each direction). The centre-
view image and the estimated depth maps are fed into a warping layer to render other
viewpoints of the light field using a Lambertian approximation. The reconstruction error
between the predicted images and the ground truth is computed and the $L_1$ loss function
supervises the reconstruction quality. To further improve the quality, a regularization loss
is applied.

They employ a multi-view saliency detection subnetwork based on the VGG-19 model [82]
to extract saliency rich features. They first combine the features which share the same
convolutional block, and then use a small convolutional layer to weight the relevancy of
the features. They recursively refine the saliency from the highest convolutional layer to
lower layers.

A multi-view attention module exploits the geometric/object shifts between multi-view
saliency maps to estimate saliency. Using the geometric information of multi-view depth
maps, they warp the multi-view saliency maps so they align with that of the centre view
and then concatenate them with the depth maps in the colour channel. The resulting 4D
vector of views and their features is used as input to more deep learning layers to predict
a weight for each views’ saliency map. These weights are then spatially normalised. A
final saliency map is calculated as the weighted sum of the saliency maps of views.

• **Single Stream:** This model type consists of a single bottom-up stream which takes a
micro-lens image array as input.

MAC [1] is an end-to-end CNN based network which takes a micro-lens image as input.
The network consists of *Model Angular Changes* (MAC) blocks to extract features
from micro-lens images and a modified DeepLab-v2 backbone model which takes multiscale information and long range spatial dependencies into account. They propose three variants of the MAC blocks: MAC-9x9 with one layer of convolution and a stride of one where the kernel size is equal to the angular resolution of the micro-lens image, MAC-3x3 with two layers of 3x3 convolution and a stride of three and MAC-star shaped which has atrous angular convolutional kernels with five atrous rates set to sample four viewpoint directions thus forming a star shape. They found that the MAC-9x9 block architecture performs best and this is likely due to difficulty in training the greater number of parameters of the other architectures.

- **GAN-based**: These models make use of a generative adversarial network (GAN) which consists of two contesting neural networks, a generator which synthesizes fake data/images and a discriminator which distinguishes these fake data from real (ground truth) data.

The MGAN [110], which stands for multi-generator adversarial network, is the first GAN-based light field SOD. It consists of four cascaded generators and a discriminator. Each generator is an encoder-decoder network inspired by U-net [115]. For the decoders, they implement transposed convolution to increase the feature maps’ resolution which has been found to increase GAN stability. To increase the speed of network convergence, batch normalisations are used after each convolution. Differing from U-net they apply a Leaky (ReLU) to improve the networks non-linearity. A sigmoid activation function in the generators’ last layer outputs the saliency predictions.

The first generator $G_0$ extracts saliency features from an input all-in-focus image. To extract and fuse focal stack features, the successive generators $\{G_4, G_7, G_{11}\}$, take the output of the previous generator and focal stacks of differing sizes (indicated by the subscript) as input since varying the no. of focal slices avoids redundancy. In an effort to preserve colour and texture information and reduce computational cost, the final ‘fake’ output saliency map and corresponding ‘real ground truth map are separately multiplied by the all-in-focus image before being used as input to the discriminator.

For their discriminator, they implement PatchGan [116] which ultimately tries to classify each patch of an image as real or fake. They found that combining adversarial loss with two content losses is advantageous for learning the structure and edge information of salient objects during training.

After extensive analysis, their model is shown to offer state-of-the-art performance on the newest dataset Lytro-Illum and perform competitively against rival models on the LFSD and HFUT datasets. They concluded that, due to the relatively smaller focal stacks
in these two datasets, MGAN may have ignored depth information which diminished performance. However this network is novel and gives a competitive method for LF SOD on data with larger focal stacks.

### 2.3.3 Datasets Available

At present, to our knowledge, there are five datasets for light field salient object detection: LFSD (100 light fields) [11] and HFUT-Lytro (255 light fields) [117], DUT-LF (1462 light fields) [112], DUT-MV (1580 light fields) [114] and Lytro-Illum (640 light fields) [1].

Table 2.1: Summary of the light field SOD datasets. The dataset name, its scale, the device used to collect the data, the angular and spatial resolution, the size of the focal stack where applicable, the data types included, the multiple-object proportion (MOP) of the dataset and the corresponding LF model included with each dataset paper.

<table>
<thead>
<tr>
<th>Name</th>
<th>Scale</th>
<th>Device</th>
<th>Angular Res</th>
<th>Spatial Res</th>
<th>Focal Stack Size</th>
<th>Data Types Provided</th>
<th>MOP</th>
<th>LF Model</th>
</tr>
</thead>
<tbody>
<tr>
<td>LFSD [11]</td>
<td>100</td>
<td>Lytro</td>
<td>360 × 360</td>
<td>3-12</td>
<td>GT, focal stack, depth map, raw LF data</td>
<td>0.04</td>
<td>LFS</td>
<td></td>
</tr>
<tr>
<td>HFUT [117]</td>
<td>255</td>
<td>Lytro</td>
<td>328 × 328</td>
<td>1-12</td>
<td>GT, focal stack, depth map, SAIs</td>
<td>0.29</td>
<td>MA</td>
<td></td>
</tr>
<tr>
<td>DUT-LF [112]</td>
<td>1462 (1000 train, 462 test)</td>
<td>Lytro Illum</td>
<td>600 × 400</td>
<td>2-12</td>
<td>GT, focal stack, depth map</td>
<td>0.05</td>
<td>DLLF</td>
<td></td>
</tr>
<tr>
<td>DUT-MV [114]</td>
<td>1580 (1100 train, 480 test)</td>
<td>Lytro Illum</td>
<td>590 × 400</td>
<td>-</td>
<td>GT, SAIs</td>
<td>0.04</td>
<td>DLSD</td>
<td></td>
</tr>
<tr>
<td>Lytro-Illum [1]</td>
<td>640</td>
<td>Lytro Illum</td>
<td>540 × 375</td>
<td>-</td>
<td>GT, micro-lens images, raw LF data</td>
<td>0.15</td>
<td>MAC</td>
<td></td>
</tr>
</tbody>
</table>

Although LFSD has some variety in so far as it consists of 60 indoor and 40 outdoor scenes, the scenes mainly contain a single object centred in the frame with a simple background. HFUT, which also contains a mixture of indoor and outdoor scenes, tried to make the scenes more realistic by capturing some scenes with occlusions and background clutter, thus making the data more challenging for SOD models.

DUT-LF strived to further improve on these datasets by incorporating different object types (e.g. transparent objects), including scenes with low contrast between salient objects and the background, and varying object locations in the scene. The data used to generate the DUT-MV dataset is of the same origin as DUT-LF, therefore many of the light field scenes are common to both (1081 common scenes). However, unlike the other datasets which extracted light field information from focal stacks and depth maps, DUT-MV provides instead the multi-view array for each light field along with the ground truth. Thus, this dataset allows for the development of models which can take advantage of angular cues. Finally, the Lytro-Illum dataset, which contributes raw light field data and micro-lens images, provides scenes with a variation in object size, texture, background clutter and lighting.

Fu et al. performed a statistical analysis of these datasets in their light field benchmark and review paper [29]. They calculated the size ratio of objects in the light field data for each dataset, and found that most objects have size ratios smaller than 0.6. For HFUT and Lytro Illum the objects were relatively small, whereas for LFSD the objects were generally larger. They also found that the light fields of all five datasets had strong centre bias with regards to
object placement. By calculating the multiple-object proportion (MOP) the proportion of light fields in the entire dataset with more than one object, values shown here in Table 2.1, they demonstrate that there were very few cases of multiple objects.

From the datasets that included focal stacks, namely LFSD, HFUT and DUT-LF, they worked out the distribution of focal stack size of light fields for each of them. LFSD was found to have the greatest depth information (as a percentage of the total data) with a peak distribution of 12 focal slices compared to HFUT with a peak distribution of three consisting of mostly smaller focal stacks and DUT-LF which peaked at 6 focal slices.

### 2.4 Evaluation Metrics

In this section, we will go through the five common evaluation metrics for visual attention that are used in this thesis: area under curve (AUC), normalized scanpath saliency (NSS), Pearson’s correlation coefficient (CC), Kullback-Leibler divergence (KLD), and Similarity (SIM). These were selected for the evaluation of visual attention models in our work as they are the most commonly reported metrics in saliency evaluation [118]. To compute these metrics we used open source code [119]. These metrics measure our saliency estimators’ performance varying in approach and criteria. AUC and NSS are location-based similarity metrics, CC and SIM are distribution-based similarity metrics, and KLD is a distribution-based dissimilarity metric [118].

- **Area Under the Curve (AUC):** The area under the receiver operating curve (ROC) is a location-based similarity metric which indicates a model is returning accurate results if its value is close to 1. Where the ROC is a plot of the trade-off between sensitivity and 1 – specificity for different thresholds. Sensitivity is the true positive rate (TPR) \( \frac{T_p}{T_p+F_n} \) and specificity is the true negative rate (TNR) \( \frac{T_n}{T_p+F_n} \) where \( T_p \) is the no. of true positives, \( F_p \) is the no. of false positives, \( T_n \) is the no. of true negatives and \( F_n \) is the no. of false negatives. 1 - specificity is equal to the false positive rate (FPR).

This metric measures saliency map performance as a classifier of pixels which contain fixations or do not as follows: for saliency maps with pixel values in the range \([0,255]\), we can compute binary masks by varying a threshold value \( T \) between 0 and 255 and assign all pixels in the saliency map below \( T \) a value of 0 and the rest 255. The saliency map is thus treated as a binary classifier to separate positive samples from negative ones at each threshold. The TPR is calculated as the ratio of true positives \( T_p \) to the total number of fixations, where \( T_p \) is the proportion of saliency map values above the threshold at fixation locations. The FPR is computed as the ratio of \( F_p \) to the total number of saliency map pixels, where \( F_p \) is the proportion saliency map values above the threshold at unfixedated

[https://github.com/dariozanca/FixaTons](https://github.com/dariozanca/FixaTons)
pixels. This version of AUC which we use in this thesis is also called AUC-Judd [65]. Saliency maps that place different quantities of density but on the correct/fixated pixels will have similar AUC scores. Low-valued false positives do not have much of an impact on the AUC score since they are not heavily penalised as for other scores such as KLD [118].

- **Normalized Scanpath Saliency (NSS):** NSS is a similarity metric used to measure the average normalized saliency between the predicted saliency map $S$ and the ground truth fixation map $F$ which is a binary map of the ground truth fixation locations.

$$NSS(S, F) = \frac{1}{\sum_{i} F_i} \sum_{i} \bar{S} \times F_i$$

where $\bar{S} = \frac{S - \mu_S}{\sigma_S}$ where $i$ is a pixel index and the $F_i$ are the fixated pixels. $NSS <= 0$ indicates the model performs no better than choosing points at random. $NSS = 1$ corresponds to fixations falling in a region of the saliency map with a saliency value one standard deviation above average. The higher the NSS score the better the estimator predicts visual attention at fixation points [120]. According to Bylinskii et al. [118] NSS should behave very similar to CC and like CC is affected by false positives and false negatives equally.

- **Pearson’s Correlation Coefficient (CC):** This metric measures the linear correlation between two probability distributions. Given the model’s estimated saliency map $S$ and the ground truth saliency map $GT$, CC is calculated as

$$CC(GT, S) = \frac{\sum(GT_i - \mu_{GT})(S_i - \mu_S)}{\sqrt{\sigma^2_{GT} \sigma^2_S}}$$

where $\mu$ and $\sigma^2$ are the mean and variance. The closer the value is to +1/-1 the greater the linear relationship between maps.

- **Similarity (SIM):** SIM outputs the similarity between two distributions viewed as histograms. A SIM of 1 indicates the distributions are the same and 0 suggests that there is no overlap. After normalizing the input maps namely the predicted saliency map $S$ and the ground truth saliency map $GT$, SIM is computed as the sum of the minimum values at each pixel as follows:

$$SIM(GT, S) = \sum_i \min(GT_i, S_i)$$

where $\sum_i GT_i = \sum_i S_i = 1$. SIM is very sensitive to missing values, and penalises false negatives much more harshly than false positives whereas CC penalises both equally [118].

- **Kullback-Leibler Divergence (KLD):** This is a dissimilarity metric which measures the distance between two probability distributions. In the case of visual attention, it measures how far apart the model’s output saliency map $S$ is from the underlying visual attention distribution ie. the ground truth saliency map $GT$ [67]. Better models have lower KL divergence, viewers gaze at a a small number of regions of high model responses and avoid the majority of regions with low model responses.

$$KLD(GT, S) = \sum_i GT_i \log \epsilon + \frac{GT_i}{S_i + \epsilon}$$

where $i$ denotes the location of pixels in a saliency map and $\epsilon$ is a regularisation term. Misdetections are highly penalised [118].
Chapter 3

Light Field Visual Attention Dataset

In this chapter, we present a visual attention study on light field content that we designed and implemented. The study involved conducting perception experiments by displaying light field content to users in various ways and collecting corresponding visual attention data. Our analysis highlights characteristics of user behaviour in light field imaging applications and offers insights into what factors light field saliency models should take into account.

3.1 Introduction

Light fields hold more information than a regular image and can be used in various applications [46] including refocusing [56] and streaming [121]. It’s plausible that visual attention (where people look when they view a scene) varies according to media type. As a type distinct from 2D-image captures of scenes, light fields’ relationship with visual attention may differ from conventional images.

To our knowledge, saliency of varied renderings of light fields has not been previously investigated. Hence, we created a light field visual attention dataset by amalgamating light fields from different sources and collecting eye tracking data for varying rendering scenarios. Our goal was to obtain ground truth visual attention data for light fields and to analyse how visual attention manifests for different light field renderings using this data.

Light field refocusing was our chosen method to render the light fields which was representative of their 4D nature, but for a 2D display. We subsequently examined how changes in focus affected participants’ visual attention, treating focus as a cue characteristic of light fields. The work presented in this chapter has been published in [31].

49
3.2 Related Work

3.2.1 Eye Movements

When viewing a visual scene, an observer’s eye movements are self-perceived to be smooth and constant whereas in actuality they consist of small discrete steps and involuntary spasm-like micro-movements. These are known as fixational eye movements and help to keep regions of interest in the centre of our field of vision [75]. When observing a stationary scene, the fixation of our attention on an element of the scene is accompanied by the fixation of our eye’s gaze.

There are three main types of eye movements that occur during fixation: tremors, drifts and saccades [75], [122]. Tremors are the smallest eye movement. They are oscillatory motions of the axes of the eyes at high frequency but very small amplitude. Drifts are slow movements of the eye with an erratic trajectory that occur between saccades and simultaneously with tremors. During drifts the image of the region being fixated can move across a dozen photoreceptors. They play a compensatory role in the maintaining of accurate fixation in the absence of saccades. Saccades are small rapid jerky movements whose main purpose is to change the point of fixation by directing the fovea towards a particular element of the object of perception. Saccades may also occur when the duration of fixation exceeds a certain length of time (0.3 - 0.5 sec), to re-centre the point of fixation on the fovea when it has moved too far due to drifts and to control binocular disparity among other functions.

In our human visual system retinal adaptation has arisen, the effect of which is that the image of an unchanging stationary object would fade to an empty homogeneous field if our eye were to remain still [122]. This has been shown through experimental means in retinal stabilisation studies which shift a visual stimulus such that all eye movements are effectively cancelled out. Similarly, during fixation, stationary objects in the periphery tend to fade and disappear. This phenomenon is known as Troxler’s effect [123]. Fixational eye movements have the added function of counteracting retinal adaptation and the Troxler’s effect by generating small random displacements of the retinal image.

In 2003, Engbert and Kliegel [124] developed an algorithm for detecting saccades using the directional distribution of velocities calculated from a time series of eye positions. Using this algorithm they showed that covert shifts of attention had an effect on the orientation of saccades. From this, they proposed that saccades could be used as a means to study the dynamics of allocation of visual attention. According to the book “Eye Movements and Vision” [75], the fixational process is the process of perception that takes place between any two adjacent saccades, and in the intervals between head rotations and blinks the human eyes can be in two states: the state of fixation and the state of changing the point of fixation. Using the algorithm from [124] these two states “fixation” and “saccade” may be computed from eye positions de-
ected by an eye-tracker and are nowadays used to measure visual attention. We discuss eye trackers below.

### 3.2.2 Eyelink 1000 Plus

The Eyelink 1000 Plus [19] is a video-based eye-tracker with a camera sensor and infrared light source to its side. It tracks eye movements by capturing a series of images, up to 1000 per second, of one or both eyes. It consists of a presentation computer which displays the desired stimulus and a host computer which identifies and relays where a user is looking back to the presentation computer. This is done within 3ms of an image being captured. The eye gaze data tracked by the Eyelink has an accuracy down to 0.15º visual angle and 0.25º - 0.50º under hands-free and typical viewing conditions.

To compute eye gaze coordinates, the Eyelink Core system uses image processing algorithms to work out the centre of the pupil and the corneal reflection from the images captured. The corneal reflection is the infrared light coming from the device reflecting off the back of the eye.

![Figure 3.1: The Eyelink 1000 Plus experimental setup with the desktop mount and chin rest along with an example of an image captured by the camera sensor [19].](image)

The Eyelink can capture eye gaze coordinates from these two points of reference as follows. Keeping a user’s head fixed, the corneal reflection remains fixed. When a user rotates their eye the location of the centre of the pupil shifts on the camera sensor but it does so relative to the corneal reflection. When a user moves their head the corneal reflection and centre of the pupil shift together. Thus, using both the pupil centre and corneal reflection the eye-tracker can accurately estimate eye gaze location of a user.

The EyeLink Core System records (x,y) coordinates of eye events and labels them appropriately as saccades or fixations, it also records other relevant data such as pupil size and blinks. This data is stored in an EyeLink data file called an EDF. We used the desktop mount with head-fixed eye-tracking mode for our experiment however there are multiple mount options and
a hands-free eye-tracking mode. To design an experimental test-bed which interacts with the Eyelink, a third party software such as Psychtoolbox 3 [125] and PsychoPy [126] can be used. Using this software, a stimulus presentation can be shown and the Eyelink recording can be controlled simultaneously. Furthermore, the recorded eye gaze data can be fed back in as input to the stimulus presentation.

3.3 Light Field Eye Gaze Data Collection

In this section, we will go into detail about our selection of light field datasets and the rendering scenarios we used for our experiment. We will also describe the experiment itself, including the setup, the apparatus and software used, and the methodology.

3.3.1 Choice of Light Field Data and Rendering Methods

The data was selected from four main light field datasets: Stanford (New) Light Field Archive [40], EPFL Light Field Image Dataset [127], Disney High Spatio-Angular Resolution Light Fields [44] and HCI Heidelberg 4D Light Field Dataset [54]. We believe that the selection of these four datasets makes the collected data representative as the light fields were acquired using a camera array [40], a single camera with microlens array [127], a camera on a gantry [44], and computer generated imagery [54] respectively. The light fields of the datasets also varied in terms of angular resolution, spatial resolution and depth captured.

We selected 20 light fields from these datasets according to the following criteria: they contained multiple objects, objects with high colour contrast between each other and the back-
ground, and contained regions with great edge density and local contrast at varied depths and spatial locations. This was again done for the sake of diversity and could be expanded in future work.

We generated focal stacks (120 slices) for each of our light fields using the Fourier Disparity Layers method [56] and used them to simulate traversing a 3D scene on a 2D display. We considered three different scenarios for light field rendering and rendered each light field in five ways as follows:

1. **all-in-focus**: all the points in the rendered image are in focus. This firstly offers a comparison for the focally varying renderings below. This rendering method is also used in the majority of existing literature so can allow us to compare with previous work.

2. **region-in-focus**: one slice/image of the focal stack is rendered so only objects at that slice’s specific depth of focus appear sharp. We rendered two regions *region-1* and *region-2* which have objects in opposite positions of the frame e.g. left/right, top/bottom, foreground/background. This was chosen with the intention of allowing us to examine the effect of the focus cue in static images.

3. **focal-sweep**: all the images of the focal stack are rendered in sequence. We rendered two focal sweeps *front-to-back* with region of focus moving from foreground to background and *back-to-front* with region of focus moving from background to foreground. This was chosen with the intention of allowing us to examine the effect of the focus cue in dynamic image sequences.

We use the following labelling scheme throughout this thesis to denote the five rendering cases: all-in-focus (AiF), region-1 (RiF1), region-2 (RiF2), back-to-front focal-sweep (B2F), and front-to-back focal-sweep (F2B).

### 3.3.2 Experimental Setup

We used the state-of-the-art Eyelink 1000 plus [19] eye-tracker with desktop mount which records eye movements with a sampling rate of 1000Hz. The visual stimuli were presented on a 23.8 inch Dell P2415Q monitor (height × width: 29.6 × 52.7 cm; native resolution: 4K/UHD/2160p; refresh rate: 60Hz). The monitor was placed at 67cm from the users eye which kept the visual angle of the stimuli between 39° and 24°. The resolution of the monitor was set to be 1920 × 1080 pixels (16:9 aspect ratio).

The experiment was held in a quiet, well lit room with white walls. We used the standard Eyelink 1000 chin rest to minimise head movement. We specified the width and height of the monitor as well as the resolution and our measured eye to screen distance in the eye-tracker.
configuration files. We wrote a script in Matlab (R2019b) using the EyeLink Toolbox within Psychtoolbox 3 [125] which displays the renderings to participants and records their fixation data. This can be found at the github link provided [1], which can be used to reproduce our experiment.

3.3.3 Participants & Methodology

We conducted the experiment, following ethics approval, using 21 participants (16 male and 5 female), aged between 18 and 37 with a mean age of 25.3. A department wide email was sent to students and staff for recruitment. All participants had normal vision or corrected-to-normal vision (approximately 6/5 to 6/4 on the Snellen Chart). The experiment lasted between 25 and 35 minutes for each participant. A brief oral overview of the experiment as well as an information sheet and consent form were provided to participants. They were instructed to view the stimuli freely and naturally while keeping their heads as still as possible. The chin rest position was fixed but the participants could raise or lower their chair until they were comfortable. The distance from the eye to eye-tracker was maintained at 53cm.

Eye movements of the left eye only were recorded. We used the Eyelink default monocular

[125] https://github.com/ailbhegill

Figure 3.3: Light field rendering cases visualisation for the Tower light field [54]
nine-point calibration and validation procedure[7]. We showed the participants the light fields rendered all-in-focus for 4 seconds each to acquaint them with the data. They were then shown the five renderings of each light field, for 10 seconds each (120 frames with 12 fps), in randomised order, with a 2 second interval between. The interval screen was to ensure fixation was re-centred. Randomisation was used to avoid carryover [128]. At the cessation of the experiment, we uploaded the eye fixation data into a database for subsequent processing and analysis. This dataset contained the following fields: Participant, Time at start of fixation, Time at end of fixation, Duration, X-coordinate, Y-coordinate, Key, Light field name, Rendering type, Time at start of rendering, Time at end of rendering.

3.4 Light Field Eye Gaze Data Processing

We began by processing the data in order to be able to analyse the influence of focus cue and any other cues which may be driving visual attention. This primarily involved creating saliency maps, fixation maps and scanpaths, the details of which are explained in Section 2.2.1.

3.4.1 Saliency Map Generation

For each of the five renderings shown to participants, we chose to create average saliency maps, in order to examine the influence of the focus cue and its effect in relation to other cues. For the focal-sweep renderings, we generated temporal saliency maps which allowed us to investigate how focus cue affects saliency temporally. In order to first visualise the data, we followed previous approaches and generated heat maps to provide an intuitive sense of saliency and also to act as ground-truth data in the work outlined Chapters 4 and 5.

To obtain these saliency maps, we applied a Gaussian filter to our fixation data with $\sigma$ equal to $1^\circ$ of visual angle along the x and y axes. As our largest image width was 47.11 cm and height was 29.60 cm, we calculated the visual angle to be 24.91° to 38.74° respectively. We then found that $1^\circ$ visual angle corresponds to 47.66 pixels horizontally and 42.67 pixels vertically and used these values as our standard deviations $\sigma_x$ and $\sigma_y$ [129]. We used the duration of the fixations as a multiplicative weight when computing the Gaussian filter. This resulted in a density map, examples of which are illustrated in Fig. 3.5. These heat maps can be interpreted as the probability density of fixations. These saliency maps hold two-dimensional data but are visualised in this chapter as RGB colour images using a colour map. All data from all participants was used to create the average saliency maps for a given rendering. For temporal maps, the videos were grouped into five chunks of 2 seconds each and the fixation data for each chunk for all participants was used to generate the respective saliency maps. This allowed

us to observe changes in gaze over time which was especially important for the videos which changed plane of focus, from foreground to background and vice versa.

### 3.4.2 Fixation Map Generation

We also created fixation maps based on the fixation data gathered for each rendering and light field. We generated them by creating a binary fixation map, which had ones at the pixel coordinates where any participant looked at any point in time and zeros elsewhere.

\[
F_{C}(i, j) = \begin{cases} 
1 & \text{if there is a fixation at } I(i, j) \\
0 & \text{otherwise}
\end{cases} \quad (3.1)
\]

This is succinctly described in equation 3.1, where \( F \) is the fixation map, \( i \) and \( j \) are the row and column pixels, \( I \) is the image stimulus, and \( C \in \{AiF, RiF1, RiF2, B2F, F2B\} \) denotes the rendering. This yields a fixation map which is sparse. These fixation maps were later used to compute evaluation metrics for the models created in Chapters 4 and 5.

### 3.4.3 Scanpath Generation

Complementary to this, we also created scanpaths, which illustrate how a viewer’s gaze varies spatially over time. These were generated by drawing a line between consecutive fixations, for each participant. We then overlaid these onto the associated rendering. We used a linear colour palette from yellow to blue to illustrate the time component, with yellow indicating more recent fixations and blue indicating less recent fixations. This is illustrated in Fig. 3.4. These scanpaths were used for analysis only and were not used as input to the models created in Chapters 4 and 5.

### 3.5 Light Field Eye Gaze Data Analysis

We then proceeded to analyse the data both qualitatively, by visually comparing and observing the visual results we obtained via scanpaths and saliency maps, and quantitatively, by evaluating the entropy of fixation maps and investigating the influence of the focus cue using the \( R^2 \) metric.

#### 3.5.1 Qualitative Analysis of Scanpaths and Saliency Maps

Fig. 3.4 shows the scanpaths of the raw eye tracking data for each rendering of two sample light fields. Clusters where the colours of the scanpaths are the same reveal a common path of fixation for multiple participants. This can be observed in the scanpaths of the videos where
the focal plane varies over time, as shown in the front-to-back and back-to-front renderings in Fig. 3.4. This is in contrast to the scanpaths of the data collected with non-varying focal planes i.e, all-in-focus, region-1 and region-2, which suggest that each participant views regions of the scene in a different order. This phenomenon is also seen in images rendered from the other light fields in this data set.

![Images of scanpaths](image)

Figure 3.4: Scanpaths of the Treasure light field’s renderings. Each continuous chain represents a participant, and colour mapping shows the passage of time which starts with blue. Yellow indicates the most recent time instant.

We created saliency maps for each light field and corresponding rendering in two ways. The first method involved computing maps using the fixations of all participants for the full 10 second video. The second split each video over time into 5 segments (of 2 seconds each). For each of these, we generated a saliency map per segment using the fixations of all participants. This allowed us to see changes in visual attention over time. The method for this is described in Section 3.4 above.

![Images of saliency maps](image)

Figure 3.5: The Medieval light field renderings overlayed with their corresponding colour saliency maps. These saliency maps were generated from the ground truth eye-tracking data of all participants averaged over the full 10s recorded.

We studied the saliency maps averaged over the entire 10-second video. We found that
some had very similar saliency maps for the all-in-focus and focal-sweeps. However, there were also many saliency maps where focal-sweeps revealed other salient regions. For example, in the Medieval light field in Fig. 3.5, the centre building is salient in the back-to-front rendering whereas it is not in the all-in-focus rendering. This shows that a static rendering of a scene may not reveal all the salient regions present in the four-dimensional light field data.

Moreover, to understand what causes participants to fixate on regions of focus and whether or not they always do, we compared the segments over time of static data (all-in-focus and region-in-focus renderings) to those of focally-varying data (focal-sweeps) for each light field. We found that gaze is held on objects that are in-focus when they are also salient in the all-in-focus rendering. For example, the basketball and centre shoe in the Sideboard light field shown in Fig. 3.6 (a) and (b).

There are other cases where there are objects in a scene that have a higher level of saliency in the all-in-focus rendering and they pull the viewers attention away from the region of focus.
in other renderings. For example, observe the region-in-focus rendering of the Tarot-Small light field in Fig. 3.6 (c). Although the focus is in the foreground, the saliency map is also concentrated on the centre ball. As the centre ball has a high saliency in the all-in-focus rendering Fig. 3.6 (d), we can deduce that it is salient independent of its level of focus.

Some scenes do not depict a specific object/ exhibit a region of high saliency. These tend to produce highly dispersed saliency maps. This is demonstrated in the Treasure light field in Fig. 3.6 (e). The saliency dispersion in the all-in-focus rendering suggests that the viewer is likely to follow the region of focus almost exclusively in other renderings as seen in Fig. 3.6 (f).

This trend of following gaze is also evident when there are a few objects with similar levels of saliency in the all-in-focus rendering. This is seen in the animal heads and the central object in the Couch light field Fig. 3.6 (g). In the focal-sweep Fig. 3.6 (h), we can see the viewers gaze following the path of focus as above but not as smoothly, rather jumping between the salient objects that are in-focus in each segment.

3.5.2 Quantitative Analysis of the Distribution of Fixations

In this subsection, we provide a quantitative analysis to understand the relationship between human visual perception and light field rendering.

We infer from our qualitative analysis that viewers are likely to fixate on and follow regions in-focus and that fixations of all-in-focus renderings were more dispersed. To further examine these observations, we used the calculated entropy of fixations recorded by the eye tracker to determine if participants’ fixations varied more or less for each rendering per light field.

To calculate entropy, we first created a fixation map with the same spatial resolution as the stimuli (i.e., 1920 × 1080), which was populated with the fixations from all the users for a specific case, using the processed described in Subsection 3.4.2. The entropy values were generated for each of the five cases and reported in Table 3.1 using Matlab’s entropy($F_C$) function which computes the probability of 1s occurring in $F_C$ and is calculated using the entropy ($H$) equation:

$$H(F_C) = -\sum_{i,j} P(F_C(i,j)) \log P(F_C(i,j))$$  (3.2)

where $F_C$ is the fixation map of a given rendering and light field, $F_C(i,j)$ is the value (0 or 1) of the fixation map at row $i$ and column $j$, and $P(F_C(i,j))$ is the probability of a 1 occurring at location $F_c(i,j)$.

The results show that all-in-focus renderings have higher entropy values on average compared to region-in-focus and focal-sweep ones, which suggests participants were more focused on average in the focal-sweep and region-in-focus cases, compared to the all-in-focus
<table>
<thead>
<tr>
<th>Light fields</th>
<th>AiF</th>
<th>RiF1</th>
<th>RiF2</th>
<th>B2F</th>
<th>F2B</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boardgames</td>
<td>4.06</td>
<td>3.87</td>
<td>3.81</td>
<td>3.96</td>
<td>3.82</td>
</tr>
<tr>
<td>Church</td>
<td>4.12</td>
<td>3.46</td>
<td>3.64</td>
<td>3.48</td>
<td>3.41</td>
</tr>
<tr>
<td>Couch</td>
<td>4.27</td>
<td>4.04</td>
<td>3.82</td>
<td>3.64</td>
<td>3.52</td>
</tr>
<tr>
<td>Dino</td>
<td>3.92</td>
<td>3.82</td>
<td>3.61</td>
<td>3.82</td>
<td>3.67</td>
</tr>
<tr>
<td>Dishes</td>
<td>4.19</td>
<td>4.06</td>
<td>3.26</td>
<td>3.48</td>
<td>3.50</td>
</tr>
<tr>
<td>Friends</td>
<td>4.55</td>
<td>4.49</td>
<td>4.45</td>
<td>4.25</td>
<td>4.45</td>
</tr>
<tr>
<td>Lego Knights</td>
<td>4.02</td>
<td>3.45</td>
<td>3.62</td>
<td>3.27</td>
<td>3.34</td>
</tr>
<tr>
<td>Mansion</td>
<td>4.12</td>
<td>4.04</td>
<td>3.67</td>
<td>3.48</td>
<td>3.24</td>
</tr>
<tr>
<td>Medieval</td>
<td>4.02</td>
<td>3.71</td>
<td>3.83</td>
<td>3.65</td>
<td>3.76</td>
</tr>
<tr>
<td>Pens</td>
<td>3.91</td>
<td>3.68</td>
<td>3.79</td>
<td>3.84</td>
<td>3.80</td>
</tr>
<tr>
<td>Platonic</td>
<td>3.79</td>
<td>3.40</td>
<td>3.14</td>
<td>3.53</td>
<td>3.55</td>
</tr>
<tr>
<td>Sideboard</td>
<td>3.79</td>
<td>3.75</td>
<td>3.81</td>
<td>3.98</td>
<td>3.90</td>
</tr>
<tr>
<td>Table</td>
<td>3.85</td>
<td>3.47</td>
<td>3.85</td>
<td>3.76</td>
<td>3.65</td>
</tr>
<tr>
<td>Tarot-Large</td>
<td>4.92</td>
<td>3.46</td>
<td>4.39</td>
<td>3.44</td>
<td>3.73</td>
</tr>
<tr>
<td>Tarot-Small</td>
<td>4.62</td>
<td>3.85</td>
<td>3.80</td>
<td>3.91</td>
<td>4.16</td>
</tr>
<tr>
<td>Tower</td>
<td>3.77</td>
<td>3.72</td>
<td>3.84</td>
<td>3.63</td>
<td>3.42</td>
</tr>
<tr>
<td>Town</td>
<td>4.02</td>
<td>4.03</td>
<td>4.06</td>
<td>4.02</td>
<td>4.02</td>
</tr>
<tr>
<td>Treasure</td>
<td>4.01</td>
<td>3.47</td>
<td>3.66</td>
<td>3.54</td>
<td>3.46</td>
</tr>
<tr>
<td>Vespa</td>
<td>3.81</td>
<td>3.84</td>
<td>3.78</td>
<td>3.45</td>
<td>3.82</td>
</tr>
<tr>
<td>Vinyl</td>
<td>4.18</td>
<td>3.60</td>
<td>3.87</td>
<td>4.16</td>
<td>3.77</td>
</tr>
<tr>
<td><strong>Average</strong></td>
<td>4.10</td>
<td>3.76</td>
<td>3.78</td>
<td>3.71</td>
<td>3.70</td>
</tr>
</tbody>
</table>
case. A two-tailed t-test confirmed that all-in-focus was significantly different than the others ($\alpha = 0.05$). The differences among the other cases were not statistically significant.

### 3.5.3 Analysis of the Influence of the Focus Cue

To analyse the influence of the focus cue, we used the fixation data from the focal sweep renderings. We approximated the planes of fixation for each participant for the focal-sweep renderings (back-to-front and front-to-back) and plotted them against the focal planes rendered at the time of fixation. This was done by first expanding the depth map to be the same resolution/size as the ground truth data so it could be used to map the fixation points to a focal plane. Thus, for each focal plane we have a set of pixels which are ‘in-focus’ at this plane. For each fixation, we checked which focal planes contain a pixel coordinate within less than 35 pixels of their $L_2$-norm with the fixation point. Given a fixation point $\vec{x}_k$, for each the pixel coordinate $\vec{x}_i$, this is formulated as $||\vec{x}_i||_2 = \sqrt{x_i^2 + x_k^2} < 35\text{px}$. We then selected the focal plane which is closest to the rendered plane. This was done for all fixations for each combination of participant, focal-sweep rendering and light field. In order to determine how closely the viewer followed the in-focus regions, we utilised both a qualitative and quantitative analysis, as outlined below.

**$R^2$ for Summarising the Influence of Focus Cue**

For a given light field front-to-back or back-to-front rendering, we began by creating plots of the actual rendered focal plane vs. the focal plane we recorded the viewer looking at, shown in figures [3.8] and [3.9]. Visual inspection of these fixation plots reveal a lot of information, however, it’s difficult to quantify from this the influence of the focus cue. To address this, we created a simple linear model for each light field, under the assumption that the viewer approximately linearly follows the focus cue. These models are shown on the right hand side of figures [3.8] and [3.9]. The intercept and coefficient provide some flexibility in predicting shallow and deep renderings. Once this model has been fit, the $R^2$ coefficient of determination of the model can then be used as a metric to measure how well viewer focus can be predicted from the focal plane. Formally, the $R^2$ score measures, for a regression model, the proportion of variance in the dependent variable that can be explained by the independent variable. In our case, this measure indicates how well a regression model based on the rendered planes fits the observed data i.e. the recorded planes. The $R^2$ score is computed as follows:

$$R^2(y, \hat{y}) = 1 - \frac{\sum_{i=0}^{n-1} (y_i - \hat{y}_i)^2}{\sum_{i=0}^{n-1} (y_i - \bar{y})^2}$$  \hspace{1cm} (3.3)

where $y$ represents the dependent/target variable (the recorded planes), $\hat{y}$ represents the corresponding model prediction (the rendered planes) and $\bar{y}$ is the mean target value over n samples.
(the mean plane over 21 participants).

In our case, a high $R^2$ value intuitively indicates that the focal planes that people actually look at can be well predicted from the focal plane rendered, with a low value indicating the converse. For example, in Fig. 3.7, a strong linear relationship between the recorded focal plane viewed vs. the focal plane rendered, with a corresponding line of best fit matching the data relatively well and an associated $R^2$ value of 0.89. In contrast, the Couch front-to-back shows a line of best fit poorly matching the data and with a correspondingly low $R^2$ value of 0.01.

(a) Tarot-Small front-to-back rendering with a high $R^2$ value

(b) Couch back-to-front rendering with a low $R^2$ value

Figure 3.7: Fixation plots on the left, dense fixation in the middle. On the right the mean and standard deviation of fixations at each rendered focal plan are shown, as well as the line of best fit and associated $R^2$ value. These renderings with fixation data were chosen for the sake of illustrating how $R^2$ can capture the influence of the focus cue.

This matches our intuition, where clearly the fixation plots in Fig. 3.7 illustrate the viewer following the focus cue in the top plot and largely ignoring it in the bottom plot. Hence, we hypothesise that the $R^2$ value of a simple linear model fit to fixation data is a suitable metric to summarise the influence of the focus cue. However, this clearly has shortcomings: for example, if the user fixation stays on the middle rendering at all times, a simple linear model will fit this
data well and hence a high $R^2$ value will be reported, but clearly the focus cue is of no real influence in this case. In practice this rarely occurs, as can be seen in figures 3.8 and 3.9.

**Analysis Using $R^2$ and Fixation Data**

We first note that for some light fields, both focal-sweep renderings had a high $R^2$ score, for others there was a high score for one rendering and a low score for the other and there were some with a low score for both. Density plots of $R^2$ values shown in Fig. 3.10 suggest that for front-to-back renderings, viewers generally follow the focus cue more strongly than for back-to-front renderings. Examination of the individual $R^2$ scores as well as the fixation data plots corroborates this, which reveal there is a noticeable discrepancy in the $R^2$ values for the same light field but for the different focal-sweep directions front-to-back or back-to-front. Notable examples include Dishes, Medieval, Tower, Friends and Couch. However, for the Sideboard and Boardgames light fields, the opposite was observed, where the viewer more strongly followed the focus cue for the back-to-front renderings.

Overall 70% of the $R^2$ scores were over 0.5 which means that over 50% of the data fit the regression model for 70% of the light fields. Additionally, the majority (> 50%) of the light fields have an $R^2$ over 0.63 which means that over 63% of the data fit the regression model for half of the light fields. This gives a good indication that the focus cue has a strong influence on visual attention in general.

The very high score for both renderings of the Tarot-Small light field indicates that the focus cue has a strong influence on attention. This linear correlation can be observed in plots (a) and (b) in Fig. 3.8. There is also a high score for both renderings of the table light field (c) and (d) of Fig. 3.8 however the planes of fixation mainly consist of mid-ground planes. This indicates that focus only influences the attention within this area.

The large visual differences between the plots of the two renderings for some light fields suggest that the order in which focal slices are displayed influences where participants look. This is very prominent in the plots of the Couch light field (e) and (f) in Fig. 3.8. The plots of the front-to-back rendering (e) imply that participants mainly fixate in the background and at plane 120 however participants tend to fixate on the background and on plane 0 in the foreground when viewing the back-to-front rendering (f).

The order also has an affect on how much the focus cue guides attention. For example there is a linear correlation to be seen in the front-to-back rendering of Boardgames light field (g) in Fig. 3.9 and the back-to-front rendering of the Medieval light field (j) in Fig. 3.9 however there is no observable linear correlation in the other renderings (h) in Fig. 3.9 of Boardgames and (i) in Fig. 3.9 of Medieval.

As seen in the Couch light field (e) and (f) in Fig. 3.8 there are planes where users consis-
Figure 3.8: Three plots of the recorded focal plane of fixation vs the rendered focal plane for the front-to-back and back-to-front renderings of three light fields. The first plot contains a point for each fixation at the mid-plane that fixation occurs and each colour represents a participant. The dense fixations contain a point for all the planes rendered for the duration of a fixation. The final plot shows the mean and standard deviation of these dense points across participants.
Figure 3.9: Three plots of the recorded focal plane of fixation vs. the rendered focal plane for the front-to-back and back-to-front renderings of three light fields. The first plot contains a point for each fixation at the mid-plane that fixation occurs and each colour represents a participant. The dense fixations contain a point for all the planes rendered for the duration of a fixation. The final plot shows the mean and standard deviation of these dense points across participants.
Table 3.2: $R^2$ values of the plots of recorded focal planes against the rendered focal planes of each light field for the back-to-front focal-sweep ($B2F$), and front-to-back focal-sweep ($F2B$) renderings, along with the difference between the two. Shown in descending order of the mean of the two.

<table>
<thead>
<tr>
<th>Light fields</th>
<th>$R^2$ F2B</th>
<th>$R^2$ B2F</th>
<th>$R^2$ F2B - $R^2$ B2F</th>
<th>Mean $R^2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tarot_Small</td>
<td>0.886</td>
<td>0.835</td>
<td>0.051</td>
<td>0.86</td>
</tr>
<tr>
<td>Mansion</td>
<td>0.826</td>
<td>0.872</td>
<td>-0.046</td>
<td>0.849</td>
</tr>
<tr>
<td>Town</td>
<td>0.884</td>
<td>0.667</td>
<td>0.217</td>
<td>0.775</td>
</tr>
<tr>
<td>Table</td>
<td>0.742</td>
<td>0.762</td>
<td>-0.020</td>
<td>0.752</td>
</tr>
<tr>
<td>Dino</td>
<td>0.551</td>
<td>0.777</td>
<td>-0.226</td>
<td>0.664</td>
</tr>
<tr>
<td>Lego Knights</td>
<td>0.569</td>
<td>0.706</td>
<td>-0.137</td>
<td>0.637</td>
</tr>
<tr>
<td>Dishes</td>
<td>0.919</td>
<td>0.293</td>
<td>0.626</td>
<td>0.606</td>
</tr>
<tr>
<td>Vinyl</td>
<td>0.748</td>
<td>0.436</td>
<td>0.312</td>
<td>0.592</td>
</tr>
<tr>
<td>Treasure</td>
<td>0.631</td>
<td>0.532</td>
<td>0.099</td>
<td>0.582</td>
</tr>
<tr>
<td>Platonic</td>
<td>0.362</td>
<td>0.778</td>
<td>-0.416</td>
<td>0.57</td>
</tr>
<tr>
<td>Medieval</td>
<td>0.869</td>
<td>0.24</td>
<td>0.629</td>
<td>0.554</td>
</tr>
<tr>
<td>Tarot_Large</td>
<td>0.359</td>
<td>0.748</td>
<td>-0.389</td>
<td>0.554</td>
</tr>
<tr>
<td>Pens</td>
<td>0.501</td>
<td>0.532</td>
<td>-0.031</td>
<td>0.516</td>
</tr>
<tr>
<td>Boardgames</td>
<td>0.105</td>
<td>0.884</td>
<td>-0.779</td>
<td>0.494</td>
</tr>
<tr>
<td>Sideboard</td>
<td>0.009</td>
<td>0.937</td>
<td>-0.928</td>
<td>0.473</td>
</tr>
<tr>
<td>Tower</td>
<td>0.764</td>
<td>0.118</td>
<td>0.646</td>
<td>0.441</td>
</tr>
<tr>
<td>Friends</td>
<td>0.689</td>
<td>0.192</td>
<td>0.497</td>
<td>0.441</td>
</tr>
<tr>
<td>Church</td>
<td>0.52</td>
<td>0.335</td>
<td>0.185</td>
<td>0.428</td>
</tr>
<tr>
<td>Vespa</td>
<td>0.579</td>
<td>0.195</td>
<td>0.384</td>
<td>0.387</td>
</tr>
<tr>
<td>Couch</td>
<td>0.536</td>
<td>0.009</td>
<td>0.527</td>
<td>0.272</td>
</tr>
</tbody>
</table>
Figure 3.10: Density plots of the $R^2$ values using the fixation data. Note that front-to-back and back-to-front renderings have different densities, implying that there is a distinct difference between the way that people visually explore images depending on how they are rendered.

tently fixate regardless of the focus. This is evident in the plots of other light fields, for example the Pens light field (k) and (l) in Fig. 3.9. For both focal-sweep renderings there is a plane in the foreground that users fixate on regardless of focus however other than the points on this plane there is a linear correlation in the data.

Based on these observations and analysis, we drew the following conclusions which we used when designing our models in chapters 4 and 5. First, clearly the focus cue has a significant effect on saliency, which can be clearly observed in figures 3.8 and 3.9.

### 3.5.4 Limitations of the Datasets Used and Collected

In this section, we briefly highlight the limitations of the dataset (Section 3.3.1), used to gather the light field saliency data, which could be addressed in future work. The purpose of this work was to specifically investigate visual attention in relation to light fields, however the datasets which were used were not specifically designed for this purpose. As outlined in [29], the evaluation of data-driven models, especially in deep networks, require large amounts of diverse data.
In an ideal setting, the data would be sampled to fairly represent important features including: the number of salient objects/regions, distributions of normalised object distances from image centres, number of focal slices etc. Other critical features identified in [67] include intensity (or intensity contrast/luminance contrast), colour and orientation. However, we note that this domain is one which is not easy to describe, so therefore, any such dataset will still likely be missing other important features which the model should learn.

Further, as noted in Section 3.3.3 the participants used for this study were all from Trinity College Dublin and were not particularly diverse in age or educational background. This would likely mean that the saliency data gathered does not accurately represent the distribution across the population. In future work, we would recommend a careful evaluation of any models derived using this data to ensure that they aren’t biased towards the prominent demographics in our sample.

### 3.6 Conclusion

In this chapter, we outlined our investigation into how visual attention is affected by changes in focus to verify whether characteristics specific to light fields influence visual attention. We first outlined the methodology and implementation details of our data collection method, which we anticipate can be used as a foundation for future data collection. We then processed the raw data to generate saliency maps, fixation maps and scanpaths for the sake of analysis.

From analysis of the scanpaths of light fields, we conclude that there is a difference in visual attention of static renderings of light fields when compared to focally-varying renderings. This was reinforced by examining the saliency maps of the different rendering types. We found that visual attention was often guided by focus and objects/regions at the focal plane by observing saliency maps computed on segments of the renderings over time and analysing recorded focal plane vs. rendered focal plane plots.

The ground truth data of previous light field saliency works are segmented objects of all-in-focus renderings. We found that these do not fully capture the visual attention of light fields. Salient information not present in the all-in-focus planar rendering is often revealed by saliency maps of focal-sweep renderings. It is also apparent in saliency maps that viewers did not only fixate on objects. Furthermore, the saliency maps of segmented all-in-focus renderings were more dispersed than those of other renderings. This observation was supported by analysis of the entropy of the fixation data for each rendering where we found that all-in-focus data had highest entropy which suggested greater randomness in the data. This variation in the visual attention of different renderings shows the limitations in the use of a saliency map of only one rendering type as ground truth.

The plots and measured $R^2$ score of recorded focal plane of fixation vs the rendered focal...
plane suggest that the focus cue heavily influences visual attention of some light fields and has at least partial influence on most of the light fields in our dataset. Furthermore, this analysis suggests that the order in which the focal slices are shown affects the saliency and this warrants further exploration. To analyse the strength of the focus cue in guiding visual attention, we arbitrarily set the threshold at which we considered a possible plane of fixation at 35 pixels, however further work could examine the effect of different thresholds. Similarly, distance measures other than the $L_2$-norm could be investigated. These plots are a useful resource to assist in the understanding of what planes contain regions which draw attention despite the influence of the focus cue.

The saliency maps and fixation maps created in this work provide the ground truth data necessary for the work presented in the next chapters, Chapter 4 and Chapter 5, in eye fixation prediction for light fields. Since they depict the likelihood of eye fixation at every point of a light field capture, they have applications in this field among others such as light field rendering and compression.
Chapter 4

Focus Guided Saliency Estimation

We preface this chapter by distinguishing the terms “visual attention” and “saliency”. Visual attention, is taken to be the “true” distribution of saliency for an image, which is a theoretical but useful concept; saliency is taken to mean either ground-truth data (a sample of visual attention) or an estimation of visual attention.

Aiming to create a foundation for light field visual attention prediction, in this chapter we describe a four-dimensional light field saliency field representation analogous to the light field itself. Our analysis in Chapter 3 suggested that operations on the light field have an analogous operation on the saliency field. Thus, upon modelling light field visual attention as this 4D saliency field, we hypothesised that known light field operations could be modified to approximate their analogous saliency field operation.

Following on from our investigation in Chapter 3 into the influence of the focus cue, a cue inherent to the light field, it was evident that the effects of the refocusing operation on saliency was an important factor to incorporate into the model’s architecture. The method modifies an existing view rendering algorithm with focus guidance, obtained from the light field disparity. This facilitates the construction of saliency maps without the need to render the corresponding view itself, which will help to speed up processing operations that are compatible. The results show that the proposed saliency estimation approach yields very good predictions of visual attention across multiple planes of the light field. We anticipate that this approach can be extended for a range of rendering applications. The work in this chapter has been published in [32].

4.1 Introduction

Since light fields are four-dimensional, their captures come with an increased amount of information to take advantage of. This has stimulated ongoing light field specific research into virtual viewpoints [26] and refocusing [57, 50, 58, 59, 56]. However, the computation time
and memory required to perform these operations can make tasks such as real-time rendering impractical. One solution is to exploit the salient information of light fields to focus resources on regions that attract visual attention when using these algorithms. On the one hand, light fields’ increased dimensionality compared to images brings utility: they can be used in many different applications including estimating the geometry or the depth of the scene \[130\], rendering new views from different viewpoints \[26\], and changing the focus (or refocusing) of the scene \[50, 56\], see Fig. 4.1(c). On the other hand, it brings challenges for the visual perception aspects of this particular form of media.

Figure 4.1: Visualisation of (a) an all-in-focus rendering of an LF, (b) its saliency estimation result using a state-of-the-art VA model DeepGaze II, (c) a refocused rendering of an LF, and (d) the result of the proposed saliency estimation method.
Understanding viewers’ visual attention is important for various applications such as compression [69] or rendering and visualization [71] for all media types ranging from 2D images to light fields. It is also crucial to be able to estimate the saliency map before the corresponding view is actually rendered, especially in applications where the saliency map is used in the rendering process itself, e.g. compression and foveated rendering [131]. However, collecting visual attention in user studies is not always feasible, and so to predict the visual attention distribution, automatic saliency estimation algorithms that rely on the image characteristics are used.

In this chapter, we define the concept of light field saliency as the probability of visual importance of every ray of a light field, and introduce the corresponding representation as a saliency field $\Psi$ below (cf. Section 4.3.2). We aim to use this representation to estimate the visual attention of refocused views of the light field, see Fig. 4.1. To achieve this, we propose a focus guided light field visual attention prediction method. For this method, we modify a classical refocus rendering algorithm by integrating the disparity information relevant for the refocusing operation. The proposed algorithm is validated on a light field visual attention database visually and quantitatively. Our approach can be used to estimate saliency for the refocusing operation without having to render the views, in contrast to current approaches. Our results show that the integration of the focus guidance improves the saliency estimation and helps yield an accurate visual attention prediction.

4.2 Background & Related Work

4.2.1 Shift-and-Sum Light Field Refocusing

A common light field operation is to render a 2D image simulating a traditional photographic camera with a narrow depth of field, with the ability to choose the focal plane, also called refocusing. A refocus image $I_r$ can be produced through use of the well-known shift-and-sum algorithm [50], in which it is obtained as a linear combination of shifted light field SAIs:

$$I_r(s, t, \delta_F) = \sum_{u,v} A(u, v) I_{u,v}(s_F, t_F), \quad (4.1)$$

where $(u_r, v_r)$ corresponds to the position of the refocus image on the camera plane, $\delta_F$ is the disparity value corresponding to the focus distance with $(s_F, t_F)$ the corresponding pixel shift, and $A$ is a filter that defines the synthetic aperture. Intuitively, the shift-and-sum algorithm
aligns the regions of the SAIs corresponding to the target disparity $\delta_F$. High frequency textures and edges are thus preserved for these regions, but blurred otherwise. Increasing the size of the aperture filter $A$ will combine more SAIs and result in a shallower depth of field, as for traditional cameras.

### 4.3 Proposed Method

![Pipeline of the FGSE method for focus guided light field saliency estimation](image)

In this section, we describe the focus guided saliency estimation (FGSE) algorithm for light fields and its integration into a rendering framework. The pipeline is illustrated in Fig. 4.2.

#### 4.3.1 Focus Guided Saliency

As explained in the introduction, we define in this chapter the concept of light field saliency as the probability of visual importance for every ray of a light field. Formally, we denote the saliency field as a 4D function $\Psi(u, v, s, t)$. For convenience, we define a “saliency SAI” as $\Psi_{u,v}(s,t) = \Psi(u, v, s, t)$. While light field SAIs are natural images, which contain low to high frequencies, saliency SAIs only contain low frequencies and do not have high frequency textures or edges, see Fig. 4.1(b). Therefore, the existing shift-and-sum refocusing algorithm described in (4.1) can not be directly applied to the saliency field.

Based on the assumption that gaze is attracted by in-focus regions as examined in Chapter 3, we proposed to process the saliency SAIs using a modified shift-and-sum algorithm guided by
a focus map. We obtained the focus map from the disparity maps estimated from the light field, and denote the 4D disparity field as \( D_{p, u, v, s, t} \). The saliency field \( \Psi \) is obtained by independently estimating the saliency of the light field SAIs with the 2D saliency estimator DeepGaze II [7]. The overall pipeline of the proposed approach is shown in Fig. 4.2.

As the ground truth visual attention maps were obtained by applying Gaussian filtering (corresponding to \( 1^\circ \) visual angle) at fixation points to take visual acuity into consideration, we applied the same blur to the disparity field input of our method in order to closer match the properties of these visual attention maps. In Chapter 3, visual attention maps were obtained with a Gaussian kernel:

\[
B(s, t) = \frac{1}{2\pi \sigma_s \sigma_t} \exp\left(-\frac{s^2}{2\sigma_s^2} - \frac{t^2}{2\sigma_t^2}\right)
\]  

(4.2)

where \( \sigma_s = \frac{47.66}{1080} U \) and \( \sigma_t = \frac{42.67}{1080} V \). We used the same Gaussian kernel to blur the disparity maps:

\[
D_b(u, v, s, t) = B(s, t) \odot D(u, v, s, t), \forall (u, v)
\]  

(4.3)

where \( \odot \) is the convolution operator. In addition to approximating the visual attention maps’ properties, blurring the disparity maps is also advantageous as we can use fast disparity estimation, for which errors are removed by the blurring process. In our experiment we used the method proposed by Chen at al. to estimate the disparity field [130].

The 5D focus map \( F(u, v, s, t, \delta) \) could then be obtained from the blurred disparity field for a given target disparity \( \delta \) as:

\[
F(u, v, s, t, \delta) = \exp\left(-\frac{||D_b(u, v, s, t) - \delta||^2}{\sigma_F^2}\right)
\]  

(4.4)

We chose to express the focus map as soft probability using a Gaussian distribution rather than a binary mask to compensate for remaining errors in the blurred disparity field. The parameter \( \sigma_F \) controls the “depth of field” of the focus map. We observed in our analysis of the ground truth visual attention maps in Chapter 3 that visual attention often depends on the strength of the defocus blur. As the maximum strength of the defocus blur depends on the maximum disparity, we introduced an intermediate parameter \( \sigma_D \), such that \( \sigma_F = \sigma_D \cdot (max(D) - min(D)) \), where \( max(D) \) and \( min(D) \) are the higher and lower bound of the disparity range respectively. The parameter \( \sigma_D \) allows for easy controlling of the focus map depth of field for all light fields in the dataset, rather than experimentally defining \( \sigma_F \) for each light field.
4.3.2 Integration into Rendering

The main idea of the focus-guided rendering method was to modify the shift-and-sum algorithm to weight the saliency SAIs with the focus map:

\[
\Psi_r(s, t, \delta_F) = \sum_{u,v} A(u, v) F(u, v, s_F, t_F, \delta_F) \Psi_{u,v}(s_F, t_F)
\]  \hspace{1cm} (4.5)

By the properties of the shift and sum, all the shifted focus maps were aligned and almost equal. In addition, given that the saliency SAIs are composed of low frequency values, we could use the following approximation:

\[
F_r(s, t, \delta_F) \simeq F(u, v, s_F, t_F, \delta_F), \forall(u, v)
\]  \hspace{1cm} (4.6)

The algorithm could thus be simplified as:

\[
\Psi_r(s, t, \delta_F) = F_r(s, t, \delta_F) \sum_{u,v} A(u, v) \Psi_{u,v}(s_F, t_F)
\]  \hspace{1cm} (4.7)

With this simplification we observed experimentally that the processing time is 25% faster compared to the direct approach of (4.5), while maintaining similar saliency estimation performance (see Table 4.1).

4.4 Experimental Results

To validate the focus guided saliency estimation approach for light fields, we made use of a light field visual attention database and compared our results with a state-of-the-art saliency estimation method. Here, we briefly describe the database and selected saliency estimation method. Then, we’ll present and discuss results.

4.4.1 Dataset

To evaluate our light field rendering approach. We used the ground truth visual attention data we collected as discussed in Chapter 3. This dataset was chosen as it is the only one that has collected eye-fixation data and for light field renderings on multiple planes of focus. This was

1 For more details and the code, see https://v-sense.scss.tcd.ie/research/light-fields/light-field-saliency-estimation
necessary for our research in building and substantiating a four-dimensional light field saliency field.

We only consider the 34 stimuli from this database that correspond to refocus images of 2D full-parallax light fields - two different focal renderings of 17 light fields. These renderings were named as “Region-1” and “Region-2” in Chapter 3 and we keep the same notation in this chapter for consistency.

### 4.4.2 Saliency Estimation Method

For both the estimation of the saliency SAIs $\Psi_{u,v}$, and as an anchor metric for validation, DeepGaze II [84] was selected as one of the highest performing saliency estimation algorithms according to MIT/Tübingen Saliency Benchmark [30]. It takes as input a regular 2D-image and outputs a saliency map which represents the likelihood of eye fixation.

### 4.4.3 Quantitative Analysis

In Table 4.1, evaluation metrics are reported for the two variations of the focus guided rendering method described in (4.5) and (4.7), with and without the disparity blurring described in (4.2) and (4.3). The algorithm with blur applied performs better than that without. The simplification of the algorithm using (4.7) results in a worse score for KLD but improvements in the NSS, CC and SIM metrics.

<table>
<thead>
<tr>
<th>Saliency method</th>
<th>AUC↑</th>
<th>NSS↑</th>
<th>CC↑</th>
<th>KLD↓</th>
<th>SIM↑</th>
</tr>
</thead>
<tbody>
<tr>
<td>FGSE Eq. 4.5 - w/o blur</td>
<td>0.844</td>
<td>1.614</td>
<td>0.672</td>
<td>0.659</td>
<td>0.636</td>
</tr>
<tr>
<td>FGSE Eq. 4.7 - w/o blur</td>
<td>0.844</td>
<td>1.608</td>
<td>0.671</td>
<td>0.680</td>
<td>0.635</td>
</tr>
<tr>
<td>FGSE Eq. 4.5 - w/ blur</td>
<td><strong>0.845</strong></td>
<td>1.615</td>
<td>0.678</td>
<td><strong>0.616</strong></td>
<td>0.639</td>
</tr>
<tr>
<td>FGSE Eq. 4.7 - w/ blur</td>
<td><strong>0.845</strong></td>
<td><strong>1.618</strong></td>
<td><strong>0.680</strong></td>
<td>0.619</td>
<td><strong>0.640</strong></td>
</tr>
</tbody>
</table>

↑All FGSE methods use $\sigma_D = 0.4$. **Boldface** indicates the best result in each column.

In Table 4.2, we compare the performance of our estimator for different values of $\sigma_D$ to that of DeepGaze II run on the refocused rendering and the no focus guidance baseline.

Our estimated saliency method achieves the best score for the CC metric and results comparable to DeepGaze II for the other metrics. The worse AUC score but very close NSS score compared to DeepGaze II suggests that our model has less low valued false positives but also less intense saliency at fixation locations. This demonstrates the effectiveness of our algorithm, considering that DeepGaze II needs the light field rendering operation to be completed before saliency estimation, whereas our method computes the entire saliency field from only the SAI.
Table 4.2: Metric results\(^\dagger\) for the FGSE method compared with the baseline shift & sum saliency estimation (SSSE) without focus guidance.

<table>
<thead>
<tr>
<th>Saliency method</th>
<th>AUC↑</th>
<th>NSS↑</th>
<th>CC↑</th>
<th>KLD↓</th>
<th>SIM↑</th>
</tr>
</thead>
<tbody>
<tr>
<td>SSSE</td>
<td>0.817</td>
<td>1.348</td>
<td>0.568</td>
<td>0.695</td>
<td>0.583</td>
</tr>
<tr>
<td>FGSE(_{\sigma_D=0.7})</td>
<td>0.831</td>
<td>1.463</td>
<td>0.618</td>
<td>0.627</td>
<td>0.610</td>
</tr>
<tr>
<td>FGSE(_{\sigma_D=0.6})</td>
<td>0.834</td>
<td>1.497</td>
<td>0.632</td>
<td>0.614</td>
<td>0.618</td>
</tr>
<tr>
<td>FGSE(_{\sigma_D=0.5})</td>
<td>0.839</td>
<td>1.546</td>
<td>0.652</td>
<td>0.602</td>
<td>0.628</td>
</tr>
<tr>
<td>FGSE(_{\sigma_D=0.4})</td>
<td>0.845</td>
<td>1.618</td>
<td>0.680</td>
<td>0.619</td>
<td>0.640</td>
</tr>
<tr>
<td>FGSE(_{\sigma_D=0.3})</td>
<td>0.847</td>
<td>1.713</td>
<td>0.713</td>
<td>0.790</td>
<td>0.649</td>
</tr>
<tr>
<td>FGSE(_{\sigma_D=0.2})</td>
<td>0.835</td>
<td>1.744</td>
<td>0.717</td>
<td>1.445</td>
<td>0.629</td>
</tr>
<tr>
<td>FGSE(_{\sigma_D=0.1})</td>
<td>0.781</td>
<td>1.572</td>
<td>0.637</td>
<td>3.882</td>
<td>0.649</td>
</tr>
<tr>
<td>DeepGaze II</td>
<td>0.851</td>
<td>1.745</td>
<td>0.703</td>
<td>0.585</td>
<td>0.653</td>
</tr>
</tbody>
</table>

\(^\dagger\)DeepGaze II results are reported for readers’ information. **Boldface** indicates the best score for each column, and *Italic* indicates the best results for the FGSE method.

input without rendering. Overall, the differences between the metric values of DeepGaze II and the highest values of the FGSE method (for different \(\sigma_D\) values) are marginal. Additionally, FGSE beats the baseline SSSE with respect to AUC, NSS, CC, and SIM scores except for two instances. The KLD scores are mixed as the dissimilarity depends on the spread of the estimated saliency map. However, for higher \(\sigma_D\), FGSE attains lower (i.e., better) KLD values compared to SSSE, and for some \(\sigma_D\), it scores close to the KLD value of DeepGaze II. Considering all the metrics, we chose \(\sigma_D = 0.4\) where our model performs well overall, with high AUC, NSS, CC, and SIM scores and low KLD.

Upon observation of model performance with different \(\sigma_D\), we see that there is a bias-variance trade-off in choosing \(\sigma_D\). The performance decreases at the extremities 0.7 and 0.1 and is best between 0.3 and 0.5. Lowering \(\sigma_D\) increases the influence of focus guidance as the “depth of field” of the focus map increases. Very low \(\sigma_D\) causes high bias in our estimator as the model is too simplistic and only considers the focus map. Thus, less emphasis is put on saliency prediction of the overall image. Conversely, very high \(\sigma_D\) leads to high variance of the model and it places little weight on the in-focus region. High \(\sigma_D\) means that the predicted saliency maps for all renderings mostly resemble the SAI saliency estimation. We found that \(\sigma_D = 0.4\) balances this trade-off the best for the stimuli we tested across all metrics.

### 4.4.4 Qualitative Analysis

The qualitative performance of our model’s output is demonstrated in Fig. 4.3, which displays the saliency maps outputted by our FGSE method using the simplification of (4.7) with blurred disparity map \((D_b)\), and \(\sigma_D = 0.4\). For comparison, we also provide images alongside our
Figure 4.3: The results of our focus guided model (FGSE) shown alongside the stimulus, the ground truth fixations and VA map as well as the no focus guidance (SSSE) baseline and the results of DeepGaze II run directly on the stimulus. Here the stimuli are the Region-1 and Region-2 renderings of a selection of the light fields tested.
predictions of the following: RGB stimulus, ground truth (GT) fixations, corresponding GT visual attention maps, DeepGaze II run directly on each stimulus, and the baseline shift & sum saliency estimation (SSSE) without focus guidance. For each light field, we used two stimuli to test our model: renderings Region-1 and Region-2. These were chosen because they have sufficiently distant planes rendered to be in-focus, emphasising the difference when refocusing. Our model’s output can be broken down into two main components.

Firstly, our model closely estimates the variations in concentration of the saliency observed in the ground truth at certain regions. This concentration depends on whether or not the regions appear on the focal plane and therefore guides visual attention. These differences can be observed between the renderings of the Vespa light field Fig. 4.3(c) and (d).

Secondly, our model accurately matches the underlying visual attention across different renderings of the same light field, particularly when a shift in the salient region is observed in the ground truth due to varying the focal plane. This is seen when eye gaze follows a line of focus, for example in the Vinyl light field Fig. 4.3(e) and (f). Our model better predicts visual attention shifts between objects as they come in and out of focus compared to the baseline, and is at least on par with the DeepGaze II model run directly on the stimuli as seen in the Tower light field Fig. 4.3(a) and (b).

As discussed in the previous section, the $\sigma_D$ parameter in our algorithm controls the “depth of field” of the focus map used to generate the estimated saliency map. Consequently, increasing $\sigma_D$ decreases the influence of the focus guidance on our models predictions. For example, in the Board games Region-2 rendering Fig. 4.4, the lower $\sigma_D = 0.2$ produces a more visually similar map to the ground truth. For $\sigma_D = 0.4$, the estimated saliency over extends outside the region of focus.

### 4.4.5 Computation Time Analysis

The computation times of our FGSE model for each of the light fields are given in Table 4.3. This is the time it takes to output a saliency map corresponding to a given a region-in-focus
rendering (single image). However, the times shown in Table 4.3 are negligible when you consider that the FGSE model also requires DeepGaze II saliency maps of each SAI of the light field’s multi-view array as input. Since generating a saliency map using DeepGaze II takes approximately 9 seconds per SAI, this is comparatively a much more computationally expensive step in the pipeline.

The approximate computation times for this process for the light fields in our database were calculated as follows: For the light fields from the Stanford (New) Light Field Archive \[40\] namely Lego Knights, Tarot-Large, Tarot-Small and Treasure, whose multi-view array consists of a 17 x 17 grid of SAIs, the corresponding array of saliency maps takes approximately 2061 seconds to compute. For the EPFL Dataset \[127\] light fields, Vespa and Friends, which consist of 15 x 15 multi-view arrays, generating the array of saliency maps takes approximately 2025 seconds. Likewise, for remaining light fields, from the HCI Heidelberg 4D Light Field Dataset \[54\], the computation time for obtaining saliency maps of their 9 x 9 multi-view array is 729 seconds. It’s imperative to note that this is only one sample and that these times could vary depending on the exact light field used, which future research could address.

In the next section, as a means of reducing the number of SAIs and hence the number of DeepGaze II computations required for the FGSE method, we explore the effect that downsampling the input light field multi-view array has on the FGSE metric results.

<table>
<thead>
<tr>
<th>Light Field</th>
<th>Computation time (in seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lego Knights</td>
<td>1.17</td>
</tr>
<tr>
<td>Tarot-Large</td>
<td>1.11</td>
</tr>
<tr>
<td>Tarot-Small</td>
<td>1.08</td>
</tr>
<tr>
<td>Treasure</td>
<td>2.29</td>
</tr>
<tr>
<td>Vespa</td>
<td>0.24</td>
</tr>
<tr>
<td>Friends</td>
<td>0.33</td>
</tr>
<tr>
<td>Pens</td>
<td>0.12</td>
</tr>
<tr>
<td>Dishes</td>
<td>0.12</td>
</tr>
<tr>
<td>Sideboard</td>
<td>0.11</td>
</tr>
<tr>
<td>Boardgames</td>
<td>0.11</td>
</tr>
<tr>
<td>Medieval</td>
<td>0.12</td>
</tr>
<tr>
<td>Dino</td>
<td>0.12</td>
</tr>
<tr>
<td>Platonic</td>
<td>0.11</td>
</tr>
<tr>
<td>Table</td>
<td>0.11</td>
</tr>
<tr>
<td>Tower</td>
<td>0.12</td>
</tr>
<tr>
<td>Town</td>
<td>0.11</td>
</tr>
<tr>
<td>Vinyl</td>
<td>0.11</td>
</tr>
</tbody>
</table>

\(\dagger\) All FGSE methods use \(\sigma_D = 0.4\).
To investigate the impact downsampling the input sub-aperture array has on our model, we ran our FGSE model with the centre SAI and four SAI at corners as input and also with only the centre SAI.

![Stimulus GT Fixations](image)

Figure 4.5: The results of our focus guided model FGSE with downsampled multi-view array input, FGSE centre only and FGSE centre and four corners shown alongside the stimulus, the ground truth fixations and the VA map. Here the stimuli are the Region-1 and Region-2 renderings of a selection of the light fields tested.

For most light fields there is no visual difference and little numerical difference between downsampled results and results using the full SAI array. An example of this is evident in the results of the Boardgames light field in Fig. 4.5 and Table 4.4. However, for some of the light fields with larger SAI arrays, there is a small but noticeable visual and numerical difference. Upon observation of the saliency maps and metric results of the Tarot-Large light field Fig. 4.5 and Table 4.4, this difference can be seen.

The results averaged over the two renderings Region-1 and Region-2 over all the light fields can be observed in Table 4.5. Although the results are worse for the models with downsampled inputs there is a trade-off to be found. Since the saliency map of each SAI image takes DeepGaze II approximately 9 seconds to compute, and the size of the SAI arrays range from 81 to 289 in the light field visual attention database, using only a small subset of the SAI array as input would drastically cut down on the model run-time.
Table 4.4: Metric results\(^\dagger\) for the FGSE method compared with FGSE method using only the centre SAI and using the centre SAI and four SAI at corners. The average of the Region-1 and Region-2 renderings are shown for two light field examples namely Boardgames and Tarot-Large.

<table>
<thead>
<tr>
<th>Saliency method</th>
<th>Region-1</th>
<th>Region-2</th>
<th>Region-1</th>
<th>Region-2</th>
</tr>
</thead>
<tbody>
<tr>
<td>FGSE Corners</td>
<td>0.823</td>
<td>0.823</td>
<td>0.823</td>
<td>0.811</td>
</tr>
<tr>
<td>FGSE Centre</td>
<td>0.823</td>
<td>0.811</td>
<td>0.811</td>
<td>0.811</td>
</tr>
<tr>
<td>AUC ↑</td>
<td>0.823</td>
<td>0.823</td>
<td>0.823</td>
<td>0.811</td>
</tr>
<tr>
<td>NSS ↑</td>
<td>1.381</td>
<td>1.369</td>
<td>1.369</td>
<td>1.522</td>
</tr>
<tr>
<td>CC ↑</td>
<td>0.625</td>
<td>0.621</td>
<td>0.621</td>
<td>0.646</td>
</tr>
<tr>
<td>KLD ↓</td>
<td>0.53</td>
<td>0.537</td>
<td>0.537</td>
<td>0.668</td>
</tr>
<tr>
<td>SIM ↑</td>
<td>0.617</td>
<td>0.616</td>
<td>0.616</td>
<td>0.574</td>
</tr>
</tbody>
</table>

\(\dagger\) **Boldface** indicates the best score for each row. All FGSE methods use \(\sigma_D = 0.4\).

Table 4.5: Metric results\(^\dagger\) for the FGSE method compared with FGSE method using only the centre SAI and using the centre SAI and four SAI at corners, and the baseline shift & sum saliency estimation (SSSE) without focus guidance averaged over all light fields and the Region-1 and Region-2 renderings.

<table>
<thead>
<tr>
<th>Saliency method</th>
<th>Region-1</th>
<th>Region-2</th>
<th>Region-1</th>
<th>Region-2</th>
</tr>
</thead>
<tbody>
<tr>
<td>FGSE Corners</td>
<td>0.845</td>
<td>1.618</td>
<td>0.680</td>
<td>0.619</td>
</tr>
<tr>
<td>FGSE Centre</td>
<td>0.843</td>
<td>1.584</td>
<td>0.668</td>
<td>0.642</td>
</tr>
<tr>
<td>SSSE</td>
<td>0.817</td>
<td>1.348</td>
<td>0.568</td>
<td>0.695</td>
</tr>
<tr>
<td>DeepGaze II</td>
<td>0.849</td>
<td>1.741</td>
<td>0.709</td>
<td>0.574</td>
</tr>
</tbody>
</table>

\(\dagger\) **DeepGaze II results are reported for readers’ information. Boldface** indicates the best score for each column, and **Italic** indicates the best results for the FGSE method. All FGSE methods use \(\sigma_D = 0.4\).
4.5 Conclusion

In this study, we considered the light field as a scene representation, from which novel views could be rendered. We developed a single cohesive four-dimensional saliency field model for estimating the visual attention of a light field. Rather than treating the saliency of refocused renderings as separate entities, we employed this model and built a saliency field from which the saliency map corresponding to any refocus rendering could be generated. This model lays the groundwork for predicting the saliency field of various types of light field renderings and for generating most salient renderings. Furthermore, this model has the potential to develop light field applications that can guide a viewer’s gaze to desired regions.

We tested the efficacy of our model for visual attention prediction and found that it performs as well as a state-of-the-art visual attention model without the need to render the refocused image. Our model shows that it is possible to generate the saliency of any refocus rendering from only the SAI captures and the disparity map without the need to refocus the entire light field. Our algorithm could be further optimised as a branch of future research to reduce computational complexity, e.g. by estimating the saliency field at lower resolution.

As the $\sigma_D$ parameter controls the extent by which the focal plane affects the saliency estimation, we observed that there is a trade-off when choosing $\sigma_D$. This parameter influences the performance of each individual light field. While in this chapter we took into account the disparity range of each light field, future work could explore additional parameters that influence the “depth of field” of the focus map, such as the aperture size and shape, to automatically compute $\sigma_F$ for each individual light field.

In future work our model can be extended to other refocusing algorithms and combined with view synthesis to generate saliency maps of other novel views. In the next chapter, we apply this saliency field concept to another more recent refocusing method [56].
Chapter 5

Saliency Prediction Using Fourier Disparity Layers

In this chapter, we present a novel layered saliency model for Light Fields (LF) called Fourier Disparity Layer Saliency Estimation (FDLSE). We found that the salient information of an LF can be decomposed into multiple layers. These layers are constructed from the existing Fourier Disparity Layer (FDL) LF representation. Our FDLSE model can be used to predict the visual attention (VA) of any LF rendering with arbitrary viewpoint, aperture and depth-of-focus, without the need to generate the rendered image itself. The proposed model surpasses previous work in the following areas. Our method requires the estimation of the saliency map of only one sub-aperture image instead of the full view array. Furthermore, this model does not require pre-estimated disparity maps, but instead relies on the FDL model whose computation fully takes advantage of GPU parallelisation. Finally, FDLSE shows visual improvements and performs quantitatively on par with our previous FGSE model when evaluated on VA prediction of refocus renderings. To our knowledge these are the only two models which can be used to predict LF VA. The work detailed in this chapter has been published in [33].

5.1 Introduction

For our FGSE model presented in Chapter [4], a 4D saliency field was constructed using a saliency map for each LF SAI. While this method allows high quality saliency estimation for refocus renderings, using such a 4D model has a high computational cost both for the model construction and for the refocusing.

In this chapter, we propose an alternative LF saliency representation that can be constructed from the saliency map of a single SAI, and that allows for efficient estimation of other saliency maps corresponding to arbitrary refocus renderings. The proposed representation is a layered
saliency model inspired by and derived from the Fourier Disparity Layer (FDL) representation of the LF [56]. Building on our previous work in Chapter 4, the only other LF VA prediction model to our knowledge, this chapter outlines a new way of expressing the saliency of LFs for VA prediction. As human visual attention is affected heavily by disparity/depth we utilise a light field representation with the disparity information inherent, to create a better integrated model. As a layered saliency model derived from the FDL representation, our model is thus called FDL saliency estimation (FDLSE). Each saliency layer holds the estimated saliency information corresponding to a focal plane at a certain disparity or depth of the LF. In contrast to the previous work, these disparity values are computed to reflect the composition of the scene, so they are different for each LF. In this chapter, we outline how we construct our layered saliency model by transferring local disparity information from an FDL model to a saliency map of the LF’s central view. We then outline how to use FDLSE to render the estimated 2D saliency map of an LF refocus rendering. Finally, we compare and analyse the output of our model to that of our previous FGSE method.

5.2 Background & Related Work

5.2.1 Fourier Disparity Layer Method

In this work, we make use of the Fourier Disparity Layer method [56] for LF rendering to generate our FDLSE model. This method samples the LF in the depth dimension by decomposing the scene as a discrete sum of layers. As this method samples in the depth dimension, it allows for unlimited angular density (i.e. views at arbitrary angular coordinates can be rendered from the layers). Similarly, the proposed FDLSE represents the full saliency field with only a few layers.

5.3 Method Overview

The overview of our proposed method is depicted in Fig. 5.1. It relies on the knowledge of a saliency map previously estimated for the central view of the LF. As used for FGSE, the DeepGaze II method [7] is used for this initial estimation.

However, while for the FGSE method we applied the DeepGaze II estimator to each LF SAI, in this new method we only need the saliency map of the central view. The approach then consists of transferring the disparity information of the LF to the saliency map in order to construct a saliency layered model. This model can later be used to estimate other saliency maps corresponding to arbitrary renderings of the LF (e.g. refocused images, other SAIs).
Figure 5.1: Overview diagram of the proposed FDLSE method for light field saliency estimation using Fourier disparity layers.

Focus & aperture settings: $d_{\text{focus}}, \sigma_{\text{disp}}$

Saliency layer construction
The disparity transfer is performed by constructing an FDL model from the LF (see Sec. 5.4.1), and then separating the texture from the local disparity information. This local disparity information is extracted in the form of a disparity confidence volume (DCV) made up of a confidence map $c_k$ for each layer $k$ of the FDL, as presented in Sec. 5.4.2. As in previous chapters, we formulate the light field as the 4D function $L(s, t, u, v)$. Each ray in the LF is determined by two parallel planes: the focal plane $(s, t)$ and the camera plane $(u, v)$ containing the spatial and angular information respectively. For a pixel of spatial coordinates $(s, t)$, the DCV value $c_k(s, t)$ estimates the probability that the pixel has the disparity $d_k$. Hence, the DCV at each pixel can be interpreted as a probability mass function (i.e., with values in the range $[0, 1]$ and summing to 1). The saliency layers are then simply obtained by multiplying each map $c_k$ of the DCV by the central view saliency map.

Finally, given the saliency layers, the rendering method detailed in Sec. 5.5 allows a fast rendering of saliency estimates for arbitrary refocused images, and even for other LF SAIs.

5.4 Construction of Saliency Layers

5.4.1 FDL notations and construction

In a preliminary step, we perform calibration and construction of an FDL model from the LF views using [56]. The model is comprised of a set of layers $\{l_k\}_{k=1,n}$, where each layer is associated to a disparity value $d_k$.

First, the FDL calibration jointly determines the accurate angular coordinates $(u_j, v_j)$ of each LF view as well as a fixed number of disparity values $d_k$ that are representative of the LF’s depth distribution in the scene. The FDL model is then constructed such that shifting each layer $l_k$ by a vector $d_k \cdot (u_j, v_j)$, and summing the shifted layers, optimally reconstructs each view $(u_j, v_j)$. This is efficiently performed following [56] based on the principle that the shift operation becomes an element-wise multiplication in the Fourier domain. Hence, the FDL is constructed, by solving a linear least squares problem per-frequency component, which fully takes advantage of modern GPUs capabilities.

At this stage, only global information on the disparity distribution of the whole scene is available in the form of a set of disparity values, since each layer $l_k$ is associated only to a single value $d_k$. To transfer the disparity of the LF to a saliency map, we need to extract local (i.e., pixel-wise) information.
5.4.2 Extraction of a Disparity Confidence Volume for Disparity Transfer

Pre-Processing of the FDL

Since the FDL is originally computed in the Fourier domain, we first need to convert each layer in the pixel domain using the inverse Fourier transform, in order to extract pixel-wise disparity information. Since chromatic information is not needed here, we further convert the layers from RGB data to grayscale intensity. Visualizing the layers in the pixel domain as in Fig. 5.2 shows that each layer mostly contains the textures of objects located at the corresponding disparity. Fig. 5.2(d) further shows that at a given pixel position, the intensity tends to be higher on the layers that have a disparity close to the true pixel’s disparity in the light field. In order to enhance this effect and extract a more selective disparity confidence volume, we pre-process the FDL by raising each pixel value to the power $\gamma_{pre} > 1$. The pre-processed FDL is computed as:

$$\tilde{l}_k(s,t) = l_k(s,t)^{\gamma_{pre}}. \quad (5.1)$$

Figure 5.2: Visualisation of the FDL layers (converted to spatial domain and in grayscale intensity): (a,b,c) Layers 5, 12 and 24 respectively, from an FDL of 30 layers, (d) Intensities over the layers of the pixels at spatial positions A, B and C in (a,b,c).
Disparity Confidence Volume Extraction

For each pixel \((s, t)\) and at each layer of index \(k\), we want to estimate \(c_k(s, t)\) so that a high value (i.e. close to 1) indicates a high probability that the pixel \((s, t)\) has the disparity \(d_k\).

Although \(\tilde{l}_k(s, t)\) generally takes a higher value if the disparity of the pixel \((s, t)\) is close to \(d_k\), this may not hold in the presence of noise, especially in areas without high frequency textures. Hence, the neighbourhood of \((s, t)\) must be considered to compute a meaningful DCV that remains robust to noise. Here, we consider a neighbourhood defined by a 2D Gaussian window centered on \((s, t)\). An initial estimate \(\tilde{c}_k(s, t)\) is then obtained as a weighted sum of pixels of \(\tilde{l}_k\) with the Gaussian weights as follows:

\[
\tilde{c}_k(s, t) = \sum_{(s', t') \in \Omega} \tilde{l}_k(s', t') \cdot e^{-\frac{(s-s')^2}{2\sigma_s^2} - \frac{(t-t')^2}{2\sigma_t^2}},
\]

where \(\Omega\) is the set of pixel coordinates in the image, and \(\sigma_s\) and \(\sigma_t\) are respectively the horizontal and vertical standard deviation of the 2D Gaussian that define the spatial extent of the neighbourhood.

In practice, \(\sigma_s\) and \(\sigma_t\) should be taken sufficiently large for robustness to noise, but also to prevent strong discontinuities in the DCV and thus to obtain smooth final saliency maps, even around depth discontinuities.

Note that \(\tilde{c}_k\) is equivalently expressed as the convolution of \(\tilde{l}_k\) with a 2D Gaussian of horizontal and vertical standard deviations \(\sigma_s\) and \(\sigma_t\). Hence, for fast computations even with large values of \(\sigma_s\) and \(\sigma_t\), we implement that convolution as a frequency-wise multiplication in the Fourier domain:

\[
\hat{\tilde{c}}_k(\omega_s, \omega_t) = \mathcal{F}[\tilde{l}_k](\omega_s, \omega_t) \cdot e^{-2\pi^2(\sigma_s^2\omega_s^2 + \sigma_t^2\omega_t^2)},
\]

\[
\tilde{c}_k = \mathcal{F}^{-1} \left[ \hat{\tilde{c}}_k \right],
\]

where \(\mathcal{F}[\cdot]\) and \(\mathcal{F}^{-1} [\cdot]\) are respectively the forward and inverse 2D discrete Fourier transform (DFT), and \(\omega_s\) and \(\omega_t\) are the spatial frequencies. Here, we used the closed-form expression of the Fourier transform of a Gaussian, which avoids the numerical computation of its DFT.

In order to compensate for the pre-processing that may result in too strong selectivity (i.e. all the DCV values very close to either 0 or 1), we apply another power function to each pixel after the convolution with a power \(\gamma_{post} < 1\). Finally, in order to make the DCV at each pixel coordinate consistent with a probability distribution, we normalise the DCV along the disparity dimension. The final DCV is thus obtained as:

\[
c_k(s, t) = \frac{\tilde{c}_k(s, t)^{\gamma_{post}}}{\sum_{k'=1}^{n} \tilde{c}_{k'}(s, t)^{\gamma_{post}}},
\]
Disparity Transfer to the Saliency Map

Once we have the DCV, we directly construct our saliency layered model by multiplying each layer of the DCV by the pre-estimated saliency map $\Psi_0$ of the central view:

$$s_k = c_k \cdot \Psi_0.$$  \hspace{1cm} (5.6)

Therefore, thanks to the normalisation of the DCV in Eq. (5.5), simply summing all the saliency layers reconstructs exactly the original saliency map $\Psi_0$. This is analogous to the FDL rendering in [56], where the central view is obtained by summing the FDL layers. More generally and similar to [56], the saliency map of any view $(u_j, v_j)$ can be rendered by shifting each saliency layer $s_k$ by the vector $d_k \cdot (u_j, v_j)$, and by summing the layers. This directly shifts the detected salient regions according to their disparity in the LF, hence applying the disparity transfer.

We present in the next section how to generalise the rendering of the saliency layers to generate saliency maps of refocused images with out-of-focus blur.

5.5 Refocused Rendering from Saliency Layers

While applying the FDL rendering equations in [56] to our saliency layers provides the intended saliency maps for all-in-focus views, this is not suitable for predicting the visual attention of refocused images. In this case, the FDL rendering equations simulate defocus blur by blurring each layer $k$ proportionally to the distance $|d_k - d_{\text{focus}}|$ between the disparity $d_k$ of the layer and that of the refocused plane $d_{\text{focus}}$. However, the expected effect of defocus blur on visual attention is not a blurred saliency map, but rather a strong decrease in saliency of the out-of-focus regions relative to the in-focus regions. This is intuitive since blurred areas are typically less salient, and that was further analysed in Chapter 3 in the context of refocusing. Hence, we propose the following rendering equation for generating the saliency map of a refocused image at disparity $d_{\text{focus}}$:

$$\tilde{\Psi}_{d_{\text{focus}}} = \sum_{k=1}^{n} s_k \cdot e^{-\frac{(d_k - d_{\text{focus}})^2}{2\sigma_{\text{disp}}^2}},$$  \hspace{1cm} (5.7)

$$\Psi_{d_{\text{focus}}} = \tilde{\Psi}_{d_{\text{focus}}} / \max(\tilde{\Psi}_{d_{\text{focus}}})$$  \hspace{1cm} (5.8)

where the standard deviation $\sigma_{\text{disp}}$ of the Gaussian weight can be adjusted to control how the refocusing decreases the probability of detection of the out-of-focus regions by the human eye. The final normalisation step in Eq. (5.8) ensures that the saliency map has values in the range $[0, 1]$.

The parameter $\sigma_{\text{disp}}$ may be chosen depending on the camera aperture of the refocused
image. For instance, a large aperture results in strong defocus blur. Thus, a small value of $\sigma_{disp}$ should be chosen so that only the regions very close to the focus plane keep a high value in the saliency map. On the other hand, using an arbitrarily small aperture captures an all-in-focus image. In this case, $\sigma_{disp}$ may be chosen very large, which recovers the original saliency map $\Psi_0$ of the centre SAI.

It should also be noted that the disparity of the focus plane $d_{focus}$ does not need to be equal to any of the disparities $d_k$. Therefore, the discrete set of saliency layers allows VA prediction in the continuous range of refocused images.

5.6 Experimental Results

To measure the performance of our model we run and evaluate it on an LF VA database and compare our results to that of the FGSE method. In this section, we will describe this database, and present an analysis of the numerical and visual results.

5.6.1 Database

To evaluate the proposed FDL-based VA estimation method, we used the ground truth VA data collected in our earlier work in Chapter 3. Similar to our earlier work on LF saliency estimation in Chapter 4, we use the eye-fixation data which was collected for the “Region-1” and “Region-2” focal renderings of 17 LFs.

5.6.2 Quantitative Analysis

Table 5.1: Metric results of our FDLSE model with parameters $\sigma_{disp} = 0.4$, $\gamma_{post} = 0.4$, $\gamma_{pre} = 10$ and the FGSE model with parameter $\sigma_D = 0.4$ evaluated against ground truth.

<table>
<thead>
<tr>
<th>Saliency method</th>
<th>AUC↑</th>
<th>NSS↑</th>
<th>CC↑</th>
<th>KLD↓</th>
<th>SIM↑</th>
</tr>
</thead>
<tbody>
<tr>
<td>FGSE</td>
<td>0.845</td>
<td>1.618</td>
<td>0.680</td>
<td>0.619</td>
<td>0.640</td>
</tr>
<tr>
<td>FDLSE</td>
<td>0.840</td>
<td>1.672</td>
<td>0.697</td>
<td>0.698</td>
<td>0.650</td>
</tr>
</tbody>
</table>

In Table 5.1 we evaluate the performance of our FDLSE method comparing it to that of the FGSE method over different metrics. For both, parameters were chosen which give the best quantitative results for fair comparison. These were found to be $\sigma_{disp} = 0.4$, $\gamma_{post} = 0.4$ and $\gamma_{pre} = 10$ for FDLSE. We can use the same parameter $\sigma_{disp}$ for all experiments because all the refocused images in the database were rendered with a full aperture (i.e. the full range of views available in the input LF were used). For the Gaussian mask we apply to the DCV, we use the same Gaussian parameters as was used to generate the ground truth VA maps in the database.
These parameters $\sigma_s = \frac{47.66}{1080} W$ and $\sigma_t = \frac{42.67}{1080} H$ are adjusted to the height $H$ and width $W$ of our views.

For the distribution based metrics, FDLSE has better scores for the similarity metrics CC and SIM but worse for the dissimilarity metric KLD. Higher CC signifies a greater linear correlation between our saliency map and the ground truth VA map. Higher SIM means that there is more overlap between our saliency map and the ground truth VA map when viewing the distributions as histograms. The worse KLD score suggests there is more information lost for FDLSE compared to FGSE. This could be due to the distribution of our maps being overall less spread. However, the greater concentration of the saliency of our output maps leads to improvements in the NSS score which indicates that our model has fewer low valued false positives.

Overall our FDLSE method performs at least as good as FGSE with better scores for three out of the five metrics. The statistical significance of the results was tested by a non-parametric test, as the distribution of the scores was found to be non-normal after a Kolmogorov Smirnov test. After a Wilcoxon signed-rank test, it was found that the differences between FDLSE and FGSE are not statistically significant. As our proposed method requires only the centre SAI saliency map to be estimated by DeepGaze, it has the advantage of reduced computational complexity. Since estimating a saliency map using DeepGaze takes 9s and for FGSE this needed to be done for all SAIs (ranging from 81 to 289 in the LF VA database) per LF this is a significant reduction.

5.6.3 Qualitative Analysis

The visual results of our method are demonstrated for a variety of cases in Fig. 5.3. In this section, we analyse the observable differences between our models output and the previous FGSE method and compare both to the ground truth fixations and ground truth VA maps. For each LF, two refocused renderings were used as stimuli to evaluate the proposed model, namely, “Region-1” and “Region-2”. We chose these images as there is adequate distance between their focal planes such that there is a noticeable visual difference between them when refocusing.

The proposed method better predicts salient ‘in-focus’ regions compared to the FGSE method. Even when salient regions are not present in the centre SAI Saliency map, as observed in the Tarot-Small Region-2 output Fig. 5.3(a), FDLSE predicts the bottom salient region to some degree. However, in the FGSE saliency map this region is absent. This indicates that FDLSE better estimates disparity and more appropriately assigns weight to salient regions at the focal plane. Similarly, when comparing the outputs for the Dishes Region-2 LF rendering to the GT, our FDLSE model correctly predicts high saliency of the centre region whereas the FGSE incorrectly estimates a top right region as being the more salient.

When regions of focus guide attention, FDLSE more accurately computes the amount by
Figure 5.3: The results of our FDLSE model shown alongside the stimulus, the ground truth fixations, the ground truth VA map, the centre SAI saliency map and the FGSE method for comparison. Here the stimuli are the Region-1 and Region-2 renderings of a selection of the light fields tested.
which the centre SAI’s predicted saliency should influence the saliency map of the refocus rendering. This can be observed by comparing the output of the FDLSE and FGSE methods to the GT VA for the Medieval LF in Fig. 5.3(b) and (c). Furthermore, the saliency estimation of the proposed model is often more precise than FGSE. This can be seen, for example, when observing the Tower Region-2 and Treasure Region-1 outputs in Fig. 5.3(e) and (f) and corresponding ground truth.

FDLSE does not rely on disparity maps of equidistant discretisation as in FGSE for its disparity weighting. Instead, it uses the LF sub-aperture images to estimate disparity according to each LF’s scene’s unique depth distribution. This means that our method better estimates the depth planes of the scene’s content and at smaller distances apart compared to the FGSE method. Since it more accurately locates the focal plane to predict focus guided salient regions, we tuned FDLSE to more precisely weight the saliency of ’in-focus’ regions. This is why there is less spread in the output of our FDLSE model compared to FGSE.

There is a trade-off between having a higher concentration of saliency but the possibility of under detecting salient regions versus a wider distribution of the saliency but the potential of estimating many non-salient regions. To observe this trade-off, we compare the outputs of the two models to the GT fixations for the Medieval LF in Fig. 5.3(b) and (c). It is clear that the FGSE output takes into account more of the fixation points whereas FDLSE better weights the importance of concentrated groupings of fixations. Depending on the application, higher or lower concentration of the saliency of the output is needed. As the proposed FDLSE model has more parameters that can be adjusted, compared to the FGSE method, we can better tune the model to produce an output that suits any application.

5.6.4 Computation Time Analysis

In Table 5.2 we list the time it takes to compute a saliency map using our FDLSE model from a region-in-focus rendering (single image), for each light field.

5.7 Conclusion

In this work, we have developed a novel representation of LF VA through a model we have called FDLSE, which builds on the existing FDL LF representation. It consists of saliency layers that hold all the salient information of the LF, each corresponding to a depth plane parameterised by a disparity value. This is a more unified representation of VA in four-dimensions compared to our previous FGSE model, which represented the LF VA as an array of SAI saliency maps. To validate our FDLSE model, we used these layers to predict the saliency of LF refocus renderings and compared them to the output of FGSE.
Table 5.2: Computation times for the FDLSE model with parameters $\sigma_{\text{disp}} = 0.4, \gamma_{\text{post}} = 0.4, \gamma_{\text{pre}} = 10$.

<table>
<thead>
<tr>
<th>Light Field</th>
<th>Computation time (in seconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Treasure</td>
<td>183.46</td>
</tr>
<tr>
<td>Lego Knights</td>
<td>88.33</td>
</tr>
<tr>
<td>Tarot-Small</td>
<td>87.10</td>
</tr>
<tr>
<td>Tarot-Large</td>
<td>63.43</td>
</tr>
<tr>
<td>Vespa</td>
<td>13.53</td>
</tr>
<tr>
<td>Friends</td>
<td>9.23</td>
</tr>
<tr>
<td>Pens</td>
<td>12.44</td>
</tr>
<tr>
<td>Dishes</td>
<td>15.20</td>
</tr>
<tr>
<td>Sideboard</td>
<td>11.52</td>
</tr>
<tr>
<td>Medieval</td>
<td>11.77</td>
</tr>
<tr>
<td>Boardgames</td>
<td>11.35</td>
</tr>
<tr>
<td>Dino</td>
<td>13.77</td>
</tr>
<tr>
<td>Platonic</td>
<td>11.73</td>
</tr>
<tr>
<td>Table</td>
<td>5.89</td>
</tr>
<tr>
<td>Tower</td>
<td>16.29</td>
</tr>
<tr>
<td>Town</td>
<td>10.75</td>
</tr>
<tr>
<td>Vinyl</td>
<td>5.66</td>
</tr>
</tbody>
</table>

Compared to our previous work, the disparity levels of our FDLSE method were calibrated to better match the LF scene’s content. Additionally, FDLSE has more parameters available to fine tune the output. Thus, we were able to weight more precisely the saliency of focal plane regions for this model. We evaluated our FDLSE model for various metrics and found that it predicts the saliency of “in-focus” regions more accurately than our previous FGSE method.

Furthermore, FDLSE is optimised in three ways. It is only necessary to estimate the centre view saliency map using DeepGaze II to build the saliency field, in contrast to our previous FGSE method which required the estimation of all views. FDLSE can utilise GPU parallelisation as operations are carried out in the Fourier domain. Finally, disparity estimation is an intrinsic part of the model and does not need to be pre-computed, saving computational resources.

We intend to expand this work to generate saliency maps of virtual views and predict most salient renderings. We also plan to explore whether the LF saliency field can be used as input to LF rendering algorithms and aid the selection of LF rendering parameters. For applications involving LF rendering, this would give a grounded rationale for choosing rendering parameters and could be used to guide the viewer to specific regions of an LF scene.
Chapter 6

Optimal Focal Plane Selection: A Downstream Application Using Light Field Saliency Estimation

In previous chapters we have shown that it is possible to generate a light field saliency field which can be used to predict the saliency map of any light field rendering. So now the question that remains is “How can this be applied to real-world problems in the light field domain?”. In this chapter we explore one possible use case of our light field saliency field: as an automated ranking mechanism for selecting refocus renderings for use in gaze contingent blur of light field data.

To solve the vergence-accommodation problem in head-mounted displays, gaze contingent blur can be used to provide more naturalistic display conditions. When displaying images to a viewer, gaze contingent blur involves tracking a user’s eyes and using the pixel coordinates corresponding to the gaze location to determine the focal plane to be rendered. The image shown to the viewer is thus rendered in-focus at the plane of fixation and out-of-focus everywhere else. This can be done by using a stack of pre-rendered images focused at a sequence of depths. Each possible gaze coordinate is mapped to a disparity which corresponds to a particular slice in the pre-rendered image stack.

However, choosing how many and which rendered images to use in the pre-rendered image stack is not a trivial problem. Since even small spatial distances between gaze coordinates will change the focal plane and thus require a different image to be rendered, this also means more switching between planes. A consequence of this is an increase in lag, especially the further the distance between consecutive fixations. As using fewer images in the pre-rendered image stack decreases the chances of lag and requires less computational resources for rendering, in this chapter we strive to select fewer images while still capturing as many of the regions of interest
of the scene as possible. We do this by predicting a saliency field for the visual scene we want to display and ranking focal slices from this field according to the sum of the saliency of each slice. Based on this ranking, we use the cumulative distribution function to select the optimal slices for the application.

6.1 Introduction

When viewing an object in the real world, our eyes converge and accommodate at the same distance (an in-depth overview of these terms is given below in Section 6.2.2). This is not the case with stereoscopic displays which are typically worn as a head-mounted display (HMD). Inherent in their stereoscopic design, different images are shown to the left eye and right eye which creates binocular disparity. The two eyes converge to the simulated distance of the object they view. Thus, various virtual objects appear to be at differing depths, giving the impression they are three-dimensional. However, in the design of traditional HMDs, images shown have all objects in-focus at once. This causes the eyes to accommodate to a fixed screen distance regardless of the object they view. Thus, resulting in conflicting visual cues being sent to the viewer’s brain which can cause side-effects such as visual fatigue, eye irritation, headache and nausea and has been demonstrated experimentally [132, 133]

Techniques have been developed in multi-view display design in recent years which aim to reduce visual discomfort caused by the vergence-accommodation problem outlined above. However, some involving eye-tracking, defocus blurring and light field rendering are not computationally efficient enough to perform this task in real-time.

Duchowski et al. [134], for example, studied the effect of gaze-contingent depth-of-field on visual discomfort. They found that errors in the eye-tracking and a temporal lag resulted in defocus blurring and binocular gaze becoming unsynchronised. They found that users disliked this effect but that improvements in synchronisation would be worth further investigation. We hypothesize that choosing a small number of planes which are optimised according to likelihood of accommodation will improve this synchronisation by reducing temporal lag as there are fewer renderings to jump between.

In this chapter, we investigate how our light field saliency field can be used to choose optimal planes for adaptive depth-of-field rendering.
6.2 Background and Related Work

6.2.1 Use of Defocus Blur in Multi-View Displays

Defocus blur has been shown to be a more precise cue for depth than binocular disparity [135]. It is defined as the circle \( C \) over which a point \( Z_1 \) is imaged at the retina when the lens is focused at a depth of \( Z_0 \). \[ C = AS\left| \frac{1}{Z_0} - \frac{1}{Z_1} \right| \] where \( A \) is the diameter of the pupil and \( S \) is the distance between lens and the retina. \( C \) is also known as the circle of confusion. It has been shown that retinal blur drives accommodation and that its presence has been effective in reducing visual fatigue when viewing stereoscopic stimuli [132, 133]. This has led to research into the design of HMDs that stimulate accommodative responses by rendering objects to a viewer with increasing blur with spatial distance from the plane of eye fixation [136, 137].

6.2.2 Vergence-Accommodation Conflict

The “vergence-accommodation” conflict is a well-known problem when using HMDs [60]. Vergence is the rotation of the eyes in opposite directions to align the eyes and obtain a single fused image of a fixated object. Accommodation is the adjustment of the power of the eye’s lens to obtain a single sharp image focused on the retina [133]. Retinal disparity is the visual cue that drives vergence whereas retinal blur drives accommodation [134]. Both are integral to a comfortable viewing experience of a 3D scene, as inaccurate vergence results in double vision and inaccurate accommodation causes blurred vision.

6.2.3 Eye-Tracking and Dynamic Rendering for Virtual Reality

Here we will outline new approaches aiming to tackle the vergence-accommodation conflict in virtual reality (VR) displays such as head-mounted displays and near-eye displays. These approaches integrate technologies such as focus-tunable optics and eye-tracking to simulate retinal blur and drive accommodation. Conventional near-eye displays employ simple magnifiers to enlarge the image of a micro-display, creating a virtual image at a fixed distance from a viewer [138]. These new designs enable a user to refocus at different regions of a virtual scene depending on their gaze, which will allow their eye to accommodate to various distances.

- **Adaptive Depth-of-Field Rendering** is a software only approach that renders the fixated object sharply while blurring other objects according to relative distance [138]. Combined with eye-tracking this is known as *gaze contingent retinal blur*. These methods display an image that is in-focus at the viewer’s gaze location while simultaneously blurring other regions. A different image is presented for each different depth of objects within a scene. These images can be rendered in advance or in real-time. They can be obtained either...
Figure 6.1: Demonstration of the vergence-accommodation conflict for near and far scenarios. This occurs due to a user’s eyes verging at the stereoscopic distance and accommodating at the virtual image distance [20].

Figure 6.2: Illustration of vergence and accommodation distances matched by adjusting the virtual image distance [20] by using a 3D model or photographically from a real scene e.g. through a sequence of exposures adjusting the lens for each capture or by capturing a light field and generating a focal stack.
Mauderer et al. [136] investigated gaze contingent retinal blur using a depth-of-field algorithm on pre-rendered images. They derived a function from the depth map of the scene which maps each possible gaze point to a plane of focus. The inherent delay of changes in accommodation was simulated by using a transition function which gradually alters the focus. A change of focus was initiated only when the distance between current gaze region and the next went beyond a certain threshold. As discussed in the introduction, there are perceptual issues with the use of these types of software: rendering the incorrect focal plane results in blur at regions within the user’s gaze, slow rendering speeds means that the correct images are not rendered dynamically synchronised with the user’s gaze [134].

- **Adaptive Focus Display** is a combined software/hardware approach which involves changing either the focal length of the lenses or the distance between the micro-display and the lenses. Combined with eye-tracking this is known as *gaze contingent focus*.

![Figure 6.3: Gaze-contingent focus](image)

Padmanaban et al. [137] built a display with focus-tunable lenses i.e. the focal length of the lens between the eyes and the micro-displays is programmable. It allows for dynamic control of the distance to the virtual image, independently for each eye. It has a built in autorefractor which measures viewers accommodation while displaying content and so the system can dynamically adjust to the users stereoscopic/ inter-pupillary distance and update the virtual distance of the target stimuli accordingly. This dynamic stereoscopic distance matching display type is known as a *varifocal display*. This device is similar to that in Fig. 6.4 except an autorefractor is used instead of an eye-tracker to update the distance of the virtual image $v(e)$.

They also designed a *gaze-contingent near-eye display system* [20]. This device is a varifocal display built on the Samsung GearVR platform. It includes a stereoscopic eye-
Figure 6.4: Varifocal display with an eye-tracker [21], where $v(e)$ is the virtual image distance and $f(e)$ is the focal length of the lens tracker and a motor that can be used to adjust the distance between the screen and magnifying lenses in real time. Using eye-tracking, the depth corresponding to the object within viewers gaze is calculated and the magnified virtual image is adjusted accordingly. This is a dynamic process which simulates retinal blur. This device is similar to that in Fig. 6.4 except, instead of adjusting the focal length of the lens, the display is moved. Konrad et al. [22] designed a prototype accommodation-invariant near eye display where the focal length of the focus-tunable lens was programmed to periodically perform a focus sweep i.e. move a virtual image throughout the accommodation range of the viewer.

Figure 6.5: Accommodation-invariant near eye display [22]

To prevent artifacts, the sweeping time of the lens is faster than the accommodation sys-
tem. This produces a visual stimulus that is invariant to the accommodation state of the eye. The idea behind this is that, since retinal blur is rendered by the display in an open loop it becomes optically disabled. This potentially allows for accommodation to remain coupled with the vergence distance of the eyes and therefore, to be driven by disparity (as opposed to retinal blur). Displays which utilise this focus sweep strategy to solve the vergence-accommodation conflict are known as *multi-focal displays*. In this device, the continuous focal-sweep is a sum of individual planes, each focused at different distances. Compared to a conventional near-eye display, the invariance causes a decrease in spatial resolution. They hypothesised that a multi-plane mode which allows fewer focal planes could improve the image resolution. Since accommodation is an imprecise mechanism, they also postulate that people are unlikely to accommodate between planes and so fewer planes would be sufficient. For this multi-plane mode implementation, they found that image resolution improved significantly at the respective focal planes but this mode worsened inter-plane quality.

Similarly, Chang et al. [21] introduced another approach to the focus sweep strategy by designing a *multi-focal display with dense focal stacks*. This device displays focal planes from a dense focal plane stack according to the tracked focal length of an oscillating focus-tunable lens. Similar to the above method, this oscillation among the focal planes optically disables retinal blur.

![Multi-focal display with dense focal stacks](image)

Figure 6.6: Multi-focal display with dense focal stacks [21]

We have outlined some challenges for both eye-tracking and focus sweep solutions to the vergence-accommodation problem in head-mounted and near eye displays. The former solution
was limited by slow rendering speed and the latter forced to balance a trade-off between image sharpness and accommodation invariance along the entire accommodation range.

As adaptive focus displays are highly specialised devices, with the three discussed above only in the prototyping stage, in this chapter we devised our methodology for a software based approach only. However, we anticipate that our automatic ranking mechanism could be modified for use in adaptive focus displays if they become more widely available in the future.

Furthermore, as head-mounted devices are more prolific than near-eye displays, we developed our technology with them in mind, however, the methodology could reasonably be similarly applied to both.

Lastly, the literature on multi-focal displays with dense focal stacks [21] has compared different numbers of equidistant planes and concluded that fewer planes results in worse perceptual performance. As these devices display a lower resolution stimulus at inter-plane regions, this worsened performance could be due to them not calibrating the planes to salient regions. Thus, we would expect that selecting few but optimal planes for multi-focal displays based on saliency maps could remedy this. However, since we are unable to access these specialised multi-focal displays, we cannot verify this hypothesis. Therefore, in this chapter, we do not concentrate on this application.

Pertaining to the points made above, in this chapter, we will mainly be discussing how to improve the rendering speed and focal plane accuracy in adaptive depth of field rendering software, for the purpose of gaze contingent retinal blur in head-mounted displays, via the automatic selection of optimal focal planes.

### 6.3 Method Overview

Using our light field saliency estimation model from Chapter 5, we designed a new method of selecting optimal light field renderings for the purpose of synchronising renderings with the viewer’s gaze, and thus avoiding lag and spurious jumps in depth. Our objective was to select the focal slices of the light field (i.e. refocused images), that best represent the salient content in the scene. Each focal slice appears in-focus at a given depth plane that can be parameterised by a disparity value. Hence, selecting the most salient focal slices is equivalent to finding the corresponding set of disparity values.

The method is illustrated in Fig. 6.7 and consists of four main steps:

1. Using our light field saliency model FDLSE, we estimate the saliency maps of focal slices of a given light field.

2. From these saliency maps, we then determine a probability distribution function (p.d.f.) \( p \) that represents the probability of looking at a given depth in the scene (corresponding
Figure 6.7: Overview diagram of the proposed method for selecting optimal focal slices for a downstream gaze contingent blur application, demonstrated for the Dino light field to a given disparity).

3. The p.d.f. \( p \) is then used to sample disparity values according to their probability of being associated with salient regions in the scene.

4. Once the optimal disparity values have been calculated, a focal stack of images can be pre-rendered for display applications.

Mathematically, the p.d.f. in the disparity dimension can be defined as:

\[
p(d) = \frac{\sum_{i,j} S(d)_{i,j}}{\int_{-\infty}^{+\infty} \sum_{i,j} S(t)_{i,j} dt},
\]

where \( S(d)_{i,j} \) is the pixel \((i, j)\) in the saliency map estimated for the focal slice of disparity \( d \). From this definition, the disparities associated to the most salient focal slices will have high probability values. Thanks to our light field saliency model, we can compute \( S(d) \) directly, without the need to render the corresponding focal slices. For practical computations, we need to discretise the p.d.f. This is performed by taking \( N = 120 \) uniformly spaced disparity values.
$d_k$ in the range $[d_{\text{min}}, d_{\text{max}}]$ where $d_{\text{min}}$ and $d_{\text{max}}$ are respectively the minimum and maximum disparity values in the scene. The discrete probability values are then computed as:

$$p_k = \frac{\sum_{i,j} S(d_k)_{i,j}}{\sum_i \sum_{i,j} S(d)_{i,j}} \tag{6.2}$$

Given the discretised p.d.f., we want to sample disparity values covering the most salient slices of the light field. The method follows the conventional approach for sampling random numbers from scalar random variables with arbitrary distributions: random numbers following a distribution $p$ can be sampled by first generating random numbers with uniform distribution in the range $[0, 1]$ and by transforming these numbers with the inverse of the cumulative distribution function (c.d.f.), commonly known as inverse transform sampling. The c.d.f. is defined as follows.

$$c(x) = \int_{-\infty}^{x} p(t)dt = P[X \leq x]. \tag{6.3}$$

To aid visualisation, both the p.d.f. and c.d.f. of the Dino light field are plotted in Fig. 6.8 below.

![Figure 6.8: Probability density function and cumulative distribution function for the Dino light field](image)

The challenge is to choose disparity values which partition the p.d.f into quantiles containing a certain proportion of the probability, illustrated in Fig. 6.7. Hence, to generate these
quantiles we use equidistant values in the range \([0, 1]\) instead of uniformly distributed random values, and we apply the inverse c.d.f., which produces a deterministic sampling that follows our distribution. This process is illustrated in Fig. 6.7 for the Dino light field and in Fig. 6.9 for six other light fields. Here, \(n = 6\) slices are selected: 6 equidistant values are sampled on the vertical axis in the plots and are transformed using the inverse cumulative distribution function (red dashed line) to compute the 6 disparity samples (red bars on the horizontal axis).

Note that in practice, since the p.d.f. is discretised, numerical computations are necessary to obtain the c.d.f. and its inverse. Following the trapezoid method, we compute the c.d.f. and its inverse from a continuous p.d.f. function that linearly interpolates between the discrete p.d.f. values \(p_k\) as:

\[
p(d) = \begin{cases} 
0 & \text{if } d < d_0, \\
p_k(d_{k+1}-d)+p_{k+1}(d-d_k) & \text{if } d_k \leq d < d_{k+1}, \\
1 & \text{if } d \geq d_N, 
\end{cases}
\]  

(6.4)

where we define \(p_0 = p(d_0) = 0\).

Once the disparity values have been selected, the corresponding images can be pre-rendered in a focal stack and used for rendering applications.

Figure 6.9: Plots for six light field examples showing how the cumulative distribution function (c.d.f) is used to select focal stack slices according to saliency. The probability density function (p.d.f) (indicated in blue) is calculated from the sum of the values of each saliency map slice, of the light field saliency field, along the depth dimension. A dotted red line indicates the c.d.f (which is calculated from the p.d.f) and the solid red lines indicate where slices are selected from (equal distances along the c.d.f).
6.4 Experimental Results

To assess the value of our method, we compare its efficacy in selecting slices from visually important regions to simply sampling slices at equal distances along the depth (focus) dimension of the light field. We test the method for $n = 6$ slices as this presents a challenge (relative to the depth of light fields in our dataset) in balancing sampling along the full depth of the light field and sampling more from regions of visual importance. Some examples of the selected slices and equidistant slices are shown in Fig. 6.10, Fig. 6.11 and Fig. 6.12 for side-by-side comparison. To validate our method, we generate two “combined slices” images, one using the six selected slices and another using the six equidistant slices as shown in Fig. 6.13. To generate these images, we use the FDL function which constructs the central view image from a focal stack but with only 6 samples as input. As the six samples of the focal stacks of our light fields are a sparse sampling, the regions of the output images which correspond to the missing focal planes appear out-of-focus or blurred. The image with less blur in salient regions indicates a better sampling. An analysis of the visual and numerical results is provided below.

6.4.1 Qualitative Results

The centre SAI, combined slices images and centre SAI saliency map for the two samplings are shown in Fig. 6.13 and Fig. 6.14 presents the focus depth maps of the light fields. In our method the slices are more dense at focal planes which contain highly salient regions or objects. This is evident from observing the focus depth maps of the light fields in Fig. 6.14. In most cases, fewer slices are allocated to the background for the selected slices compared to the equidistant slices. For the Lego Knights light field, the focus depth maps in Fig. 6.14 (a) show that more of the slices are chosen at focal planes containing the knights which is as we expect. This is backed up from observing the selected slices of Lego Knights in Fig. 6.10 as four of the selected slices’ planes are sampled at the knights compared to three using the equidistant method.

This can also be observed for the Platonic light field in Fig. 6.14 (b) where the focal planes are concentrated at the areas where the large objects occupy in the scene. However, a disadvantage of this method is that the background may not be selected optimally and so appears blurred, as seen in the Platonic light field Fig. 6.13 (b) and in the Pens light field Fig. 6.13 (e). The equidistant slices often take a larger area of the depth plane into account, thus, the background is typically better captured.

As more slices are selected at highly salient objects, the details of these objects are more likely to occupy a slice compared to the equidistant sampling. This is clear in the depth focus map of the Treasure Light field in Fig. 6.14 (d), on observation of the chains on the chest. They appear sharper in both the combined slices image and the selected slices images compared to
Figure 6.10: Side-by-side comparison, for the Lego Knights and Platonic light fields, of the six focal stack slices selected using our method to those sampled at equal distances along the depth dimension of the light field.
Figure 6.11: Side-by-side comparison, for the Table and Treasure light fields, of the six focal stack slices selected using our method to those sampled at equal distances along the depth dimension of the light field.
Figure 6.12: Side-by-side comparison, for the Pens and Dishes light fields, of the six focal stack slices selected using our method to those sampled at equal distances along the depth dimension of the light field.
Figure 6.13: Side-by-side comparison of the ‘combined images’ generated from slices selected by our method to those generated from equidistant slices shown alongside images indicating the error for both compared to the original centre SAI and lastly the centre SAI saliency map generated by DeepGaze II.
Figure 6.14: The focus depth map generated for the six slices selected using our method to those generated for the six equidistant slices, for each light field. They indicate which slice the pixels of the combined images map to.
those of the equidistant slices.

In the equidistant slices, the first and last slices are often wasted as they are usually sharp at extreme foreground or background regions that are not important and blurred at important regions. For example, comparing the first and last slices between the two sampling types for the Table light field in Fig. 6.11, the slices of the equidistant selection are overall more blurred.

A disadvantage of this method is that some salient regions of the light field with a relatively low saliency compared to the most salient regions do not get as many slices dedicated to them. They are instead dedicated to the more salient region(s). For example, the “Kiehl’s” writing in the background of the Dishes light field in Fig. 6.13(f) only gets one slice where it is not fully blurred. Yet it is still partially blurred as the focal plane is effectively pulled close to the large salient area in the centre of the scene.

### 6.4.2 Quantitative Results

To compare the combined slices images generated from selected slices to those generated from equidistant slices, we use a saliency weighted Peak Signal-to-Noise Ratio (PSNR) metric we call Sal\_PSNR. The PSNR metric is often used to quantify image reconstruction quality for compressed images. For our case, PSNR gives a measure of how well selected slices reconstruct the centre SAI compared to the equidistant slices. Based on the assumption that quality is more important in more salient regions, we weight PSNR with saliency maps. Sal\_PSNR is calculated according to the following equations:

\[
Sal\_MSE = \frac{1}{qmn} \sum_{c=0}^{k-1} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} S(i,j) \cdot (I(c,i,j) - K(c,i,j))^2 \\
\tag{6.5}
\]

\[
Sal\_PSNR = 10 \cdot \log_{10}(\frac{\text{max}^2}{Sal\_MSE}) \\
\tag{6.6}
\]

These equations first calculate the saliency weighted mean squared error Sal\_MSE between \( m \times n \) images \( I \) and \( K \) with \( q \) colour channels. This is the sum, over all colour channels and pixels \((c,i,j)\), of the squared per pixel difference between the centre SAI image \( I \) and a combined slices image \( K \) multiplied by the DeepGaze II estimated centre SAI saliency map \( S \). Secondly the Sal\_PSNR is the regular PSNR function with Sal\_MSE used as input and \( \text{max} \) set to the highest possible pixel intensity of 255.

The Table 6.1 displays the Sal\_PSNR results for 17 light fields. The selected slices sampling has higher (better) scores, indicated in bold, for 14 out of the 17 light fields. The differences are very small between the results for two of these lower Sal\_PSNR light fields Vespa and Boardgames. However, for the Pens light field, the difference is greater. One possible reason for this is that the background of the selected slices combined slices image has lots of detail.
Table 6.1: Comparison of the saliency weighted PSNR results of two combined slices images, one generated from equidistant slices of the focal stack and the other from our method’s selected slices

<table>
<thead>
<tr>
<th>LF</th>
<th>Sal_PSNR Equidistant</th>
<th>Sal_PSNR Selected Slices</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boardgames</td>
<td>36.83</td>
<td>36.82</td>
</tr>
<tr>
<td>Dino</td>
<td>47.77</td>
<td>49.13</td>
</tr>
<tr>
<td>Dishes</td>
<td>47.47</td>
<td>48.85</td>
</tr>
<tr>
<td>Friends</td>
<td>46.54</td>
<td>46.57</td>
</tr>
<tr>
<td>Lego Knights</td>
<td>36.99</td>
<td>39.08</td>
</tr>
<tr>
<td>Medieval</td>
<td>44.53</td>
<td>45.61</td>
</tr>
<tr>
<td>Pens</td>
<td>51.01</td>
<td>47.93</td>
</tr>
<tr>
<td>Platonic</td>
<td>51.21</td>
<td>51.37</td>
</tr>
<tr>
<td>Sideboard</td>
<td>40.33</td>
<td>41.99</td>
</tr>
<tr>
<td>Table</td>
<td>37.49</td>
<td>37.75</td>
</tr>
<tr>
<td>Tarot_Large</td>
<td>26.90</td>
<td>26.94</td>
</tr>
<tr>
<td>Tarot_Small</td>
<td>38.48</td>
<td>38.99</td>
</tr>
<tr>
<td>Tower</td>
<td>33.80</td>
<td>33.99</td>
</tr>
<tr>
<td>Town</td>
<td>47.57</td>
<td>49.28</td>
</tr>
<tr>
<td>Treasure</td>
<td>33.61</td>
<td>35.01</td>
</tr>
<tr>
<td>Vespa</td>
<td><strong>43.85</strong></td>
<td>43.71</td>
</tr>
<tr>
<td>Vinyl</td>
<td>36.03</td>
<td><strong>36.16</strong></td>
</tr>
</tbody>
</table>

but is much more blurred than the equidistant one and so it might result in a lower Sal_PSNR.

6.5 Conclusion

In this chapter, we developed a novel method for selecting refocus renderings for use in gaze contingent retinal blur software in head-mounted displays. This method was derived from the saliency maps which are outputted by the FDLSE model we created in Chapter 5. We surmised, based on the conclusions of other works, that having a small number of focal planes (renderings) to switch between in gaze contingent adaptive depth-of-field technology would alleviate lag and other problems caused by storage and speed limitations. Since fewer focal planes means each one represents a greater area of the depth dimension and thus where the viewer might look, it is feasible that sparser renderings leads to less switching between them, improving the users experience.

Through experimental analysis, we demonstrated that the slices selected by this method achieve better sharpness at regions users are likely to view/fixate overall compared to choosing slices at equal distance along the depth dimension. This indicates that our method can be used to make better choices of renderings of light fields for gaze contingent adaptive depth-of-field software. This work serves as proof that the light field saliency field can be used as an automatic
ranking mechanism for choosing slices of the light field focal stack according to where people are most likely to look.

We would expect future work to test this application on subjects in an experimental setting using a HMD with a built-in eye-tracker such as the FOVE [139] using stereoscopic images generated for each slice and a stereoscopic focus depth map. For this application in HMD adaptive depth-of-field software, the ranking was optimised to yield more slices from highly salient regions of the scene while also ensuring that the slices are spread out along the depth dimension. However, we anticipate that this ranking capability of the saliency field can be exploited in alternative ways to advance areas such as quality assessment, compression for streaming or rendering speed for multi-view display technology.
Chapter 7

Conclusion

At the beginning of this work, we identified that visual attention in relation to light fields was a sub-field of saliency which to our knowledge had no existing research conducted, later corroborated by Fu et al. [29]. We identified that a useful end goal to target would be to develop a model which can predict visual attention for any light field rendering. This was formalised in the research questions presented in Section 1.2.

We hypothesised that a dataset could be collected consisting of accurate eye fixation data, which could be used for future research related to visual attention for light fields. Further, we anticipated that with such a dataset, a visual attention prediction model could be created and evaluated, setting a benchmark for future research. This thesis outlines the details of the work undertaken to investigate this, and we conclude our findings below.

7.1 Summary

We began by designing an experiment which involved using a state of the art Eyelink-1000 eye tracker to record temporal eye fixation data for five focally varying renderings of 20 light fields. We view this dataset as relatively comprehensive, since the light fields were acquired using the most ubiquitous techniques: camera arrays, cameras with microlens arrays, and computer generated imagery. The scenes used were diverse and challenging for saliency, containing a mixture of objects and shapes with a variety of colour contrast and textures, which occupy different positions in the frame and along the depth plane. The experiment involved 21 participants and we have found that it has served as a valuable resource for our own research. We strongly anticipate that the research community will benefit greatly from this database and that it will facilitate subsequent research into light field saliency. The methodology behind the experiment will also serve as a useful foundation for any related data collection that will be carried out in the future.
Once we had processed the data from our subjective experiment, we performed an analysis of the data to understand how the focus cue, which is a characteristic cue of light fields, affects visual attention in relation to other cues such as centre bias, contrast, etc. The details of our results can be found in Chapter 3, but the main findings were that the focus cue has a non-trivial impact on visual attention, the variation in visual attention between the different rendering types was measurably significant and participants’ fixations were substantially more dispersed for all-in-focus renderings compared to the others. From these final two points, we deduced that a visual attention model aiming to estimate the saliency map of only the 2D all-in-focus image of the light field would be over-constrained. This justified a 4D saliency field approach to modelling light field visual attention.

In the next stage of our work, we developed the first 4D saliency field prediction model (called FGSE) for light fields, to our knowledge, the details of which are given in Chapter 4. Our evaluation showed that this model performs as well as the state-of-the-art model without the need to render the refocused image. This means that it can be used to predict the visual attention given the LF multi-view array and a depth map.

The subsequent alternative approach to our FGSE model developed a more unified representation of light field visual attention. This model (FDLSE) performed similarly to the FGSE model according to our chosen metrics, but visual inspection indicates superior performance. The quantitative metrics we used do not give a perfect measure of performance so the qualitative characteristics are also an important indicator of how well the saliency maps match the ground truth, which are illustrated in Section 5.6.2. A major advantage of this approach is that it eliminates the need for pre-computed disparity maps, which is a much more streamlined approach. As only the centre SAI needs to be estimated by DeepGaze II in the FDLSE model, computational complexity is greatly reduced.

In Chapter 6 we have shown that the saliency field can be used to rank renderings of the light field according to where people are most likely to look. To evaluate the effectiveness of our saliency ranking scheme for focal slice selection we integrated the saliency maps with PSNR and found that overall better sharpness was achieved at salient regions in the slice selected by our automatic ranking compared to equidistant slices.

7.2 Re-Addressing Research Questions

We set out our research questions in Section 1.2. Here we re-address them and provide a discussion for how our research has answered them.
How can we quantitatively and qualitatively estimate how phenomena specific to light fields affect visual attention?

This question is a broad one, but the work outlined in this thesis goes a long way in answering it. Chapter 3 outlines the subjective experiment we carried out which was carefully designed to gather gaze data from participants from a range of demographics. The experiment was set up to compare whether all-in-focus renderings have different visual attention to refocus renderings. Our refocus renderings were shown to participants for both a fixed focal plane and for a varying focal plane (i.e. a sequence of focally varying renderings). We hypothesised that if we saw a large enough difference between the visual attention of the all-in-focus renderings compared to the refocus renderings, this would indicate that the visual attention of light fields is fundamentally different to that of an all-in-focus image. We further hypothesised that a model could be created (to take advantage of depth information) which could predict saliency for light field renderings. This could then be compared to saliency predictions for all-in-focus images to provide a quantitative and qualitative measure of the influence of light-field specific phenomena.

The analysis of the dataset gathered indicates a number of key insights. Firstly, the ground-truth saliency maps of focal-sweep renderings and region-in-focus renderings reveal salient regions not present in the all-in-focus rendering’s saliency map. As a tangible example, Fig. 3.5 clearly shows that compared to the all-in-focus rendering, the region-1 rendering saliency is much more focused on the upper window, which is in-focus. This is captured to a degree in Table 3.1 where for the same Medieval example, the all-in-focus entropy was 4.02 and region-1 entropy was 3.71. Unfortunately, existing metrics mostly fail to capture this observation, so we have to rely more on qualitative analysis and visual inspection to reach this conclusion. However, it is clear from Table 3.1 that the entropy values for the various light field renderings give an indication that people tend to look at different regions of an image depending on how it is rendered.

Secondly, we found that the order of focal sweep renderings (i.e. front-to-back vs. back-to-front) affected the saliency differently for some light fields, but for others were symmetrical. This is again something that existing evaluation metrics struggle to fully capture, but can be observed by visual inspection.

Thirdly, focus affected the saliency of light fields in varying ways, where we observed that it had a strong influence for some light fields and a relative weak influence for others. We speculate that this could be due to our light field data having varying properties such as depth, aperture size, etc. This is captured in the $R^2$ scores in Table 3.2 which clearly show that, for some light fields, people will follow what is in-focus very strongly (for example Tarot Small) whereas for others, people tend to ignore the focus (for example sideboard). Similar to the point above, this also depends on the order in which the renderings are displayed (i.e. front-to-back...
rendering may guide visual attention very strongly according to focus, but back-to-front may not).

In conclusion, we cannot claim to comprehensively understand every possible cue that could affect visual attention of both light-field and non-light renderings, but we can say that there are characteristic phenomena intrinsic to light fields which influence the saliency maps of light field renderings, but which do not influence the corresponding all-in-focus saliency map. Some of these cues are likely to be deep-rooted behavioural responses, for example looking at a person’s face more, which are very difficult to detect. However, there are some cues which can be anticipated by a model, which is addressed in the next research question.

**How can we build a model to predict the visual attention of light fields which uses intrinsic properties specific to light fields, thereby providing interpretability as well as high prediction accuracy for a suitable evaluation methodology?**

This research question is mainly addressed by Chapter 4 and Chapter 5. The FGSE and FDLSE models presented in these chapters are clearly evidence that a light field saliency field model can be constructed to predict the visual attention of any light field rendering. However, our research question specifically asks whether models which use intrinsic properties of light fields can be built, which we claim our models both do use. The FGSE model uses the depth maps of the light field as input. This is unique to each light field and is used to infer the focus cue, which is then combined with the DeepGaze II output saliency map(s). The FDLSE model uses the multi-view array to estimate a disparity cost volume, which again is unique to each light field. This is also combined with the DeepGaze II output saliency map(s). The details of these models have been provided in the respective chapters. Our results show that our models performance are comparable to leading models. However, our models have a huge added benefit in that they do not require the pre-rendering of the refocused image. Our models are more interpretable than approaches such as direct deep learning, as they use depth information from the light field (which may be inferred) which weights the saliency map more heavily towards regions that are likely to be in-focus. This question could be investigated further, and we expect that our models will serve as a baseline for future models in light field visual attention research.

**How can the models proposed in question 2 be used to generate saliency maps of light field renderings, which in turn can be applied to downstream applications such as ranking and selection mechanisms for light field renderings?**

We have shown that the FGSE and FDLSE models we created were specifically designed to generate saliency maps of light field renderings. Chapter 6 demonstrates how this functionality allows us to then address downstream applications which depend on this knowledge. We
specifically outlined how a selection algorithm can be designed to rank focal slices. We first generated a series of saliency maps (saliency slices) corresponding to slices of a focal stack and then ranked them according to the sum of pixel intensities (saliency sum). Secondly, we selected a small number of the focal slices with an effort to balance the following trade-off: the saliency sum of the corresponding saliency slice was as high as possible while ensuring the slices chosen were as spread out as possible along the depth plane. This was developed for gaze contingent blur rendering to facilitate the synchronisation of defocus blurring and binocular gaze which may become unsynchronised due to temporal lag in scenarios with computational constraints. Since, having fewer pre-rendered viewing planes to switch between would likely help to reduce lag among other benefits, our selection scheme limits the number of slices selected to six.

7.3 Future Work

This thesis has outlined a complete overview of the work done to date on visual attention prediction for light fields. We view this field as one which has still a lot to yield. Given the recent technological advances and pervasiveness of Head Mounted Displays there is strong motivation to develop more accurate and quicker LFVA models to help resolve problems such as the vergence-accommodation conflict and visual fatigue/nausea that arise from renderings that do not correspond with the user’s gaze.

Deep Learning Models

Generally, deep learning models provide the best performance for visual saliency prediction and computer vision in general. The work we have carried out would suggest that light field visual attention prediction is no exception to this trend. A simple way to potentially improve the performance of our LFVA models, would be to replace DeepGaze II with the current highest performing image visual attention model on the MIT/Tübingen Saliency Benchmark [30]. Additionally, future work could also investigate the implementation of deep learning models that are specialised to work with the light field and ground truth data specifically, with special consideration given to the nature of light field renderings in relation to regular 2D images. This has already recently been investigated for light field SOD [11] with promising results. However, a major hurdle for deep learning is the vast datasets [140, 90, 65] that they tend to require to sufficiently learn a sufficiently accurate model. To gather our dataset for 21 participants, it took approximately 2 weeks from the start of recruitment to finishing the data collection phase and required the highly specialised Eyelink-1000 eye tracker (costing approximately 30 thousand euro). This suggests that a large amount of resources may be required to gather sufficient data for a deep learning model, which might not be feasible for the majority of research teams.
Therefore, methods such as data augmentation and synthesis could be investigated to alleviate and circumvent this issue.

**Quantifying the Effect of the Focus Cue**

Since the focus cue was found to have a noticeable effect on visual attention and is a variable parameter of the light field which distinguishes it from other visual media, we made this a hallmark of our study on light field visual attention. To expand on this research topic, an important next step would be to analyse how adjustments to angle and changes in aperture type and size may affect light field visual attention. To assess the strength of the focus cue on attention for the different light fields we used the $R^2$ measure in relation to focal length. However, other thresholds/parameters and distance measures could be used. There would also be value in addressing the correlation between the layout and features of the scene and the strength of the focus cue. Since we only evaluated the VA estimation of two refocus renderings per light field, there is scope for our models to be extended to evaluate on refocus renderings from a greater variety of disparities.

**Evaluation Metrics**

Another key area which could benefit from research is that of evaluation of these models. Commonly used metrics such as KLD, SIM, CC, AUC and NSS [84] do not fully capture some key aspects of saliency in the light field domain. For example, an evaluation metric which could provide some meaningful way to measure the difference in saliency for two renderings could greatly help in identifying how light field specific cues affect visual attention. Another useful metric would be one which could give a measure of how well a saliency map predicts ground truth visual attention for the full stack of renderings. Our FGSE model was not statistically different to our FDLSE model according to the quantitative metrics that we used, but a more carefully designed metric would have revealed the subtle improvement that the FDLSE produced. We are aware that metrics such as Wasserstein distance may be suitable for this directly, or with slight modifications.

Future work in the field of quality assessment metrics would also be worth investigation. As seen in Croci et al.’s work [70], saliency maps obtained for omnidirectional images can be integrated into quality metrics and thus reflect more closely human perception. This is not mutually exclusive from the above, as light field saliency metrics usually aim to capture aspects of human perception.
Developing a more Efficient Model

As mentioned in Chapter 5, saliency maps are typically used for downstream applications, such as selecting which renderings to display to a user. However, for each possible refocusing of the scene, a saliency map can be generated. This can be circumvented by sampling, as we did in Chapter 5, but in some cases it would be of value to be able to quickly dynamically render a saliency map. Our models both use DeepGaze II, which we found takes approximately 10 seconds to run per image using a desktop with a GPU. This creates a computational bottleneck and rules out the possibility of generating and using saliency maps in real-time. Aside from training an end-to-end light field specific saliency prediction deep learning model as mentioned above, using a faster method than DeepGaze II for predicting the saliency of the centre SAI could enable our FGSE and FDLSE models to be used for near real-time applications. Similarly, a faster depth estimation method would reduce the computational overhead for the FGSE model. Our models could be further optimised to reduce computational complexity, for example, by estimating the saliency map in a lower resolution, using a method faster than DeepGaze II for predicting the saliency of the centre SAI, or using a faster depth estimation method.

Application in Virtual Scene Design

We anticipate that light fields will be widely used in content creation in areas such as gaming, film, etc. Virtual scenes are being more widely used, however their creation is a very time consuming process. Often they will be roughly prototyped in the pre-viz stage, but then a decision has to be made on which parts of the scene to spend time detailing. To add further complexity, for light field displays, occasionally changes will occur in how a scene is rendered, such as refocusing. We postulate that our light field visual attention prediction model could be used to output saliency maps of sample renderings to act as a guide for virtual effects (VFX) artists for which parts of the scene to spend time developing and which parts people rarely look at and can be kept more simplistic. Similar work has been carried out for cinematic virtual reality (VR) post-production [71]. In this work, the saliency maps of omni-directional images were used to inform scene presentation and give content creators an idea of possible unintended viewer distractions among other useful insights. This gives a strong indication that analogous work, using light field saliency maps in the post-production of light field rendered media, would be worth investigation.

Application in Light Field Compression

Since light fields are four-dimensional, they contain a large amount of information relative to traditional media. This leads to a longer processing time when working with light fields, as well as other practical issues such as efficient storage and distribution. In past research, saliency has
been used as a way to compress images. For example, Chen et al. [141] proposed the multi-resolution wavelet domain foveation algorithm which can achieve higher compression rate than the traditional model with low computational cost. Li et al. [142] has also proposed a high efficiency video coding (HEVC) main still picture profile algorithm, which optimizes saliency-guided PSNR (viewed as perceptual distortion) such that the coding efficiency of HEVC-based image compression can be significantly improved from a subjective perspective. These works strongly suggest that investigating the use of saliency for light field compression could yield good results. For light field compression, Pendu et al. [143] have proposed a binary hierarchical scheme to encode and efficiently transmit light fields in the Fourier Disparity Layer (FDL) representation. We anticipate that using our saliency field in combination with this approach could yield an efficient and high-quality compression of the light field from a subjective perspective.
Bibliography


