École polytechnique de Louvain

Design, Implementation, and Evaluation of Multi-Platform Graphical Adaptive Menus

Authors: **Gilles NGONGANG T., Mohammed AL LAMI**
Supervisor: **Jean VANDERDONCKT**
Readers: **Charles PECHEUR, Benoît DUHOUX**
Academic year 2019 - 2020
Master [120] in Computer Science and Engineering

Abstract

The advent of multiple computing platforms, ranging from small smart watches to large wall displays, poses the problem of adapting their graphical user interfaces to multiple screen resolutions. Instead of developing separate versions for each platform, responsive user interfaces promote the idea of developing a single graphical user interface that accommodates the variations induced by the numerous screen resolutions. While this principle applies to the entire interface, it does not scale down easily to graphical menus, such as menu bars, pull-down menus, and sub-menus.

To identify potential solutions, we conduct a Systematic Literature Review (SLR) summarizing the characteristics and findings of N=407 scientific references on graphical menus. We highlight the *descriptive*, *comparative*, and *generative virtues* of our examination to provide practitioners with an effective method to (i) *understand* how graphical menus position in the literature; (ii) *compare* their studies from different authors; and (iii) *identify* opportunities for new development. We make our large corpus of papers accessible online as a Zotero collection. Based on this review, we develop two original graphical menus intended to adapt themselves to the various screen resolutions of multiple platforms: a fractal menu exploiting the scale invariance of fractal geometry and a bilinear menu based on geometric deformation. For both menus, the motivations are justified, the design, the implementation, and the evaluation are performed. Finally, a conjoint analysis examines the end users' preference of forty graphical adaptive menus, separately over time.

Contents

Chapter 1

Introduction

This chapter first discusses the context of the problem: the multiplication of computing platforms with varying screen resolutions which raises the need of multi-platform Graphical Adaptive Menus. It then defines the mission statement of this master thesis along with its working hypotheses. It finally provides an overview of the thesis structure.

1.1 Context of the Problem

In today's world, where many computing platforms exist with varying screen resolutions from 85 inch smart TVs to 1.60 inches smart watches, the need for custom GUI relevant for these varying platforms arises. In fact, menus are they most used items of GUI and with devices nowadays providing more and more functions, and most at times same functions are available on different platforms with varying screen resolution making menu design more complex especially for platforms with small resolution (e.g., smart glasses, smart watches etc.). The complexities introduce by these menu designs usually makes the user experience less efficient especially in small resolution screens e.g., reaching a particular menu item may sometimes need several navigation steps as smaller resolution usually reduces menu items displayed and increases menu depth. To reduce these complexity and maintain an efficient user experience, the need for multi-platform graphical adaptive and responsive user interface arises. Given these new limitations, it is vital to design menus that accommodates various screen resolutions and orientations at the same time, which is challenging. Such new menu design should take into account the depth of the menu (how many levels) and the breadth (how many items per level). These designs should equally implement mechanisms to optimize menu items viewing and selection.

As of today, many menu designs have being proposed (e.g fish-eye menu [70], circle menu [115] etc.) to address some of these limitation but most don't address the varying resolution of the GUI. Hence these designs aren't always adequate for all devices.

1.2 Mission Statement

1.2.1 Thesis Statement

The central objective of the thesis is to investigate original adaptive menu designs that accommodates platform variations, namely a fractal menu and bilinear menu .

The fractal menu investigates a new design based on fractal geometry to address the depth of the menu and its breath. The bilinear menu, investigates new techniques for efficiently viewing menu items. The two designs will enable the users to navigate the menu space more efficiently by providing better screen space utilization and viewing experience.

Practically, the fractal menu designs will investigate new menu shapes and layout (arrangement, view items, view depth) so as to accommodate platform variations. Once this shape and layout have being addressed by the fractal design, bilinear menus investigates how to better visualize menus items mainly through distortion but without affecting the menu layout and shape. Both techniques investigated in this thesis, one the fractal for new multi-platform menu design and the bilinear design for optimizing item visualisation and selection, we believe will provide practitioners with more information on designing innovative adaptive multi-platform menus.

1.2.2 Working Hypotheses and Research Method

Multi-platform graphical adaptive menus are necessary for a uniform and efficient multi-platform user experience. Our research methodology consists in a systematic literature review to motivate our thesis statement. Once the SLR have being done, we proceed to the design and implementation of the our innovative adaptative menu interaction techniques namely fractal and bilinear. Finally, we do an analysis of end users' preference of forty graphical adaptive menus.

1.3 Outline of the thesis

This thesis is structured as follows: Chapter 1 is an introduction to our thesis, we present the context of the problem and mission statement. Chapter 2 presents a systematic literature review we did to motivate the creation of the two innovative menu selection technique. Chapter 3 and 4 respectively presents fractal menu and bilinear menu which are innovative menu design and interaction techniques. Chapter 5 presents a conjoint analysis of existing graphical adaptive menus. In chapter 6 , we discuss limitation of our thesis, further studies and improvements which can be considered and we present a conclusion statement on the study.

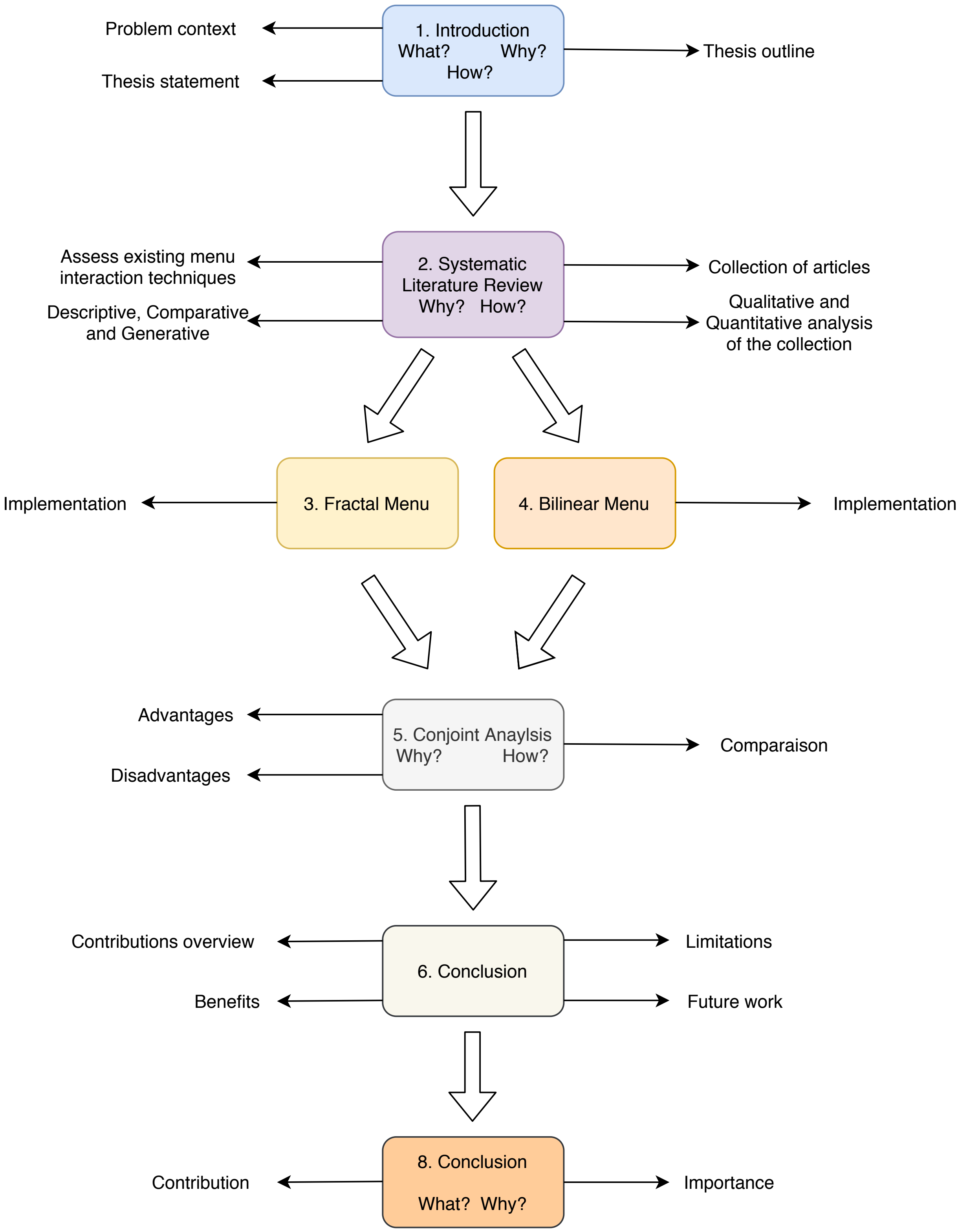


Figure 1.1: Thesis Flow Chart

Chapter 2

A Systematic Literature Review of Graphical Menus

In chapter 1, we introduced the context of the problem and mission statement. In this chapter, we describe the systematic literature review we did to assess existing menu interaction technique and motivate the creation of the two new menu interaction technique named Fractal and Bilinear which we propose in chapter 3 and 4 respectively. In this chapter, section 1 motivates the need of a SLR. In section 2, we describe the research method we used to conduct the SLR and lastly, section 3 and 4 respectively presents quantitative and qualitative results of our SLR and their analysis and lastly in section 5 we present a conclusion on the various virtues.

2.1 Introduction and Motivations

The goal of the SLR on menus both adaptive[1] and non-adaptive are threefold:

- Descriptive: Every menu will be described in the same way, with the same classification criteria which we will present later.
- Comparative: Two or more menus can be compared to each other based on the same classification.
- Generative: Generate new types of menus that do not exist yet, in our case, fractal menu and bilinear menu .

2.2 Research Method

To conduct our SLR, we used the four phase flow procedure as inspired by Liberati *et al.* [306] and used in [499]. The four phases are Identification, Screening, Eligibility and Inclusion. The following figure resumes our method using a PRISMA[2] diagram:

[1] Here, we make reference to those menus whose adaptation is controlled by the system.
[2] Prefered Reporting Items for Systematic reviews and Meta-Analyses.

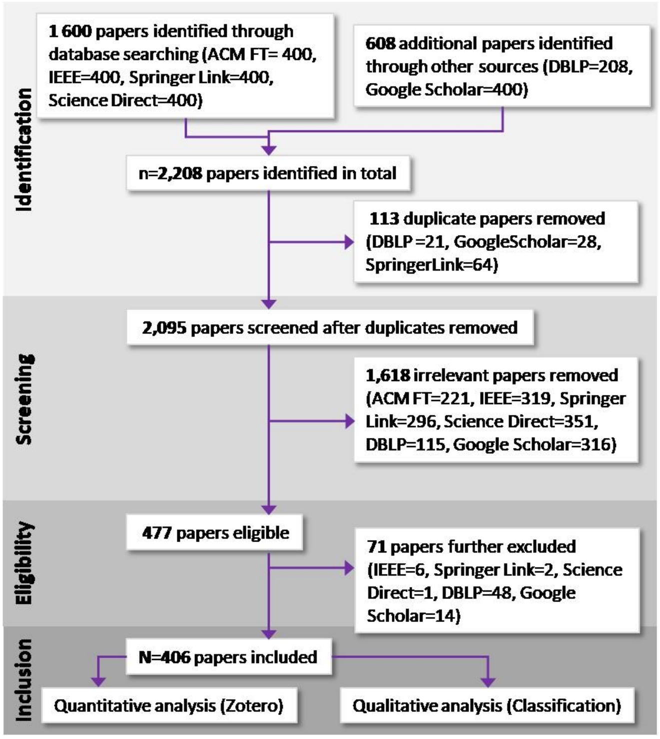


Figure 2.1: The four-phase flow diagram of our SLR process.

Phase 1: Identification

To obtain the papers relevant to our study, we queried 4 digital libraries (ACM Digital Library[3], IEEE Xplore[4], Elsevier ScienceDirect[5], SpringerLink[6]) and 2 multi-publisher sources (DBLP CompleteSearch[7], GoogleScholar[8]) using the following syntactical query:

SQ = (Menu) AND (technique OR system OR widget OR structure OR breadth OR depth OR item OR interaction)

To ensure consistency of the results, all the libraries were queried and results/references downloaded the same day. Once, the references obtained, we downloaded the related PDF files over several subsequent days, firstly to avoid excessive request that might block our access to the libraries and secondly because for most of the queried libraries, we needed to be on our institution's network to have an authorized and free access. We used the following rules as proposed in [1] when querying the libraries and downloading the results:

- The "advanced search" rule : We made sure to run the queries on each library/multi-publisher sources with the advanced features activated and when possible, sorted the results in order of relevance.
- The "minimum number of references" rule : If a query returned less than 400 references, they were all retained.

[3] https://dl.acm.org/ queried in Full Text
[4] https://ieeexplore.ieee.org/Xplore/home.jsp
[5] https://www.elsevier.com/solutions/sciencedirect
[6] https://link.springer.com/
[7] https://dblp.org/
[8] https://scholar.google.fr/

- The "maximum number of references" rule : If a query returned more than 400 references, just the first 400 were retained. We made a quick analyses of the rest just to make sure we didn't ignore a relevant reference.

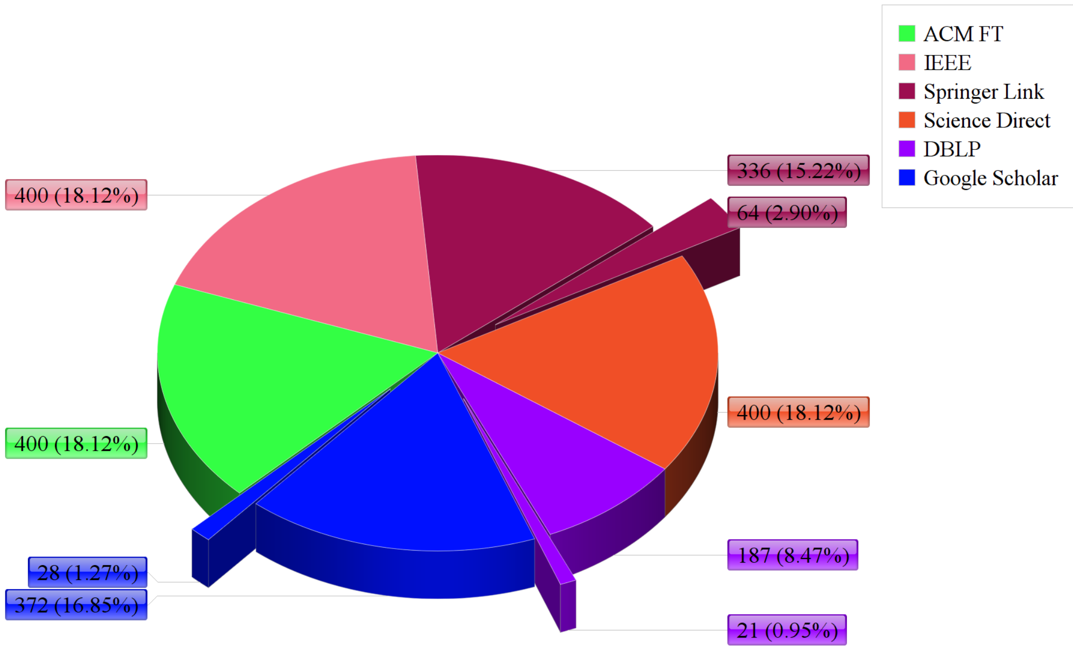


Figure 2.2: SLR Identification Phase Summary.

Our queries identified a total of **2208** references from the six sources. More precisely, 400 references each for ACM FT, IEEE, Springer Link, Science Direct, Google Scholar and 208 references for DBLP. We further excluded duplicates references (DBLP CompleteSearch = 21, GoogleScholar = 28 and SpringerLink = 64). Giving a total of 113 references excluded on this basis. We therefore ended with a total 2208 - 113 = **2095** references whose corresponding papers we downloaded.

Phase 2: Screening

We evaluated each of the **2095** papers for relevance with respect to our research scope by checking it against inclusion and exclusion criteria related to both form and content as proposed in Vanderdonckt *et al.* [488]. Our evaluation technique included reading the paper's abstract and if necessary the paper as a whole to have more information before accepting or rejecting it with respect to a given criteria. We define below the various criteria used during this phase:

A. **Criteria related to Form**

1. **Inclusion criteria** : Are included all references written in English that underwent a competitive peer-review process and where the full text is available. This subsumes all short and full research papers published in peer-reviewed journals, conferences, symposia or workshops.
2. **Exclusion criteria** : Are excluded all references that did not result from a competitive peer review process or are applied contributions. This subsumes

such as books, Ph.D. theses, master theses, patent descriptions, standards, style guides, book or thesis summaries and reviews, technical reports, white papers, invited talks, demonstration papers, doctoral consortium papers, tutorial papers, poster publications, editorials, prefaces, articles or columns in magazines, newsletters, encyclopedia entries, blog posts, or social network entries. References where the full text was not published or accessible, such as abstracts, extended abstracts, and presentations (e.g., slideshows) were also excluded. For instance, a reference which was considered relevant after examining its abstract could be unfortunately withdrawn if the full paper requires a paying access.

B. **Criteria related to contents**

1. **Inclusion criteria**: Are included all references complying with at least one of the following three conditions.
   a. The abstract or the full text explicitly introduces a new menu interaction technique taken as a whole.
   b. The abstract or the full text discusses any menu interaction technique emphasizing at least one discriminating feature either as input or as output or both. A feature is referred to as discriminating when it is the subject of a study that proves some originality and impact with respect to other techniques. Menu features are classified into three mutually exclusive levels:
      - Generic features: are independent of both the menu interaction technique and any context of use. Such features include as device, computing platform, input/output interaction modality, amount of modalities, pressure. For instance, pressure is considered as generic because it could be studied for any widget (not just menu) in any context of use (not just menu selection).
      - Specific features: are dependent of the menu interaction technique, but independent of any context of use. Such features include breadth, depth, activation area, item rendering, geometric shape, item ordering (e.g., by frequency, by importance, by critically, by consensus, optimized according to a certain criteria), menu selection performance. For instance, the item ordering is considered as specific because this property affects the overall quality of the menu and cannot be found in other widgets.
      - Contextual features: are dependent of the menu interaction technique as applied in a particular context of use. Such features include: domain of human activity, user profile, environment, location, and task. In principle, any contextual parameter may be considered. For instance, a menu could be aimed at elderly people taking pills at home.
   c. The abstract or the full text explicitly uses terminology for studying a set of menu interaction techniques. This subsumes for instance references that are themselves addressing the research questions of this SLR, such as survey papers, design space papers, comparison papers. As a sanity check, each such reference will be recursively examined to check whether references satisfying conditions (1a) and/or (1b) are considered. The condition (1c) ensures not excluding technical, mostly implementation-oriented references performed within a particular context of use, but nevertheless addressing

some discriminating menu feature. Additionally, the following mandatory conditions need to be fulfilled.

- It can be deduced from the abstract or the full text that the menu interaction technique is explicitly researched within the context of Human-Computer Interaction (HCI) in general, and within the context of menu interaction techniques in particular. It tackles one or several research topics directly relevant for the constitution of menus as the main topic.

2. **Exclusion criteria**: Are excluded all references that fulfills any of the following conditions

   a. Research questions dependent conditions: the reference explicitly mentions a menu interaction technique, but

      - Ri1: it is aimed at studying features not belonging to the three aforementioned categories, e.g., psychological, environmental factors that may influence menu design.
      - Ri2: it quotes a menu interaction technique without studying it, e.g. as a reference.
      - Ri3: it presents opposite contents, e.g., ”contrarily to pie menus, rotary widgets are...”

   b. Research questions independent conditions: the reference does not explicitly mention any menu interaction technique or any discriminating feature, such as in the following cases:

      - Unrelated keywords (UKW): the query keywords are actually included in the reference, but in an order or in a position that makes them unrelated, without any connection. For instance, ”menu”, ”interaction”, and ”technique” could be present in a reference but far away from each other so that there is no more any semantic relationship between them.
      - Homonymy of keywords (HOM): the query keywords are actually included in the reference, in a right order or position, but with a different meaning. For instance, any reference that is performed outside the scope of HCI and/or SE is excluded, such as menu interaction in the context of restaurants is considered out of scope.
      - Homography of keywords (HOG): the query keywords are actually included in the reference, but are written, spelled, or formatted differently than for the intended meaning. For instance, ”the MENU project in chemistry has an interaction with...”, ”the menuet technique is more complicated than...”, ”the M.E.N.U. organization has a strong technique for interaction with other companies”.
      - Antinomy of keywords (ANT): the query keywords are actually included in the reference, but convey an opposite message than the intended meaning. For instance, ”anti-menu patterns offer an interaction that...”.

The pie chart below summarizes the percentages of papers excluded per library after this phase for the 2095 paper considered. At the end of the screening phase, a total of **1618** papers were excluded with respect to the above mentioned criterion.

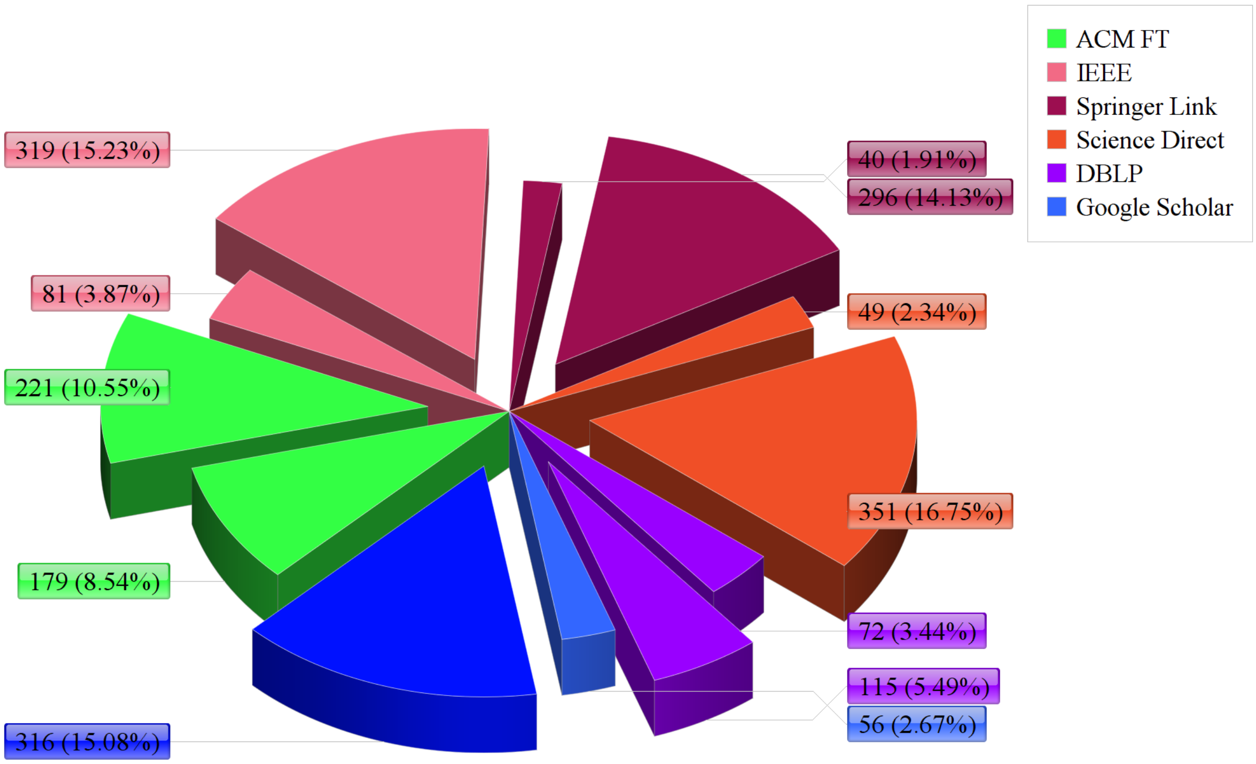


Figure 2.3: SLR Screening Phase Exclusion.

Leaving us with 2095 - 1618 = **477** papers.

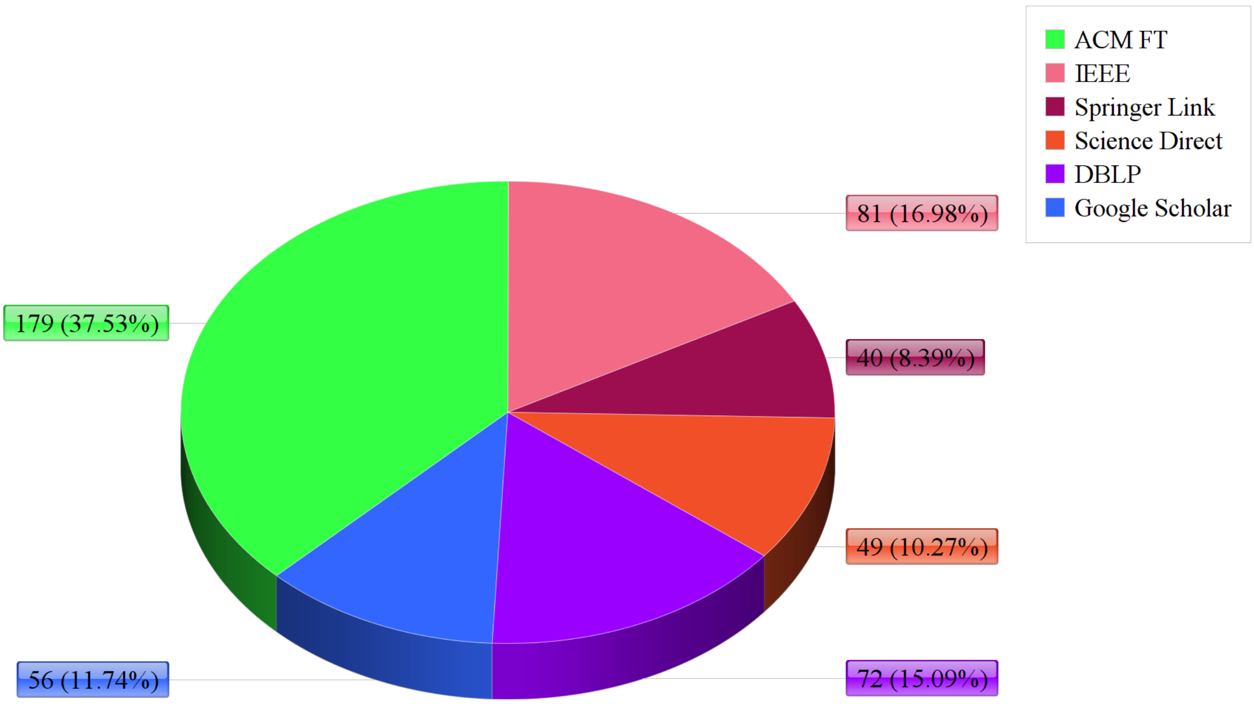


Figure 2.4: SLR Screening Phase Summary.

Phase 3: Eligibility

We further excluded:

- Those papers that matched our research topic but did not report its results.
- Duplicates papers that appeared in the another queried library. At the end of the identification phase, we had removed duplicates within the same library. Here, we removed eligible papers from a given library if they were already added in another using as reference the inclusion date.

Using these rules, we further excluded: **71** papers leaving 477 - 71 = **406** papers.

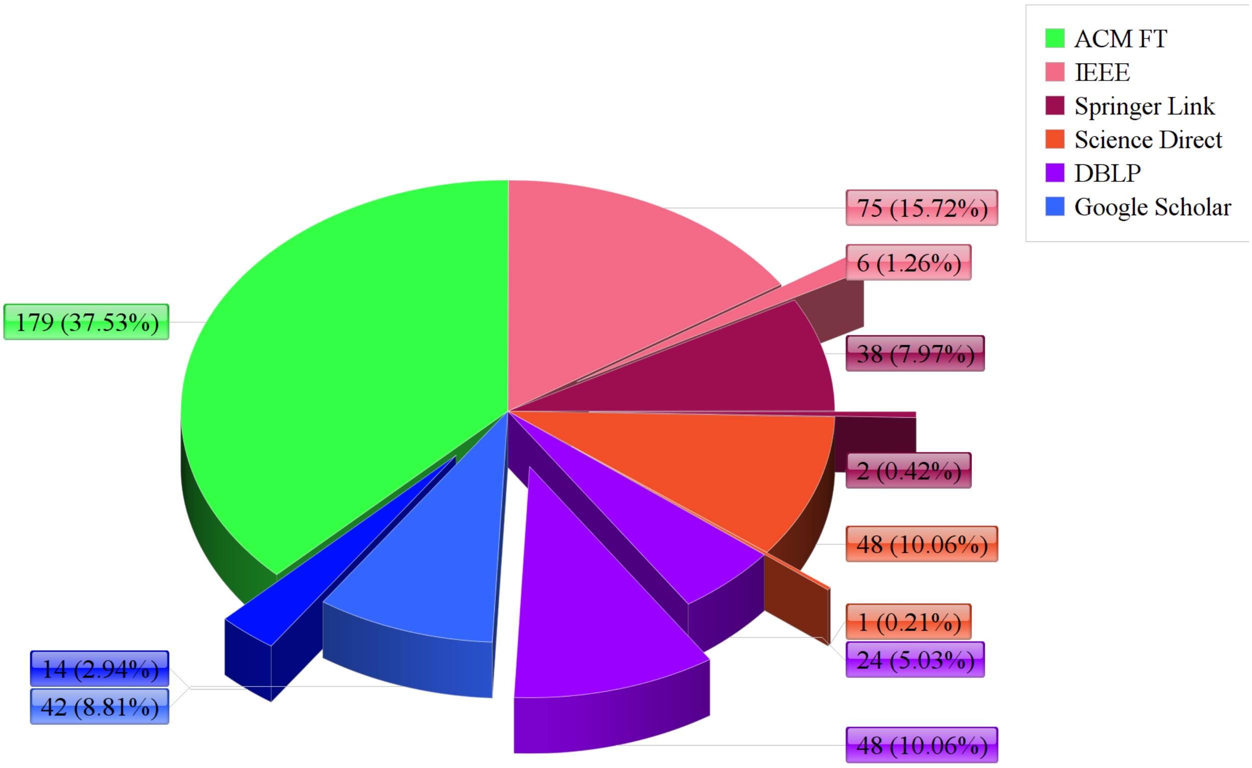


Figure 2.5: SLR Eligibility Phase Summary.

Phase 4: Inclusion

In this phase, we did a quantitative and qualitative study on the selected papers using several tools. In the next sections, we present the results of these studies and discussion on some of the graphics generated from the tools used.

2.3 Quantitative analysis

For the quantitative analysis, We employed a series of tools to create a collection of papers and generate summary statistics. Namely, we used :

- Zotero[9], which is a bibliography management software which we used to create a collection of the papers.
- Zotero Connecter, for automatically saving items to Zotero via a single click.
- Zotfile[10], a Zotero plugin to manage references: automatically rename, move, and attach PDFs (or other files) to Zotero items ...
- PaperMachines [11], to generate various graphics from our included papers some of which we will discuss in the qualitative analysis sub section.

Most of the quantitative analysis was summarize at the end of each phase. By the end of phase 4, our zotero collection had the following composition:

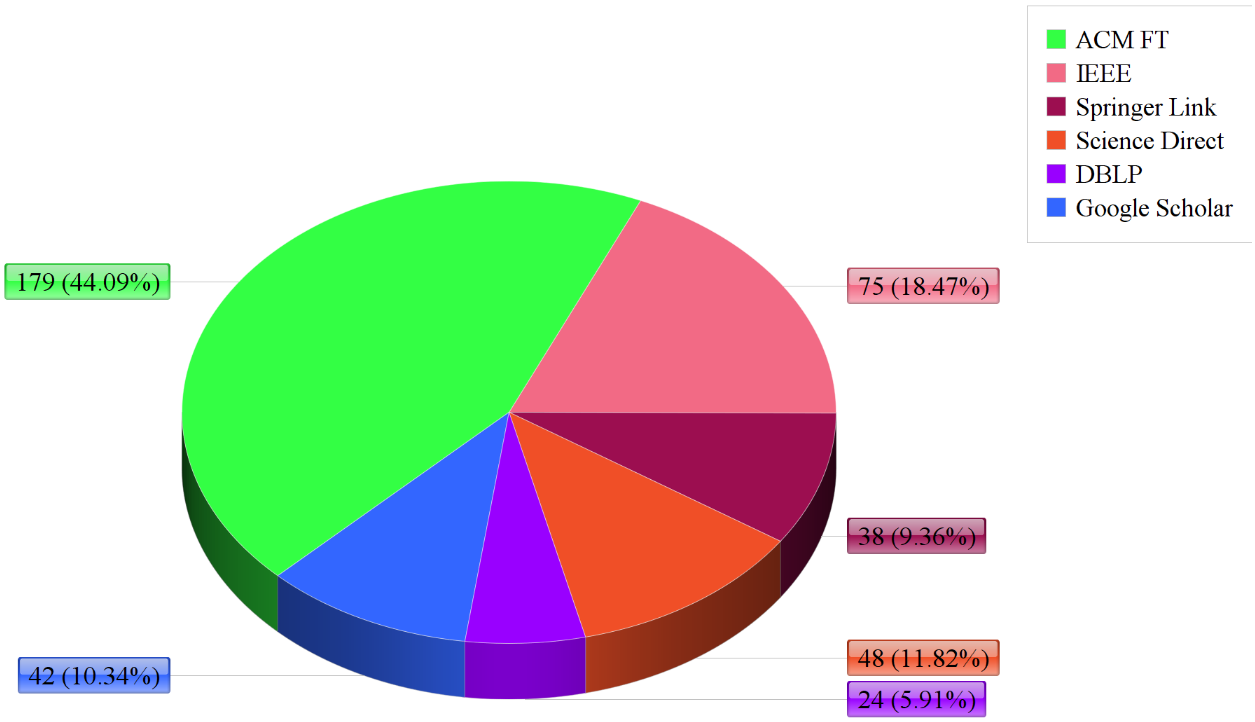


Figure 2.6: SLR Inclusion Phase Summary.

Table 2.1 summarizes the references count obtained by the queried libraries and multi-publisher sources.

[9] https://www.zotero.org/groups/2317119/menu_review/library
[10] http://zotfile.com/
[11] http://papermachines.org/

Source	**Query**	**Rules**	**Duplicates**	**Excluded**	**Included**	**References**
1. ACM Digital Library	575,142	400	0	221	179	[178, 176, 53, 233, 173, 221, 407, 55, 118, 252, 172, 330, 218, 434, 401, 520, 219, 198, 49, 444, 46, 298, 236, 254] [370, 202, 44, 84, 312, 530, 510, 519, 91, 305, 342, 117, 50, 101, 310, 377, 265, 26, 122, 485, 461, 443, 48, 403] [43, 283, 391, 354, 492, 410, 120, 398, 255, 270, 61, 276, 381, 429, 37, 258, 92, 13, 308, 100, 335, 469, 357, 438] [352, 442, 140, 282, 493, 119, 284, 447, 114, 458, 359, 104, 121, 80, 241, 250, 279, 399, 28, 318, 531, 128, 286, 483] [472, 72, 71, 291, 99, 534, 19, 15, 52, 277, 336, 59, 209, 206, 180, 253, 124, 197, 30, 139, 129, 215, 416, 371] [392, 477, 115, 69, 538, 333, 523, 299, 257, 430, 376, 45, 347, 404, 453, 337, 155, 432, 226, 423, 177, 467, 171, 20] [192, 108, 285, 303, 468, 526, 18, 154, 532, 334, 211, 189, 527, 199, 300, 194, 225, 16, 521, 481, 428, 288, 208, 533] [484, 148, 85, 174, 502, 517, 465, 524, 93, 95, 94]
2. IEEExplore	1,505	400	0	325	75	[375, 238, 280, 60, 515, 66, 127, 264, 339, 193, 319, 528, 162, 512, 436, 75, 420, 460, 498, 348, 123, 457, 361, 179] [390, 332, 231, 422, 409, 262, 190, 412, 311, 408, 522, 397, 147, 21, 363, 278, 384, 86, 353, 159, 421, 459, 143, 230] [82, 110, 242, 22, 383, 466, 529, 413, 151, 274, 435, 132, 188, 130, 440, 275, 537, 259, 81, 411, 244, 191, 145, 96] [297, 507, 240]
3. Elsevier ScienceDirect	55,862	400	0	352	48	[345, 107, 324, 212, 449, 372, 246, 135, 213, 535, 464, 441, 482, 70, 152, 364, 54, 340, 47, 446, 385, 272, 328, 163] [379, 380, 269, 514, 338, 415, 386, 133, 229, 29, 349, 356, 355, 393, 503, 38, 98, 39, 141, 228, 14, 418, 516, 295]
4. SpringerLink	106,708	400	64	298	38	[501, 65, 220, 315, 223, 320, 51, 6, 109, 146, 243, 360, 260, 518, 144, 474, 475, 150, 232, 64, 203, 131, 271, 341] [40, 137, 158, 462, 142, 5, 247, 525, 3, 2, 486, 4, 248, 495]
5. DBLP CompleteSearch	208	208	21	163	24	[362, 394, 317, 301, 378, 36, 374, 322, 113, 68, 292, 227, 126, 196, 57, 358, 452, 58, 367, 207, 125, 287, 309, 261]
6. GoogleScholar	>1,000,000	400	28	330	42	[456, 268, 513, 455, 536, 245, 34, 168, 448, 389, 478, 479, 216, 222, 439, 433, 136, 463, 293, 217, 369, 500, 25, 344] [506, 237, 350, 302, 325, 116, 187, 160, 450, 181, 480, 373, 74, 62, 454, 351, 473, 256]
Total	>1,739,217	2,308	113	1,689	406	406

Table 2.1: References obtained from the queried libraries

2.4 Qualitative analysis

In this section, we report and discuss some significant observations made on some of the material resulting from the aforementioned process. The graphics were generated using paper-machines as mentioned above. All the generated graphics including those not analyzed here can be found in the appendix under Paper machines generated graphics.

2.4.1 Word Clouds

To evaluate the important terms in the corpus, we generated two global word clouds namely: a Simple Term Frequency based word cloud (Fig 2.7) and a term frequency-inverse document frequency based word cloud (Fig 2.8). From these clouds, we concluded the following:

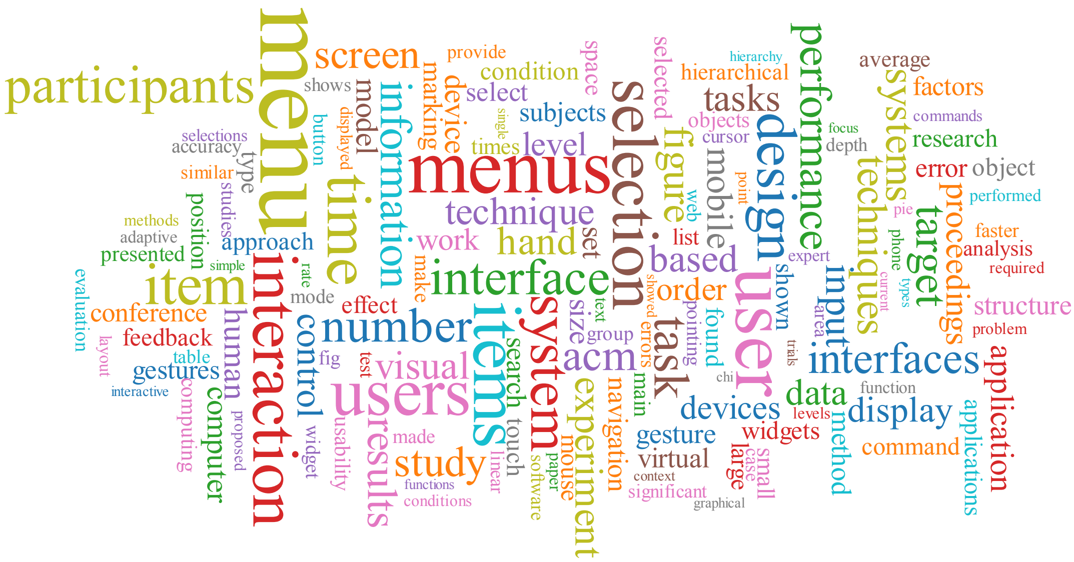


Figure 2.7: Global Word Cloud (based on simple term frequency).

1. Besides discipline-related common terms found in these clouds (e.g object, function, method, application etc.), the cloud suggests the following terms "menu(s), user(s), interaction, selection, performance". The appearances of the keywords "*selection*" and "*performance*" in this most important terms cloud makes us believe that selection and performance are very important aspect. The term "*selection*", we believe makes reference in general to item selection and particularly efficient item selection techniques. The term "performance" we believe makes reference to the efficiency of the item selection technique *i.e.,* how easily and efficiently a user can get to a desired item.

2. Subsequently, we equally identified terms such as "*accuracy, selections, interactive, phone, frequency, time, usability, navigation*" which we can group under the "usability" aspect of a developed menu. Menu should be developed such that they are easy to use regardless of the platform. These ease of use should consider aspects such as the accuracy of the selection, the selection times of desired menu item, frequency of selection to make more frequently accessed items more easily reachable.

3. We equally identified a less important term "gesture", which lets us believe visual menus are more frequently addressed that gesture menus.
4. We equally see the keyword "hand", in the large word cloud making us believe menu selection by hand is the most used form of interaction with menu. With the advent of mobile phones and smart devices which mostly have touch screens, this seems to be normal that this form of menu interaction is the most used.
5. We equally identify terms related to ways of interacting with these menus such as "*hand, cognitive, touch, finger*".

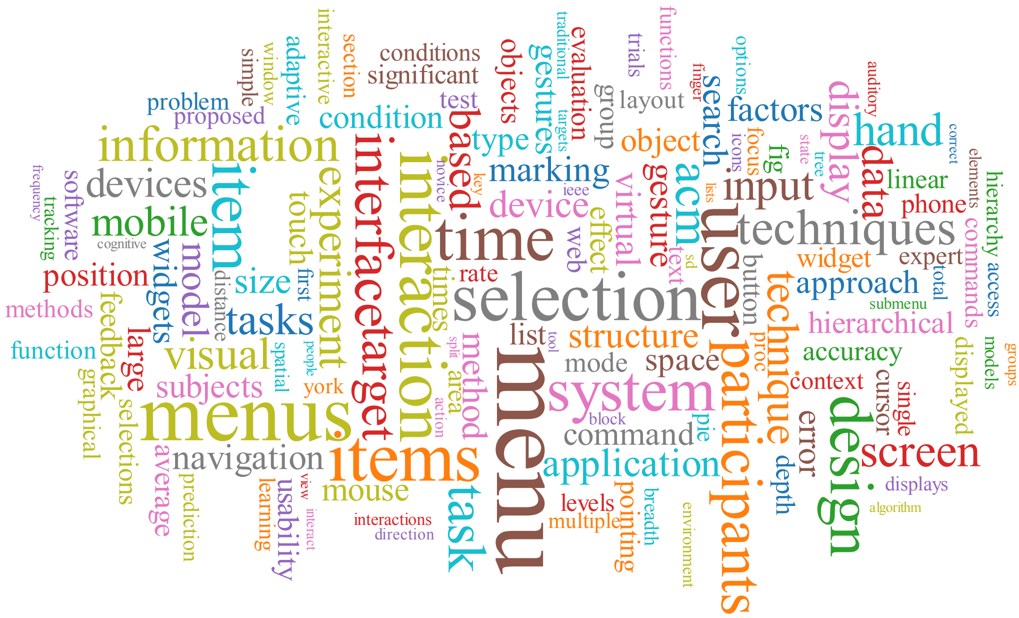


Figure 2.8: Large Word Cloud (based on Term Frequency-Inverse Document Frequency).

2.5 Conclusion on the Three Virtues

In this section, discuss preferences on adaptive menus as inspired by Jean *et al.* [496], graphical user interface of interactive applications are subject to three forms of adaptation namely: *user-control adaptation* where the user controls the menu adaptation, *system-control adaptation* where the adaptation is controlled by the system as the name implies and *user-system controlled adaptation* where both the user and system collaboratively manage the adaptability. Adaptability usually aims at improving three usual virtues namely : effectiveness [505], efficiency [184] and subjective satisfaction [165] [406]. As discussed in [496], adaptability has its advantages and shortcoming. A *user-control adaptability* gives the user full control and its appreciated for its flexibility but has the short comings of being time.

From our SLR, apart from Bailly *et al.* [53], there is no SLR for GUI menus today. In addition, this paper is a survey, not a SLR, and only targets graphical menus while we have been dealing with all forms of menus, including vocal, auditory, and multi-modal menus. However, the two menus that we are inventing are completely original and visual.

Therefore, following our SLR, we can draw the following conclusions on the three virtues:

A. Descriptive : There is a need to classify all menus with respect to variables like depth, breadth, structure, selection method, etc. Only a few of them are really adaptive to the platform. Most of them are just fixed and constant.

B. Comparative : The most frequent aspects addressed are often selection time, dwell time, and other human factors, but not much adaptive menus aspects are addressed.

C. Generative : We will create two new innovative adaptive menu selection technique: a fractal menus that proposed a multi-platform menu design and a bilinear menu that proposes new menu visualisation technique.

Chapter 3

Development of a Fractal Menu

Based on the SLR conducted in Chapter 2 , this chapter develops a first original multi-platform graphical adaptive menu, *i.e.,* the fractal menu, based on fractal geometry. The chapter is structured as follows: the background and the motivations for introducing this menu are first discussed. Then, we provide a definition of the menu and explain its design options which guided the implementation. Finally, an individual evaluation highlights benefits and shortcomings in a local conclusion. Chapter 4 introduces a second multi-platform graphical adaptive menu, *i.e.,* the bilinear menu. In chapter 5 , we present a conjoint analysis of graphical adaptive menus.

3.1 Background and Motivations for a Fractal Menu

From our SLR, we noticed not much papers except for Bailly *et al.* [53] address the large variety of menus that exist and how to characterize menu design space to facilitate menu creation. Most of the papers that mentioned menu designs instead proposed or investigated menu designs that aimed at optimizing the menu's usability by optimizing its effectiveness and efficiency. A menu's usability is usually optimized via three usual aspects [491] as mentioned in chapter 2 namely : effectiveness [505], efficiency [184] and subjective satisfaction [165, 406]. To optimize the menu's effectiveness, some design propose new ways to layout menu items. To optimize the menu's efficiency, most design try to optimize the reach time[1] for item selection. The reach time for a menu item is composed of two main components :

- **The visual search time** : The time needed to locate the item of interest among the set of a menu item [402]. This time reduces once the user becomes familiar with the menu structure.
- **The movement time** : The time needed to move the cursor (or finger) to the item of interest located by the visual search [402]. This time is a factor of the menu structure. Smaller menus turn to have smaller movement times.

Fitts' law[2] is generally used to calculate the **reach time** for menu items.

Lets consider some of these designs : for example expandable menus, where the list of options is exposed when a menu handler is clicked [402]. We consider in particular

[1] **Or selection time** correspond to the time needed to identify a item on the menu and select it.
[2] Fitts' law states "The time to acquire a target is a function of the distance to and size of the target."

drop-down menus [502], pie menus [358], marking menus [178, 176] where the pattern used to arrange menu items aims at optimizing the reach time. Pie menus and marking menus have proven to be measurably faster and more accurate in mouse and pen-based interfaces [274] . Pie or radial menus also have their limits [285]. In their study "Why it's Quick to be Square" Alstrom *et al.* [20], propose a new design for menus called square menus to minimize the selection time for experts[3]. In Racula *et al.* [402] we equally see that the average time to an item inside a rectangular menu is smaller than if the items were placed in a long pull down menu. Most websites use rectangular menu to optimise selection time by reducing movement time. If we want to archive a constant time from the handler to the items, then pie menu are the best option but due to their lack of familiarity the movement time is lost as the visual search time increases [402].

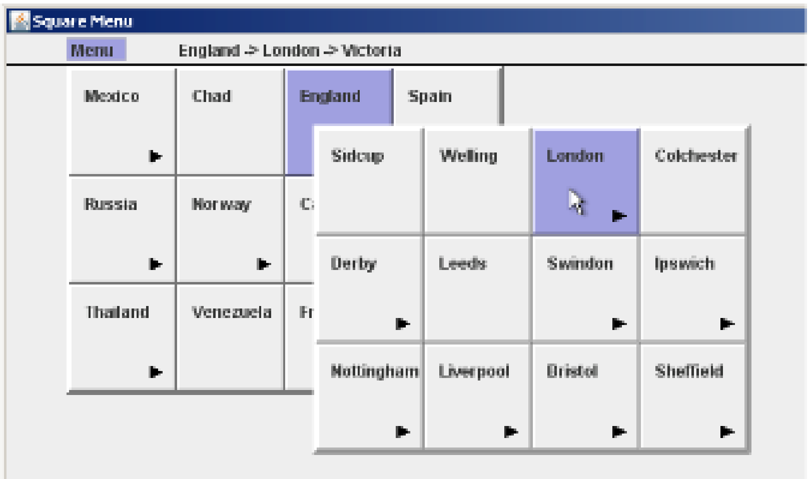


Figure 3.1: A hierarchical Square Menu by Alstrom*et al.* [20]

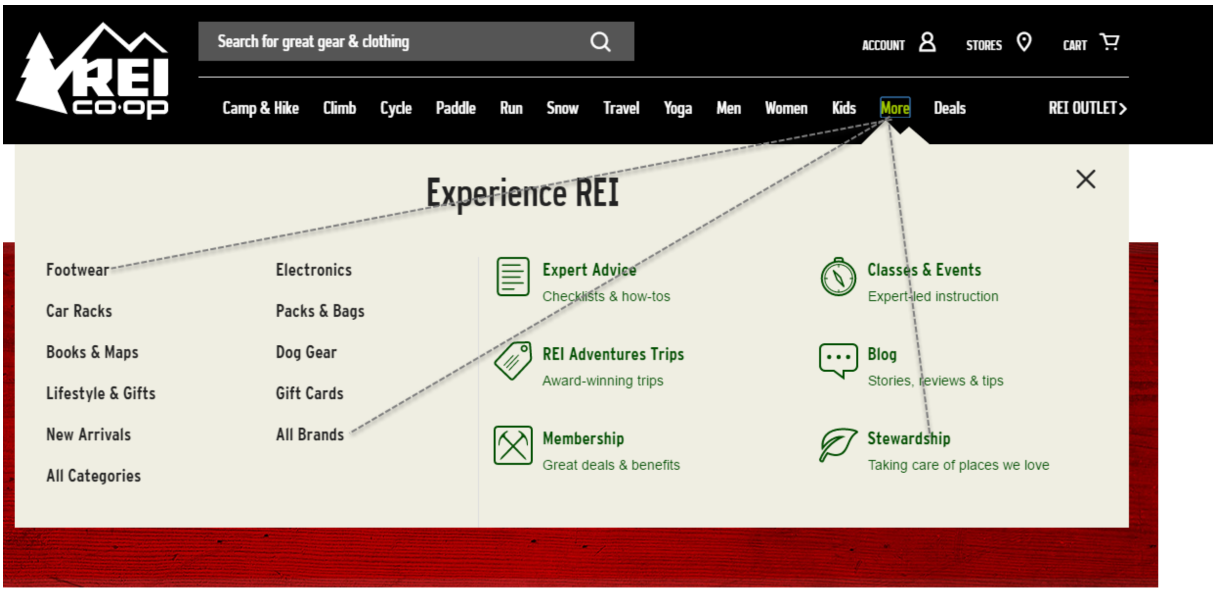


Figure 3.2: Rectangular Menu [402]

[3] People already familiar with the menu structure.

Most of these designs as we noticed don't take into account the variation induced by the varying screen resolution of the devices. Hence, some designs maybe adapted for some screen resolutions but not for others. For example, a pull-down menu can be adequate for a TV, desktop and laptop but not for a smart glass and watch especially when the menu items are many. The pie or radial menus may also have their limitation as the take up more screen space on display device due to their expansion over two dimension this may not be ideal for smaller resolution. Moreover, pie menus with more than eight choices don't scale well [73].

The need to design a multi-platform graphical adaptive menu and interaction technique that accommodates the variations induced by these varying screen resolutions therefore arises. The varying resolution and limited interaction options of some devices also affects the menu's usability. A user turns to adaptability only if the win exceeds the cost. Therefore, the new menu designs and interaction technique should not only accommodate varying screen resolution but should also implement mechanisms to optimize the menu's usability mainly: its efficiency and effectiveness.

In this thesis, we propose a new menu design and interaction technique based on fractals geometry which we called fractal menus. Our fractal menus will provide a menu design that can adapts to varying screen resolutions and will equally implement mechanism to optimize the menu's effectiveness and efficiency. In this thesis, we concentrate on implementing a menu layout and selection technique. Optimizing the menus selection time is out of the scope of the thesis. Nonetheless, we implement our fractal menu such that these selection times can be optimize using some options that affect our fractal menus configuration.

Practically, for the fractal menu to adapt to varying resolutions, it should have the capability of displaying variable amount of items in a variable structure such that the layout and shape could be adapted. Hence, the fractal menu design we propose should solve the constraints induced by varying resolution while implementing mechanisms to optimize the menu's usability.

3.2 Definition and fractal theory

To better understand fractal menu design, it is necessary to discuss fractal theory and how this concept can be use to design multi-platform innovative adaptative menus. This section is structured as follows: we begin by defining what fractals are and how they are created, then we look at some notation and representation that have be created to ease fractal creation. In the next section, we discuss our implementation, its limitations and possible improvements.

3.2.1 What are Fractals?

As said in Saupe *et al.* [427], *"there is a strong connection between chaos and fractal geometry, namely, as one follows the evolution of the states of a chaotic nonlinear system, it typically leaves a trace in its embedding space which has a very complex geometric structure: this trace is a fractal"*.

A fractal is a complex pattern that is self-similar across different scales. Fractals self similarity property means they show the same details at different scales. This self-similarity can be exact or statistical [427]. An example of an exact self-similar fractal is the snowflake curve devised by Helge von Koch in 1904, whose creation we will describe in a subsequent section. Fractals are not only created mathematically but also exist in nature.

Figure 3.3: Fern plant

Figure 3.4: Elm trees

Photos courtesy of Jonathan Wolfe https://fractalfoundation.org/

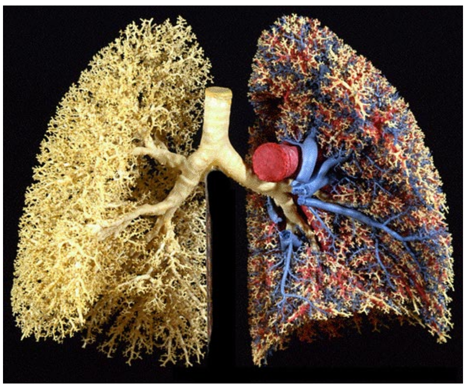


Figure 3.5: Human Lungs[1]

Figure 3.6: Human retina[2]

[1] Photo courtesy Ewald Weibel http://fractalfoundation.org/OFC/OFC-1-2.html#

[2] Photo courtesy of Paul van der Meer http://fractalfoundation.org/OFC/OFC-1-3.html

3.2.2 Mathematical fractals

Mathematical fractals[4] refers to a subset of Euclidean space for which the fractal dimensions[5] strictly exceeds the topological dimension[6]. Let's illustrate this by using the famous snowflake curve devised by Helge Von Koch[7]. Let E be an Euclidian space: Let's choose a finite line segment which can be compared to a compact interval of the real axis X.

1. Iteration 0: We draw a straight line from the origin to the extremity of the chosen segment say [0,1]

2. Iteration 1: On the central third of the interval, we will draw an equilateral triangle of dimension and length equal to one third of the initial interval, pointing upwards:

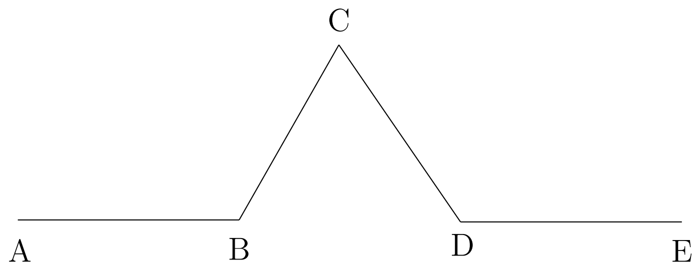


Figure 3.7: Iteration 1 of a Von Koch curve

3. Iteration 2: The same procedure is applied to the new segments thus determined, namely AB, BC, CD, DE

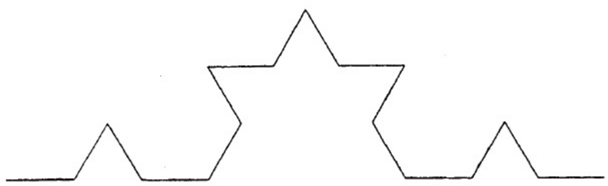


Figure 3.8: Iteration 2 of a Von Koch curve

[4] Also called geometric fractals.
[5] Also called Hausdorff dimension is a measure of roughness and/or chaos. E.g for a single point it is 1, for a square it is 2 and for a cube 3 [8].
[6] Also called the Lebesgue covering dimension is one of several different ways of defining the dimension of the space in a topologically invariant way [11].
[7] Helge Von Koch was a Swedish mathematician who gave his name to the famous fractal known as the Koch snowflake, one of the earliest fractal curves to be described.

4. If we apply the rules described above to the segments of iteration 2, we obtain the following curve for iteration 3 and applying the same rule to iteration 3 gives the curve below for iteration 4 :

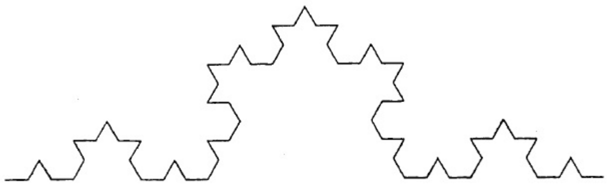


Figure 3.9: Iteration 3 of a Von Koch curve

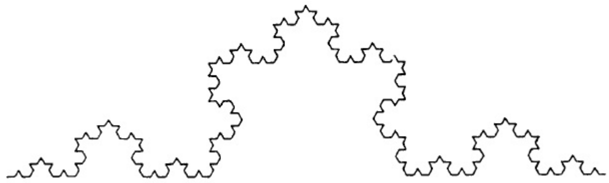


Figure 3.10: Iteration 4 of a Von Koch curve

3.2.3 Fractal properties

In this section, we look at some fractal properties that characterizes fractal shapes. In his paper *"How Long Is the Coast of Britain? Statistical Self-Similarity and Fractional Dimension"*[326], Mandelbrot[8] defines some properties that characterize fractals. We are going to define some of these properties and also consider additional properties that define fractals.

Self-similarity

One of the main properties of fractals is self-similarity. This means that each portion of the fractal can be considered a reduce-scale image of the whole fractal [326]. This property equally applies to a Mandelbrot set[9]. To appreciate fractal self-similarity, fractals have to be re-scaled using an anisotropic affine transformation[10].

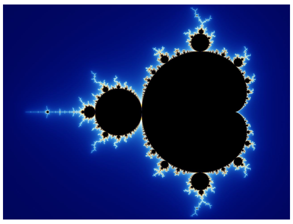


Figure 3.11: Mandelbrot Set [11]

[8] Benoit Mandelbrot was a Polish-born, French and American mathematician who studied fractal geometry

[9] Named Mandelbrot set by Adrien Douady as tribute to Benoit Mandelbrot, represents a set of complex number c for which the function $f_c(z) = z^2 + c$ does not diverge when iterated from z = 0. It is connected to a Julia set and related Julia set produce similarly complex fractal shapes [9].

[10] In Euclidean geometry, an affine transformation, or an affinity, is a geometric transformation that preserves lines and parallelism.

Fractal dimension

1. **Fractal dimension:** Fractals are generally characterize by a fractal dimension which is a ratio providing a statistical index of complexity comparing how detail in a fractal pattern changes with the scale at which it is measured [10]. In other words, it defines how complex a self-similar object is.

2. **Fractal set:** According to Mandelbrot is considered a fractal set, every set of E that satisfies the relation D > D_T i.e its fractal dimension is strictly greater than its topological dimension.

Mathematically a fractal dimension is defined as: [235]:

$$\text{Fractal dimension} = \frac{log(\text{Number of self similar pieces})}{log(\text{magnification factor})}$$

For example, the Von Koch Curve we saw above will have a fractal dimension of $\frac{log(4)}{log(3)} = 1.26$. The number of self similar pieces it obtain at the generator level by counting the pieces at iteration 1 (we know that the number of line segments in the Koch curve is 4). The magnification factor as each section of the generator is 1/3 of the unit length [175].

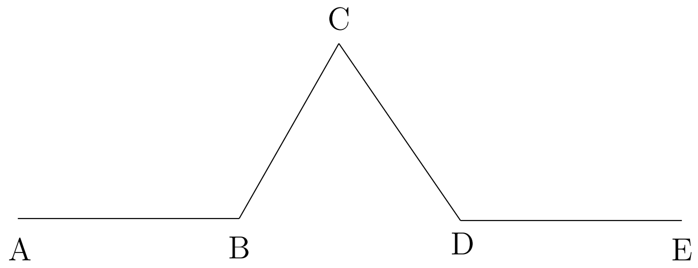


Figure 3.12: Iteration 1 of a Von Koch curve (Generator)

We can therefore conclude that, a fractal object's complexity is directly proportional to the fractal object's dimension. A fractal dimension is always between 1 and 2. For example, The Cantor Square Fractal has a fractal dimension of 1.26 and the Sierpinski Carpet Fractal has a fractal dimension of 1.89.

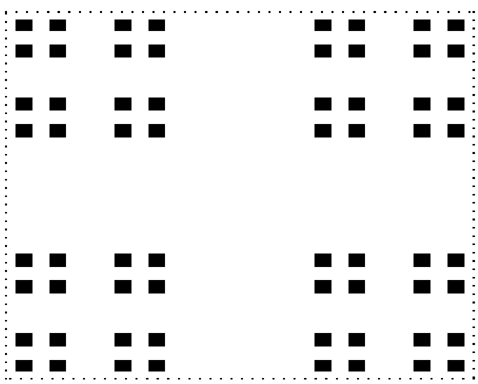


Figure 3.13: Cantor Square Fractal[4]

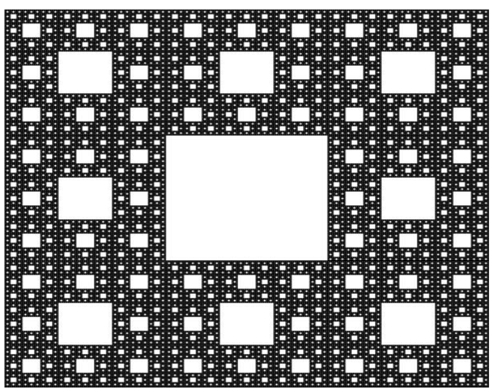


Figure 3.14: Sierpinski Carpet[5]

[4] Source : https://www.robertdickau.com/cantordust.png
[5] Source : https://en.wikipedia.org/wiki/Sierpinski_carpet

3.2.4 Fractal creation

Several techniques have being devised over the years to facilitate fractal generation. We will describe their generation using three techniques: Repeated removal technique, Iterated Function System and L-System (Lindenmayer).

Repeated Removal Technique (RRT)

The technique consists in the repeated removal of one or more parts of the initial shape. This approach is very common to the rep-tile[12] notion in tessellation geometry. Let's describe this technique via the generation of the Sierpinski Triangle [175]:

1. Iteration 0: We start with a single black filled triangle.
2. Iteration 1: Remove the middle triangle. This results in 3 black triangles surrounding a central white triangle.
3. Iteration 2: We do the same process on the 3 remaining black triangles.
4. Iteration 3: Remove the middle triangle. Resulting in 9 smaller black triangles.
5. Iteration 4: We continue the process on the smaller black triangles resulting in 27 smaller black triangles.
6. By the firth iteration, we obtain 81 tiny black triangles. Each iteration triples the triangles and the number of triangles can be expressed as 3^n where n is the iteration number. Geometric fractals don't need to end we can go on with the iterations.

Figure 3.15: Sierpinski Triangle Evolution [175]

The RRT can be extended to other shapes. For example, using a square as initial shape and cutting its recursively into 9 congruent sub-squares in 3-by-3 grid results in the famous Sierpinski Carpet by the firth iteration. Another famous fractal form called *the Menger Sponge* is indeed a cubic square Sierpinski Carpet.

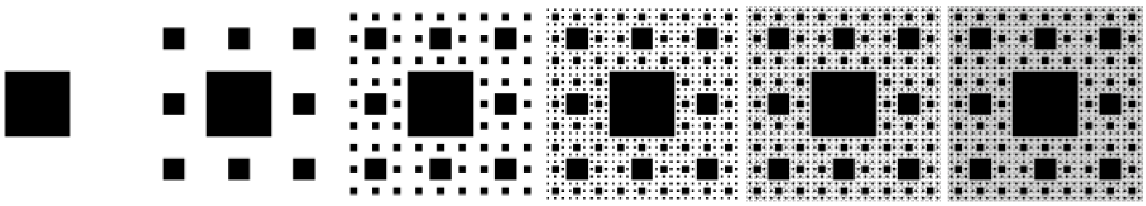


Figure 3.16: Sierpinski Square Evolution[6]

[12] Rep-tile references a shape that can be divided into smaller copies of itself.
[6] Source : `https://en.wikipedia.org/wiki/Sierpinski_carpet`

Iterated Function System (IFS)

Another way of generating geometric fractals is using a repeated substitution technique called Iterated Function System(IFS). It composes of a union of several copies of itself, each copy being transformed by a function. The Sierpinski triangle, we described above can also be created using the IFS technique. An IFS is a finite set of contraction mappings[13]. Formally, an IFS is defined as follows:

Let $T_1, T_2, ..., T_n (n \geq 2)$ be a finite collection of contraction mappings defined on a complete metric space (X, d). Suppose that $\alpha_1, \alpha_2, ..., \alpha_n$ be their respective contraction factors. Then the system $(X; T_i : i = 1, 2, ..., n)$ is called a IFS with contraction factor $\alpha = max(\alpha_1, \alpha_2, ..., \alpha_n)$ [323].

More practically, a fractal can be generated from an IFS by applying its set of functions to an initial image (a set of points in 2d space) recursively. We begin the process with a generator image at iteration 0. For subsequent iterations, we replace each segment of the image by the generator and rotating when necessary so it fits. For example, the following Dragon curve generated via an IFS.

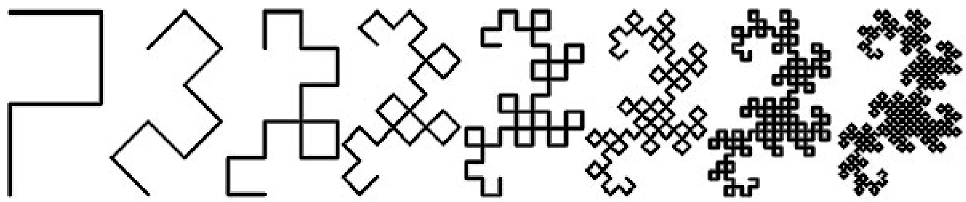


Figure 3.17: IFS Generation Of Dragon Curve

Lindemayer notation (L-System)

The Lindemayer notation was developed by a Hungarian botanist Aristid Lindenmayer in 1968 at the University of Utrecht. This grammar base system was destined to model and stimulate plant growth patterns. The aim was to use this approach to provide a formal description of the development of simple multi cellular organisms while illustrating the neighborhood relationships between plant cells. The system was later extended to describe higher plants and complex branching structures [12].

1. **Composition**

   An L-system is defined formally by a quadruplet (V, ω, P, C)[316, 12] where:

   - V is an alphabet[14], used to define the valid list of symbols that can be handled.
   - ω is an axiom, a non-empty word of the alphabet ($\omega \subset V^+$). The axiom describes the initial state of the system (i.e at iteration 0) For example: "F", "FX", "F — F — F"

[13] Functions that map a set of points closer to one another.
[14] An arbitrary set of symbols.

- P is a set of production rules, each mapping a symbol to a word: $P \subset V$ x V^*. That is, a rule is a transform that takes a symbol as parameter. The rule are applied successively to each symbol of the axiom over and over at each iteration. Symbols not present in P are assumed to be mapped to themselves. For example consider the rule "A $\rightarrow$ AB": whenever an "A" symbol is found in the current state, it is replaced with "AB."
- C represents constants. These are symbols that cannot be replaced. They are usually the following: {!, [,], +, —} and are generally used for graphical instructions. We will see more details on this when we see the turtle interpretation

2. **Example**

   As example, let's consider an L-System with the following composition:

   - Alphabet: A, B, C
   - Axiom: AC
   - Rules:

     (A $\rightarrow$ AB),
     (B $\rightarrow$ AC),
     (C $\rightarrow$ AA)
   - Constants: none

   The system starts with "AC" which is the axiom (iteration 0) and has three production rules, one for A, one for B and one for C. Let's now analyse the first four iterations:

   - Iteration 0 : AC
   - Iteration 1 : AB AA
   - Iteration 2 : AB AC AB AB
   - Iteration 3 : AB AC AB AA AB AC AB AC
   - Iteration 4 : AB AC AB AA AB AC AB AB AB AC AB AA AB AC AB AA

   The recursive nature of the L-system rules leads to self-similarity and thereby fractal-like forms.

3. **The Turtle Interpretation**

   Recursive L-systems, like the one described above, often produce intricately complex patterns that are self-similar across multiple scales. The strings generated by L-systems are lists of symbols and to produce a visible L-system, we use a representation based on the Turtle Interpretation.

   Imagine a turtle in a Cartesian plan to which you could issue a set of commands: **move forward, turn left, turn right, draw a line, etc.**. We generally associate a set of symbol to these commands. The following dictionary is commonly used in Turtle representation for L-Systems [12, 470]:

- F : Move forward (draw a line forward along the turtle's heading).
- G : Move forward (without drawing a line).
- $+$: Turn left (of a certain angle α).
- $-$: Turn right (of a certain angle α).
- & : Pitch down (by an angle α).
- ^: Pitch up (by an angle α).
- $<$: Roll left (by an angle α).
- $>$: Roll right (by an angle α).
- | : Turn around (by 180°).
- [: Push current state onto a push down stack.
-] : Pop current state from the stack and make it current state.

4. **Example using turtle interpretation**

   Consider the string

   - F — F — F — F
   - with a turning angle of 90°
   - starting facing north

   This will result in the following in turtle interpretation

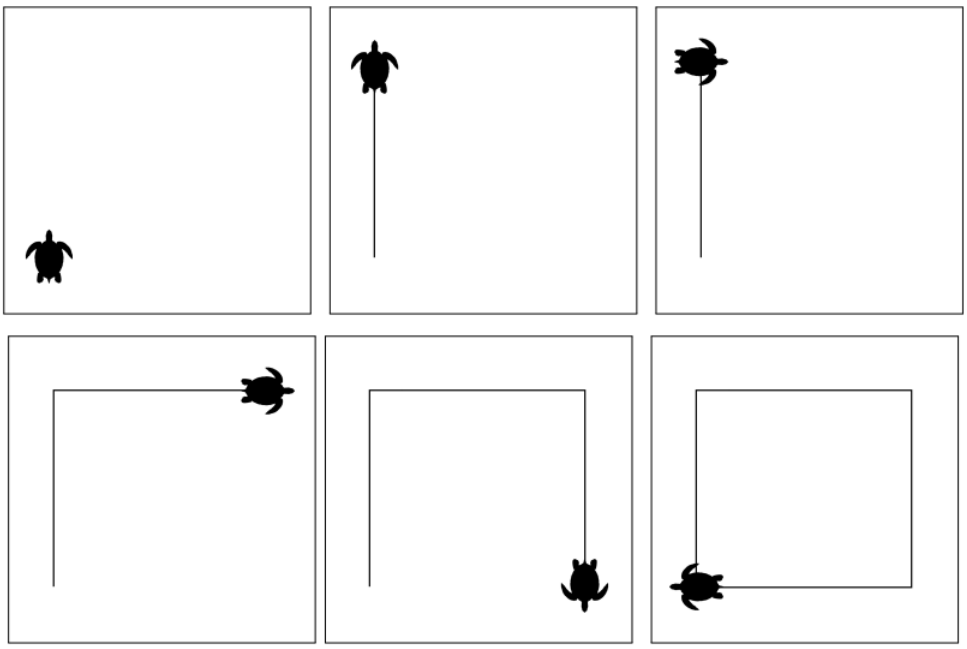


Figure 3.18: Turtle Interpretation for example[15](string F — F — F — F with 90° TA)

[14] The following tool can be use to emulate https://demo.hurna.io/#path=fractal/d0l_system

[15] Source: https://medium.com/analytics-vidhya/generating-fractals-using-lindenmayer-systems-6214dddbe223

Using the turtle interpretation, we can now adapt the composition of our L-system to include instructions necessary for drawing a visual interpretation. As example, consider the following L-system composition we will use to produce the "Koch Triangle" seen previously:

- Alphabet: F
- Axiom: F
- Rules:

  F → F+F — F+F
- Constants: none
- Turning angle: 60°

Let us interpret this using the turtle interpretation we have just described.

- Iteration 0: Is the axiom ‘F’: Which means **Draw forward**
- Iteration 1: We transform ‘F’ using the production rule F → F+F – F+F which is interpreted as follows:

  “F”: Draw forward
  “+”: Turn left by 60°
  “F”: Draw forward
  “–“: Turn right by 120° (x2)
  “F”: Draw forward
  “+”: Turn left by 60°
  “F”: Draw forward
- Iteration 2: We apply the production rule on each segment, the same motif is repeated at smaller scale.
- We obtain similar curves as those above. Below is the ’Von Koch Curve’ fractal at depth 4, but we could continue indefinitely to refine the fractal.

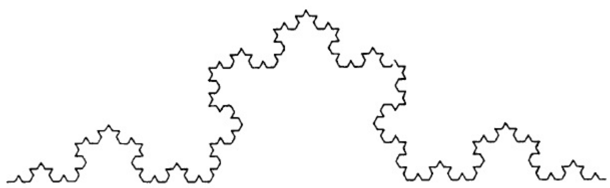


Figure 3.19: ’Von Koch Curve’ fractal at depth 4

Different types of L-Systems exist we used the simplest form. For more information on the other types see [12].

3.3 Implementation of a Fractal Menu

3.3.1 Introduction and design choices

In this section, we motivate the design choice considered for our fractal menus and we explain its implementation in details. We developed a fractal menu that creates an adaptable menu by altering its layout and size. As mentioned in the introduction of this chapter, this design will surely affect the resulting menu's usability but we don't go into details on that since evaluating the resulting usability of this menu is out of scope of this thesis. But we implement mechanism that should optimize its usability.

We define a fractal menu as a responsive adaptative multi-platform context menu that can adapt to varying screen resolutions and where selection depends on the user's preference and usage. In this thesis, we proposed two designs of fractal menus namely a rectangle and hexagonal fractal menus. Our fractal menu will be designed to be context-sensitive, and adaptative by updating the menu to reflect the user's preference and usage. Hence, the menu will be user adaptable so as to increase its effectiveness, efficiency and subjective satisfaction.

The fractal menus are designed such that they have a low fractal dimension so as reduce the menus complexity. For example, iteration 1 of our rectangular fractal has the following turtle interpretation:

A. **Rectangular Fractal Menus** : Using the turtle interpretation :

- Alphabet : FG
- Axiom : GGGG-GGGG-GGGG-GGGF
- Rules : F = G++GG+GG+GG+G+GG+G+G+GG
- Constants : none
- Turning angle : 90°
- Starting angle : 90°

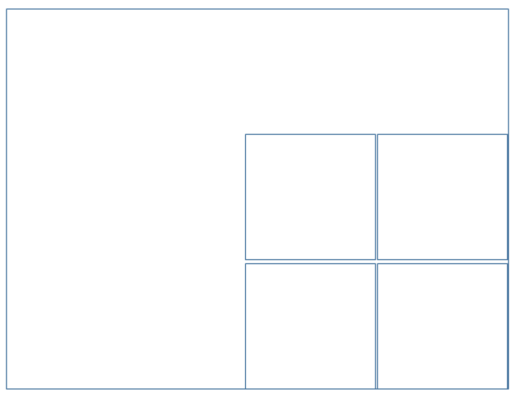


Figure 3.20: Fractal Shape Considered For Our Rectangular Iteration 1

This gives it a fractal dimension close to 1. For subsequent iterations, each small rectangle will be replaced by a reduced version of the global shape.

Programming environment

For the implementation of the two fractal menus, we chose as programming language Java. Our choice of Java as implementation language was motivated by the following reasons:

- The large online support ecosystem.
- Its multi-platform support nature.
- Our already existing familiarity with the language.

Program structure

They class diagrams for our implementation can be found in appendix B.1. Before creating our rectangular fractal, we first designed a parser class. The aim of the parser class is to parse a menu structure given in an XML format into an object that will be interpreted by the fractal generator we will create. Our parser distinguishes two types of menu items namely: final and non final menu items representing respectively a final menu item and sub menu item. To create our fractal menu, we used recursion and the IFS technique. Hence being able to distinguish between final and non final menu items helps us create an exit conditions for the recursion.

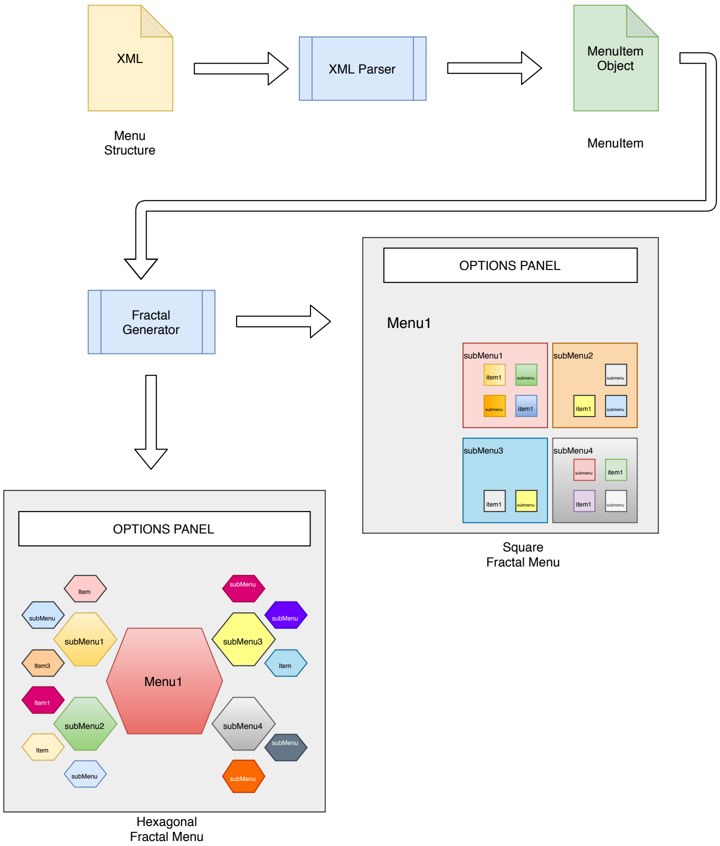


Figure 3.21: Fractal Menu Generator Summary Steps.

We created a MenuItem Object to be returned by our parser and whose structure is given by the class diagram below. Non Final MenuItem (sub menu) contain a list of final and/or non final MenuItem. This list is empty for final menu items. All MenuItem have a parent MenuItem. This will help us navigate between screens. The home page's parent will be itself. The other attributes are self-explanatory. Our rectangular and hexagonal fractal menu generator will be programmed to generate the menu using this object.

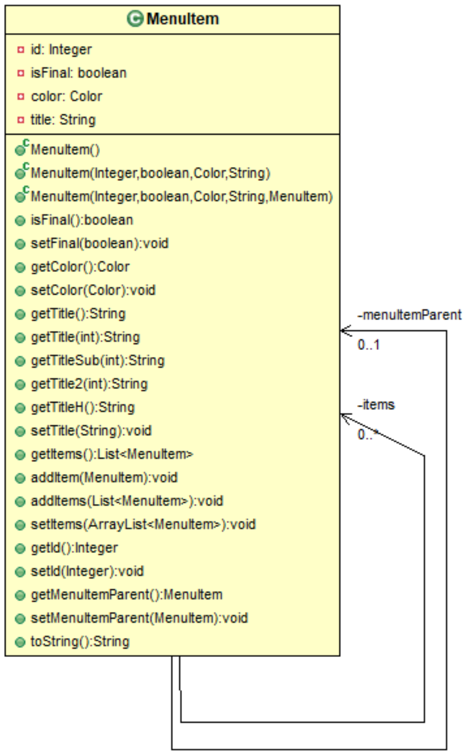


Figure 3.22: Class diagram of a MenuItem

Our parser class can be described using the following pseudo code:

Algorithm 1 : Menu Structure Parser

Result: Parse menu structure into MenuItem Object
MenuItem *items* = new MenuItem();
foreach *menuItem in List* **do**
  int $id = generateId()$;
  Color $color = generateColor()$;
  MenuItem mI = new MenuItem(id, color, menuItem.title, ...);
  Add created MenuItem to corresponding parent in **items**;
end
return **items**

Simplified Pseudo-Code of our Fractal Generator

Below we describe a simplified pseudo-code of our fractal generator.

Algorithm 2 : Generate Window with a Configuration panel and Fractal panel

```
Class FractalGenerator
    FractalGenerator():
        setFrameSettings();
        setFrameDimension();
        size ⟵ initSize();
        loadConfiguration();        ▷ Load menuStruct, menuType, depth, breath, dimension, etc.
        MenuItem menuItem = parseMenuStructure();
        if menuType == Rectangular then
            mainMenu = MenuS(menuItem, size);   ▷ MenuS is the rectangular fractal generator.
            options = MenuO();                  ▷ MenuO is the configuration panel generator.
            frame.add(options);
            frame.add(mainMenu);
        else
            mainMenu = MenuH(menuItem, size);   ▷ MenuH is the hexagonal fractal generator.
            options = MenuO();                  ▷ MenuO is the configuration panel generator.
            options.deactivate()                ▷ Deactivate options not implemented in MenuH.
            frame.add(options);
            frame.add(mainMenu);
        end
        frame.display();

    Function update():
        saveConfiguration();
        execute(mainMenu.getMenuItem());
    End Function

    Function execute(menuItem):         ▷ update display with new panel of menuItem.
        loadConfiguration();
        mainMenu = GeneratePanel(menuItem);
        updateFrame();
    End Function
```

3.3.2 Rectangular Fractal V1

As described in the introduction, we first create a first fractal menu called rectangular fractal. To maintain its fractal dimension low (reduce the menu's complexity), we limit the number of items per menu as well as its view depth. In this initial version, items are limited to 4 items per view screen. According to Millers assertion [343], seven items plus or - two was the optimum number of items an average person can process. Even though this limit has being contested, it is still widely used in the design of human computer interfaces [73]. In a later version, we increase this limit to a max of 6 which we believe is an adequate value for our menu to adhere to Millers assertion. The generator is initialized by the Fractal generator using the MenuItem from the parser and a size value based on the screen's resolution. The following simplified pseudo code describes its generation:

Algorithm 3 : Generate a Rectangular Fractal Panel

```
Class MenuS
    MenuS(mi, s):          ▷ mi = MenuItem from parser, s = size based on resolution.
        panelColor ⟵ mi.getColor();
        panelText ⟵ mi.getText();
        setPanelDimension(s);
        nextSize ⟵ s/2;
        if mi.isFinal() == false then
            for MenuItem m in mi.getMenuItems() do
                addSubPanel(MenuS(m,nextSize));
            end
        else
            addSubPanel(MenuS(m,nextSize));
        end
        assembleSubPanels();
        setColorAndBorderLayout();
        addListeners();
```

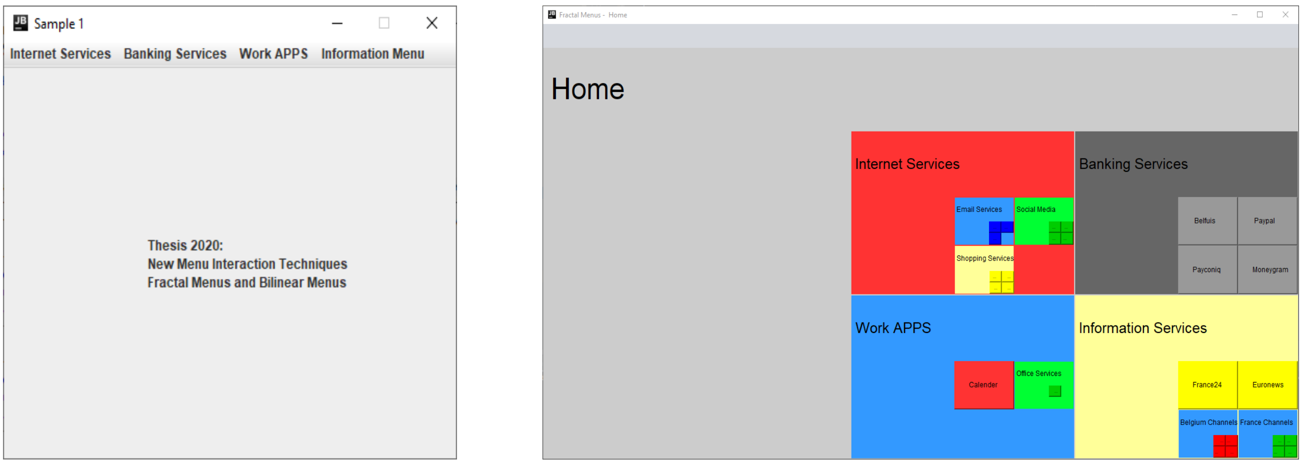


Figure 3.23: Rectangular Fractal Menu V1

Item selection

We use mouse event mainly : **left-mouse click** (to zoom into the clicked menu item i.e. display its children) and **right-mouse click** (to zoom out of the clicked item i.e. display its parent menu.). On a click event, we get the MenuItem of the clicked item, if its a right mouse click and we get its parent MenuItem if its a left mouse click. The MenuItem is loaded for display via our FractalGenerator. Depending on the device, other item selection techniques could be considered and implemented e.g. touch selection, eye selection etc.

Algorithm 4 : MenuItemPanel listener

Function `listener`(*event*):
  MenuItem *chosen* ⟵ *event.getSource().getMenuItem()*;
  MenuItem *parent* ⟵ *chosen.getParent()*;
  if *event.isLeftMouseButton* **then**
    FractalGenerator.execute(chosen);
  else
    FractalGenerator.execute(parent);
End Function

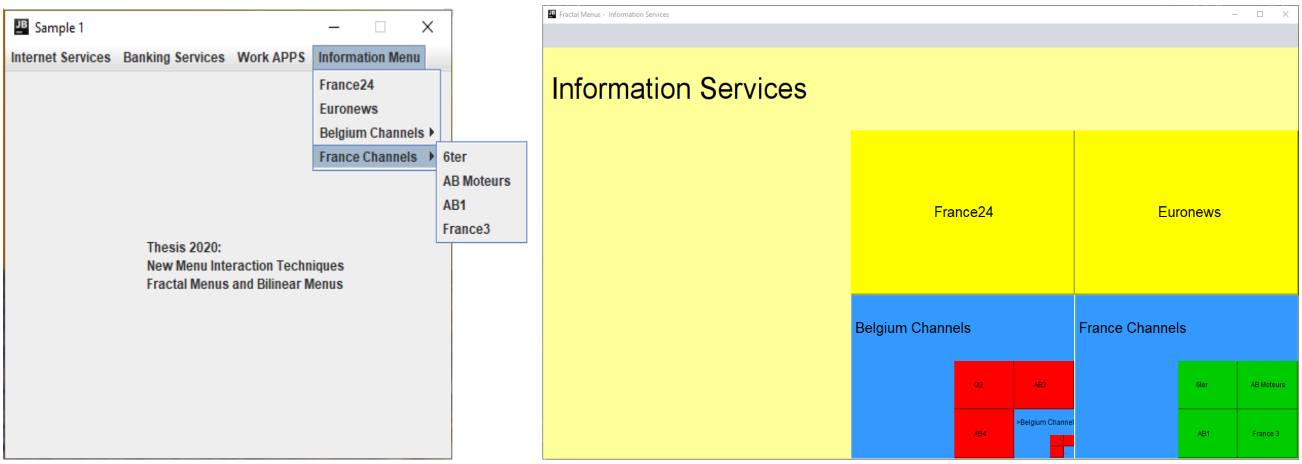


Figure 3.24: Rectangular Fractal Menu V1 : Item Selection

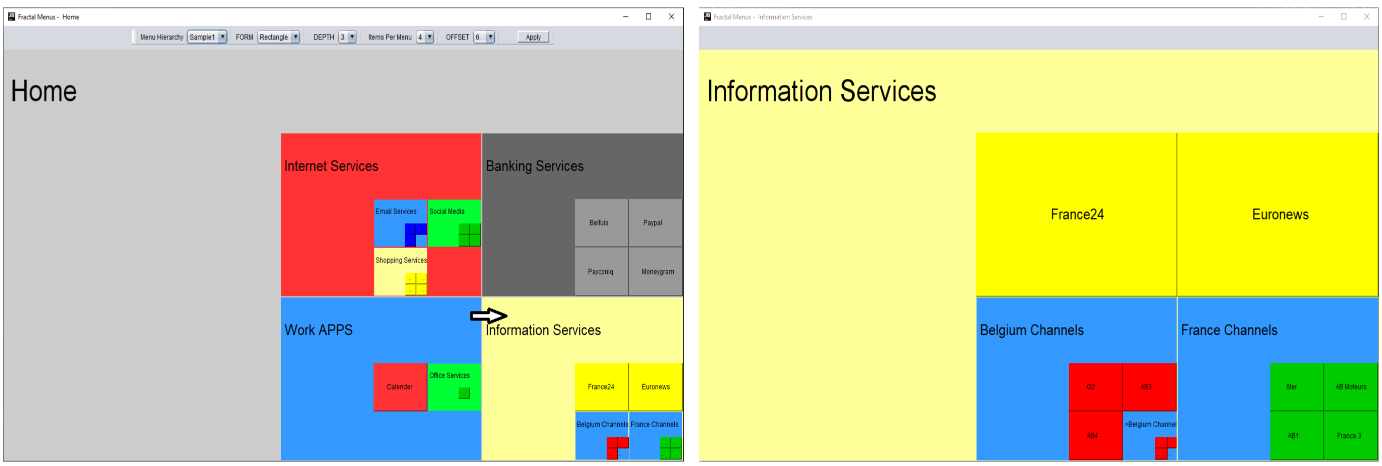


Figure 3.25: Rectangular Fractal Menu V1 : Item Selection Steps

Deeper items or sub-menus can be accessed by sequentially selecting the sub-menus leading to them (as illustrated in 3.3.2). **Menus or sub-menus with more than 4 items** will have three of the items shown and the rest will be on a subsequent screen accessible via the fourth menu item bloc. To access a given item say AVS in Belgium Channels, we can proceed by sequentially choosing the parent sub-menus items leading to it.

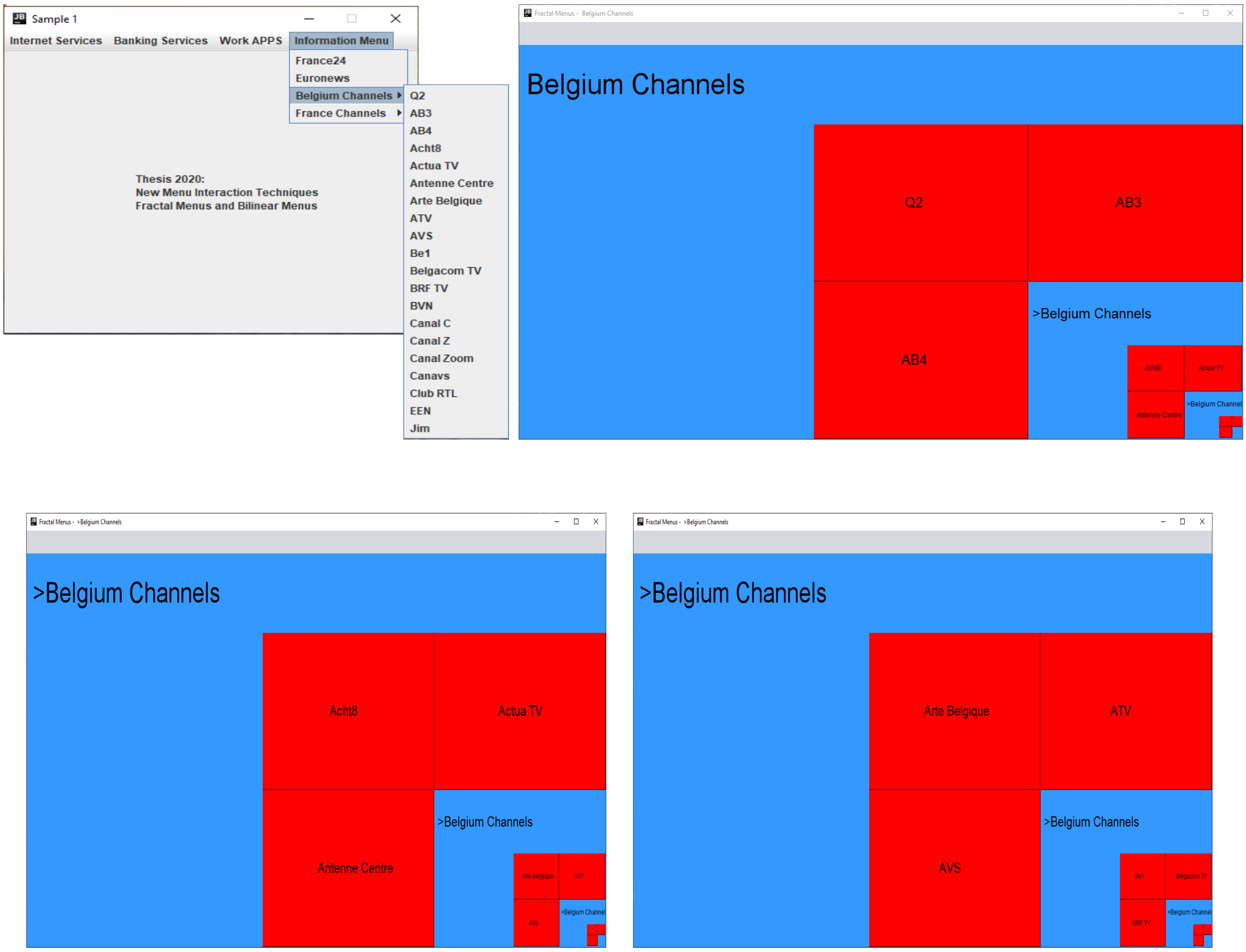


Figure 3.26: Rectangular Fractal Menu V1 : Selection and large submenu 1

3.3.3 Rectangular Fractal V2

For **version 2** of our rectangular fractal, we did several optimizations namely: we reorganized the display order of menu items for better viewing experience, we increased the max items per menu to six as a good compromise so as to adhere to Millers assertion [343], we optimized item selection, we added an options field which could be used by the user to tweak some properties of the fractal menu. We equally added a debug console to display which item is selected. It's class diagram is given in appendix B. The rectangular fractal panel generator described in v1 was modified accordingly to incorporate these optimizations. Each of these optimisations are described in details in the following points:

1. **Reorganized the display order of menu items**

   We reorganized the display order of menu items to optimize the view space and for better viewing experience.

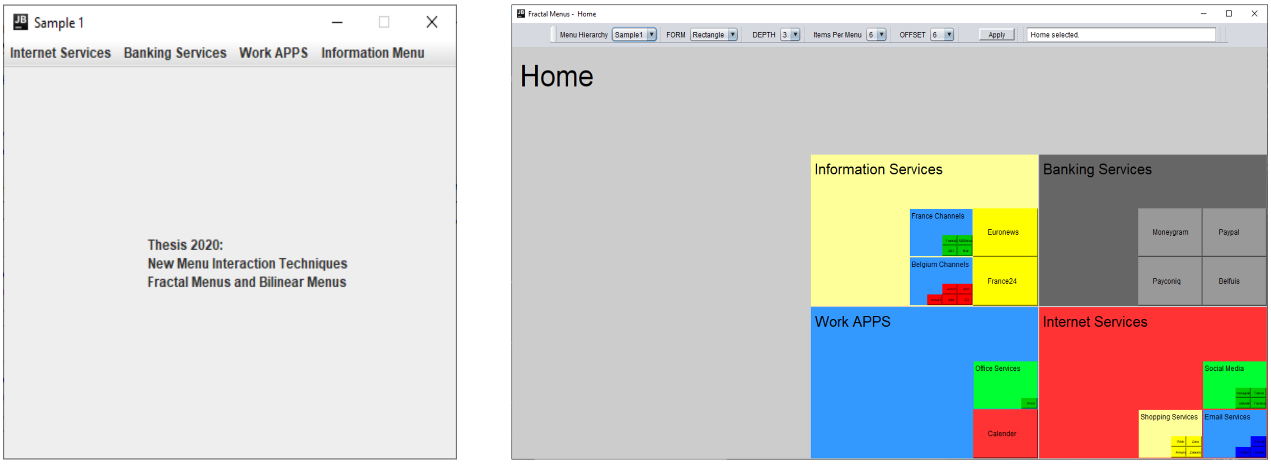


Figure 3.27: Rectangular Fractal Menu V2

2. **Increased Items per view screen**

   We increased the items per menu from 4 to 6, as such menu with more than 6 items will have 5 shown and the rest accessible via the sixth menu item bloc and so on.

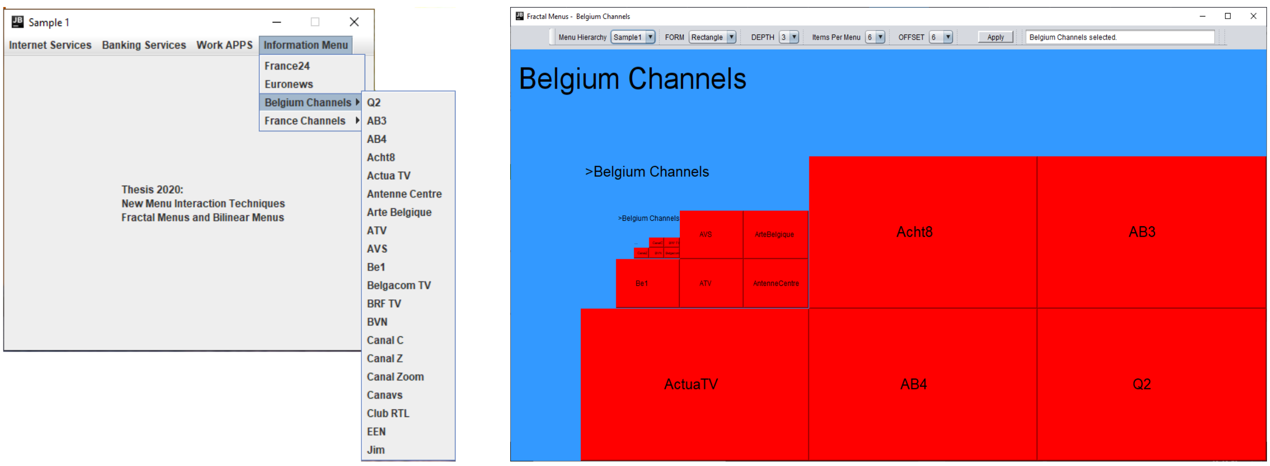
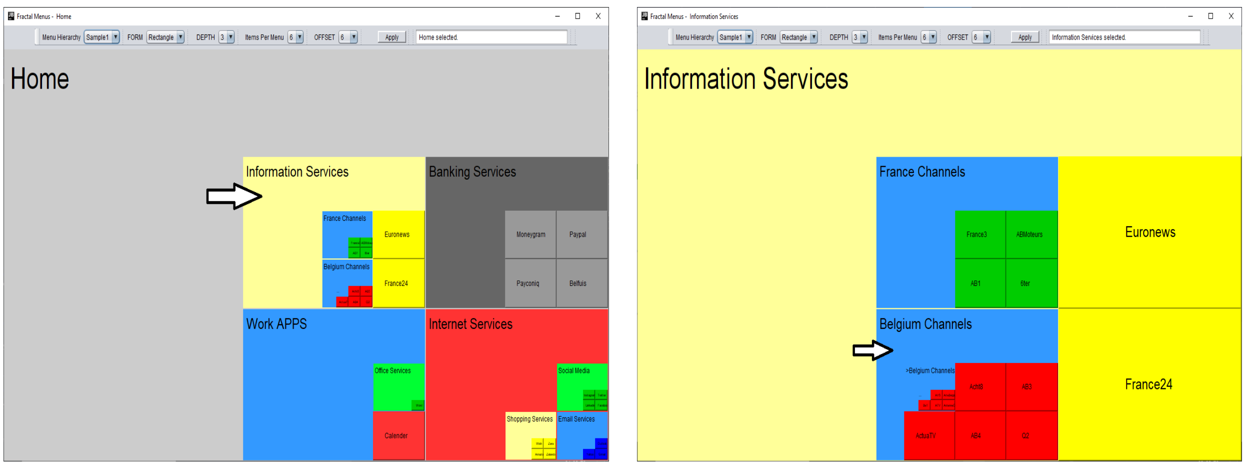


Figure 3.28: Rectangular Fractal Menu V2 : Increased Item Per Menu

As described in v1, deeper items can be access by sequentially selecting the sub-menu leading to them. Increasing the maximum items per menu increases the selection steps for deeper items (which increases the selection time). In point 3., we present an optimized item selection technique we implemented to reduce these selection steps thereby reducing the selection time it induced. Taking back our example with Belgium Channels, to access say AB4, we can proceed sequentially as follows:

Figure 3.29: Rectangular Fractal Menu V2 : Sequential Item Selection - Example 1

For a second example, say we wish to open ClubRTL, sequentially we proceed as follows:



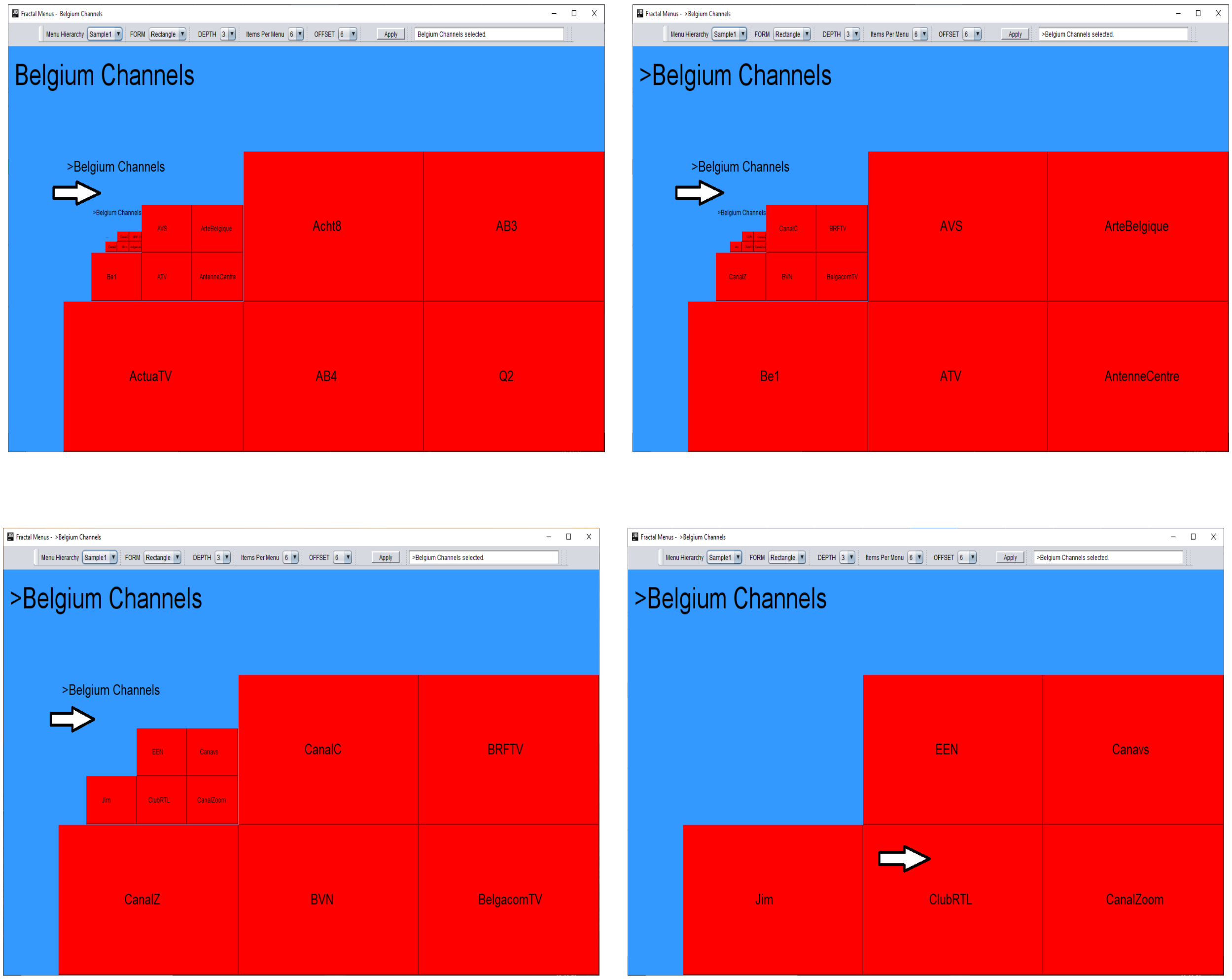


Figure 3.30: Rectangular Fractal Menu V2 : Sequential Item Selection - Example 2

As explained in V1, the menu shouldn't have too many items per view screen especially in devices with smaller resolutions. With smaller resolution screens, too much items per view screen may render the menu unusable. Also, menus with too much items may not adhere to Miller's assertion [343]. We present in point 4., an option we included in V2 to allow the user to tweak the maximum number of items per view screen.

3. **Optimized Item Selection**

   In V2 of our rectangular fractal, deeper menu items in the viewed screen can be directly selected or zoomed in without necessarily selecting their parent component sequentially as we saw in V1 and point 2. For example, if we wish to select Gmail, we can proceed in two ways : one way is sequentially which results in more steps and hence increases selection time or we can directly click on Gmail. Deeper items whether sub-menus or final items in the view screen can be selected and zoomed in without necessarily sequentially selecting sub-menus leading to it. Taking back our example:

   To select Gmail, sequentially we will proceed as follows:

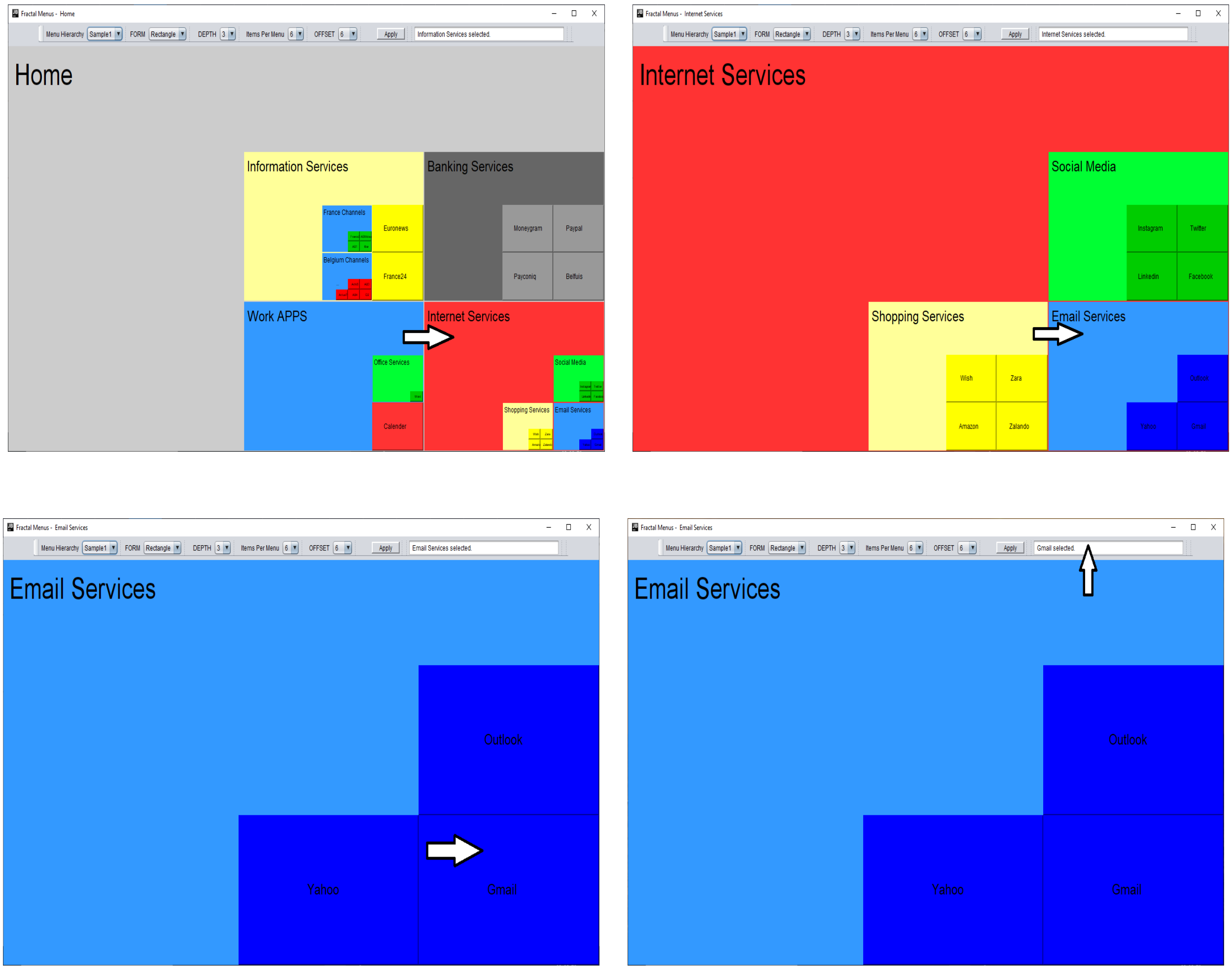


Figure 3.31: Rectangular Fractal Menu V2 : Sequential Item Selection - Example 3

Using the optimized item selection, we will proceed as follows:

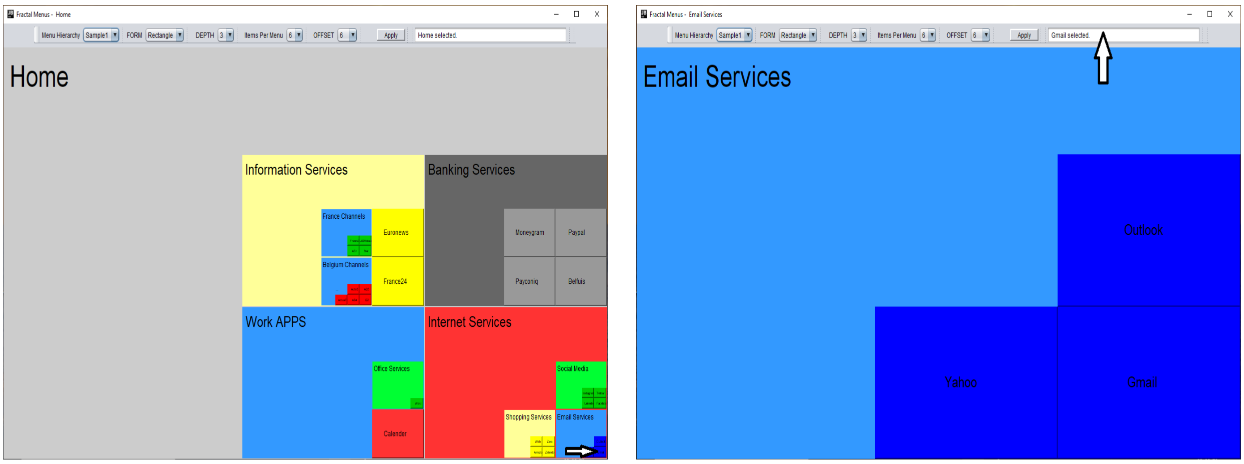


Figure 3.32: Rectangular Fractal Menu V2 : Optimized Item Selection - Example

From the above example, we see that the optimize selection reduces the selection steps and therefore the selection time needed to select a given menu item. Deeper menus items may not be very readable for small resolution. We discuss in the next point, an option we implemented so the user can limit the viewed depth.

4. **View Depth**
Via the added options panel, the user can alter the view depth. By default, this view depth is set to 3. From our experimentation, a view depth of 3 is a good compromise. The user can alter the view depth to a minimum of 1 and a maximum 6. Depending on the device's resolution, some view depth may be unpractical as items at those level wouldn't be very visible/readable. E.g for a little 7 inch screen, items a depth greater than 3 are not very visible and their parent component will have to be zoomed in for them to be visible and selectable. Another point to note is that, reducing the view depth renders unselectable items that were at a visible view depth and could have being selected directly as described in 3. Hence with a reduced view depth, their parent component will have to be selected first for them to be visible and selectable. Reducing the view depth therefore induces addition selection steps which increases the selection time. This option once applied doesn't require rebuilding the whole menu, the user sees the effect directly at the view level he is.

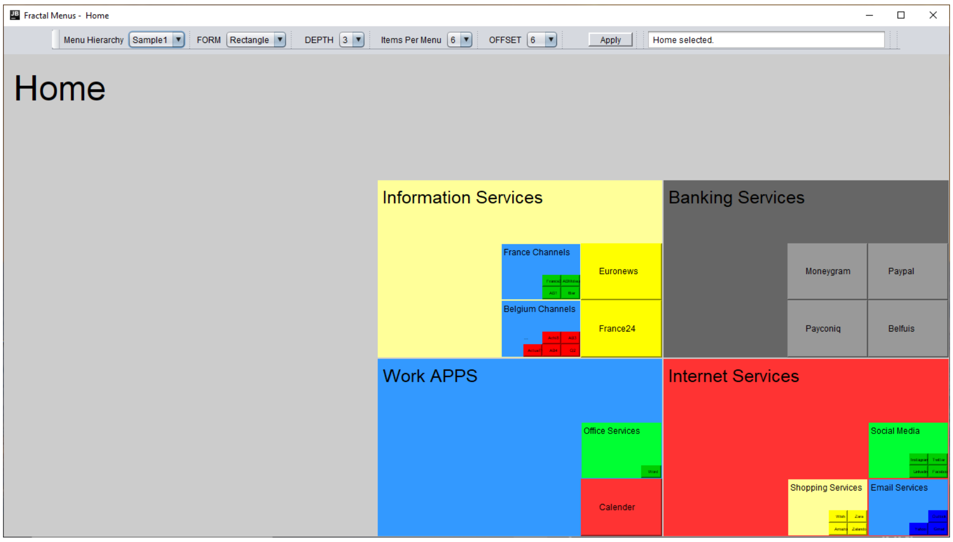


Figure 3.33: Rectangular Fractal Menu : Depth 3

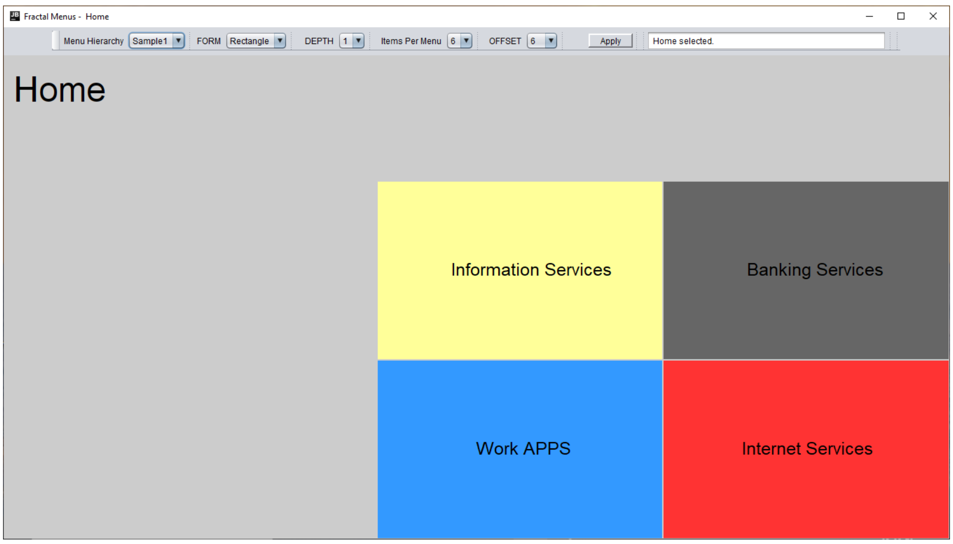


Figure 3.34: Rectangular Fractal Menu : Depth 1

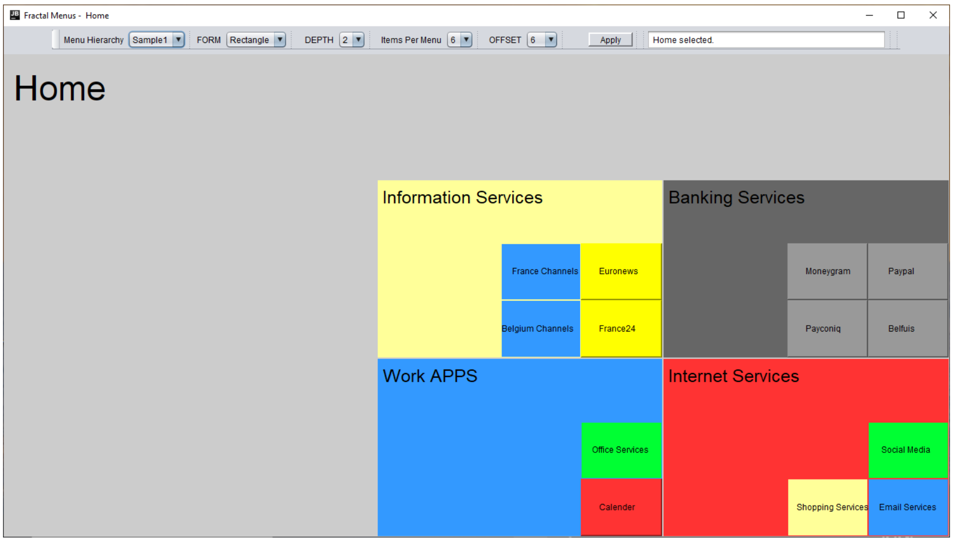


Figure 3.35: Rectangular Fractal Menu : Depth 2

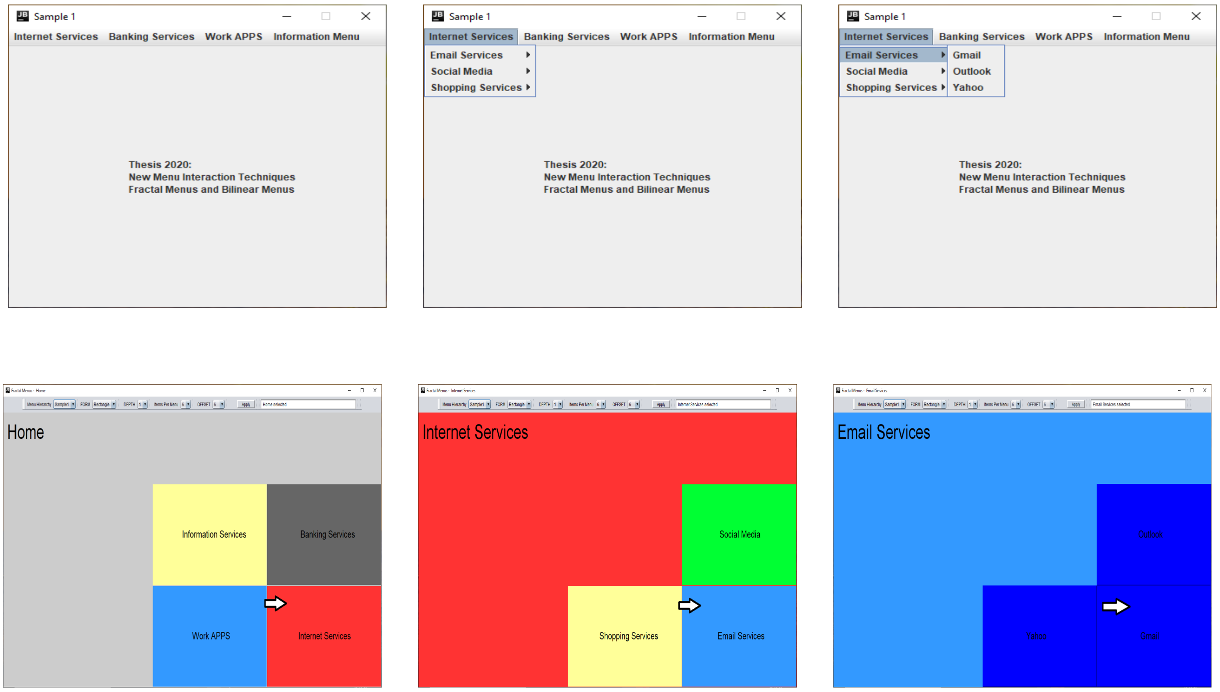


Figure 3.36: Rectangular Fractal Menu Depth 1 : Selection

View Breath (Items per view screen)

This option lets the user define the maximum number of menu items per view screen. Altering the breadth also increases the depth as items removed from the view screen due to this limitation will have to be kept in a child sub-menus of the viewed screen. The following screen shot illustrates the result of using this option: Such as option can be ideal in small resolutions situation where we have many menu items to display but limited screen space. Once this option is applied, the whole menu is rebuilt to suit the setting. For example, choosing 3 items max per menu with a view depth of 2 produces the following fractal menu. Say, we wish to select Moneygram, we proceed as follows using the optimized menu selection.

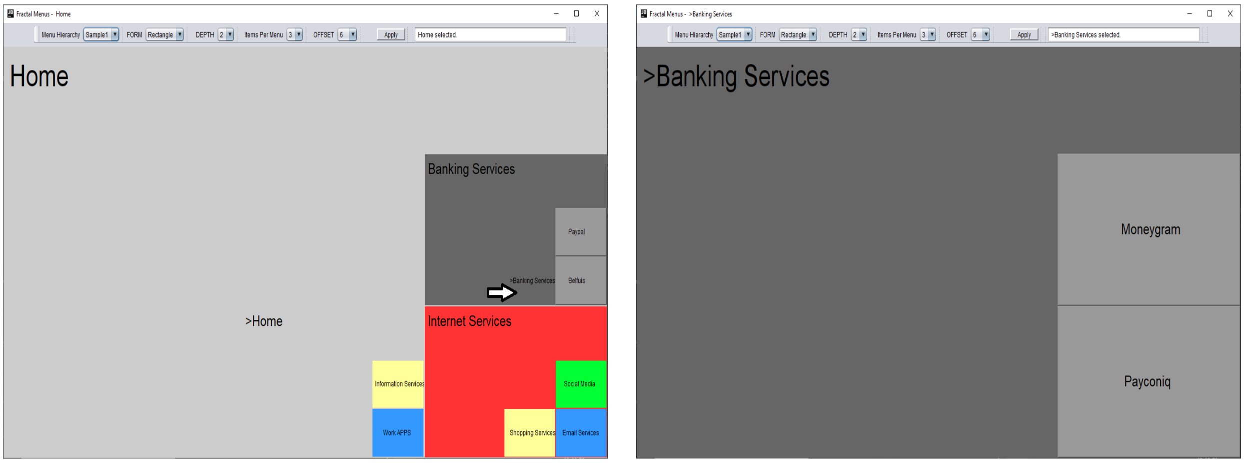


Figure 3.37: Rectangular Fractal Menu V2 : View Breath Modification - Example

Using the previous settings of 6 items per view with a view depth of 2 would have required a single step.

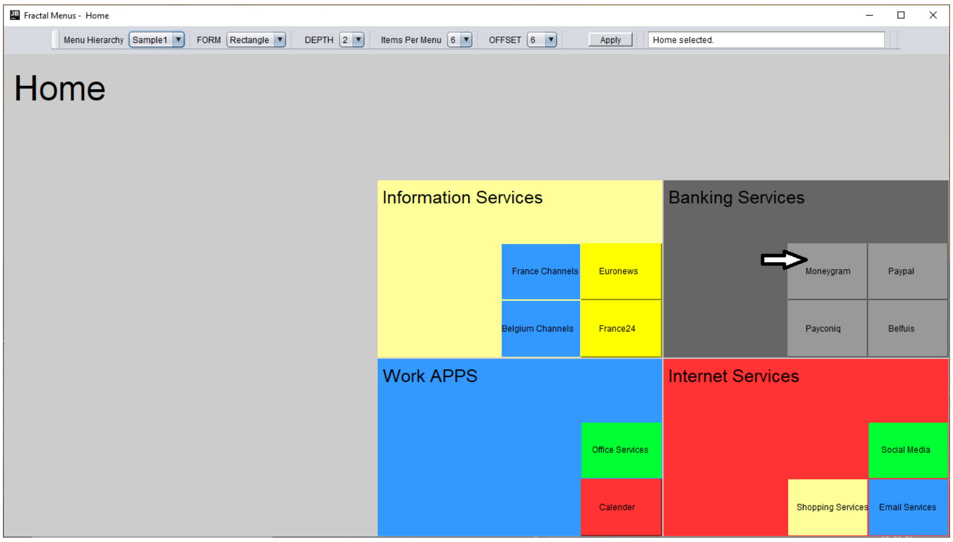


Figure 3.38: Rectangular Fractal Menu V2 : Breath 6 selection example

As, we can see increasing the breath (number of items per menu) increases the selection steps which increasing the selection time. Hence, if reducing the breath is necessary, it will induce an increase in selection time. As said previously, we propose others techniques of reducing these selection steps in further studies.

5. **The offset**

   Simply serves to move the title of menu items by a given number of spaces.

6. **The Menu Hierarchy**

   For loading other menu structures.

3.3.4 Rectangular Fractal V3

In V3 of the rectangular fractal, we added to the options panel, a configuration for the right click. In V2 of the rectangular fractal, the right click behavior was to load the parent menu of the viewed menu item. We notice that this was not very practical especially when using the optimized selection. We therefore added an option that lets the user tweak the behavior to either return to the parent menu of the viewed item or return to the previously viewed item.

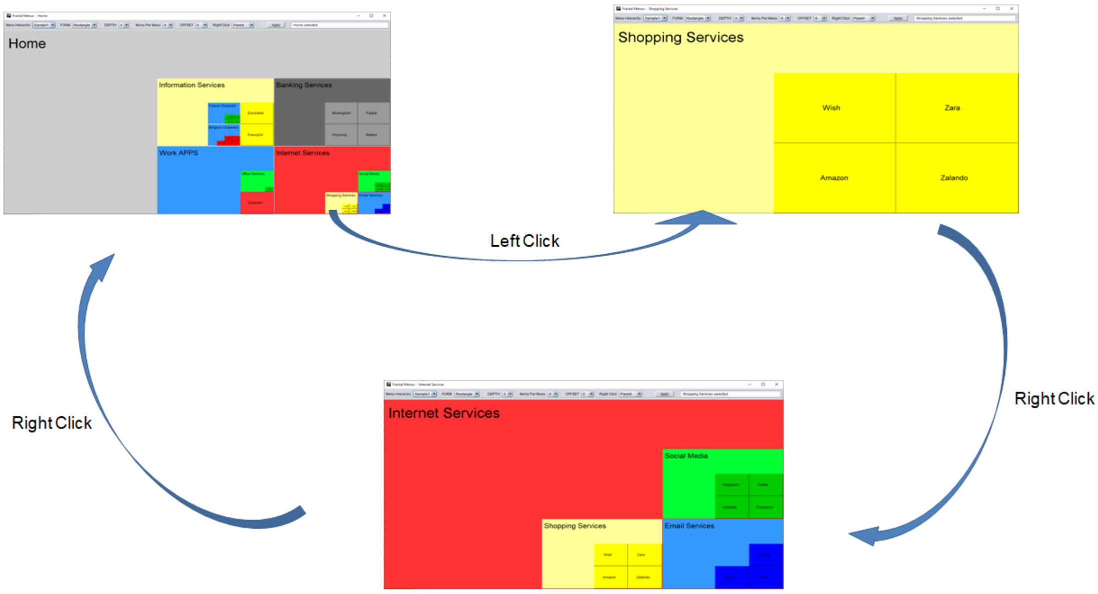


Figure 3.39: Illustration of right click configuration : Parent Menu Mode

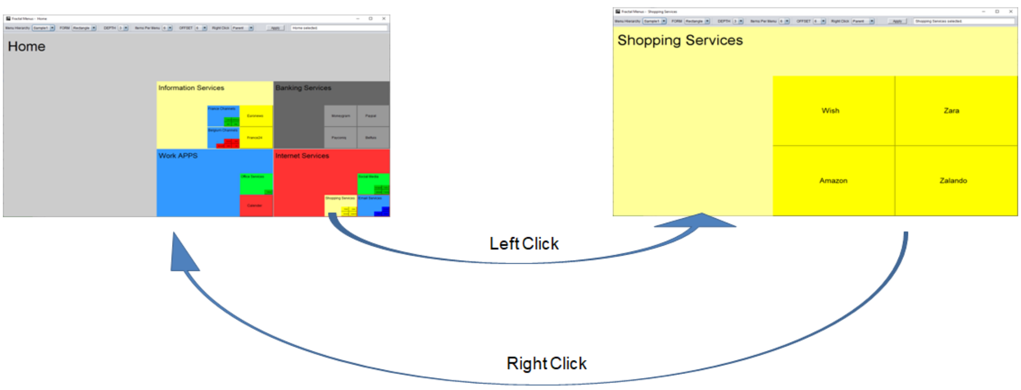


Figure 3.40: Illustration of right click configuration : Previous Menu Mode

3.3.5 Hexagonal Fractal

We developed a second fractal design which we called hexagonal fractal. It's class diagram is described in appendix B.1. In the Hexagonal fractal's implementation, sub-menus are limited to 3 items, and a maximum view depth of 2. The breath and offset were not implemented for the hexagonal fractal and as a result these options can't be modified from the options panel.

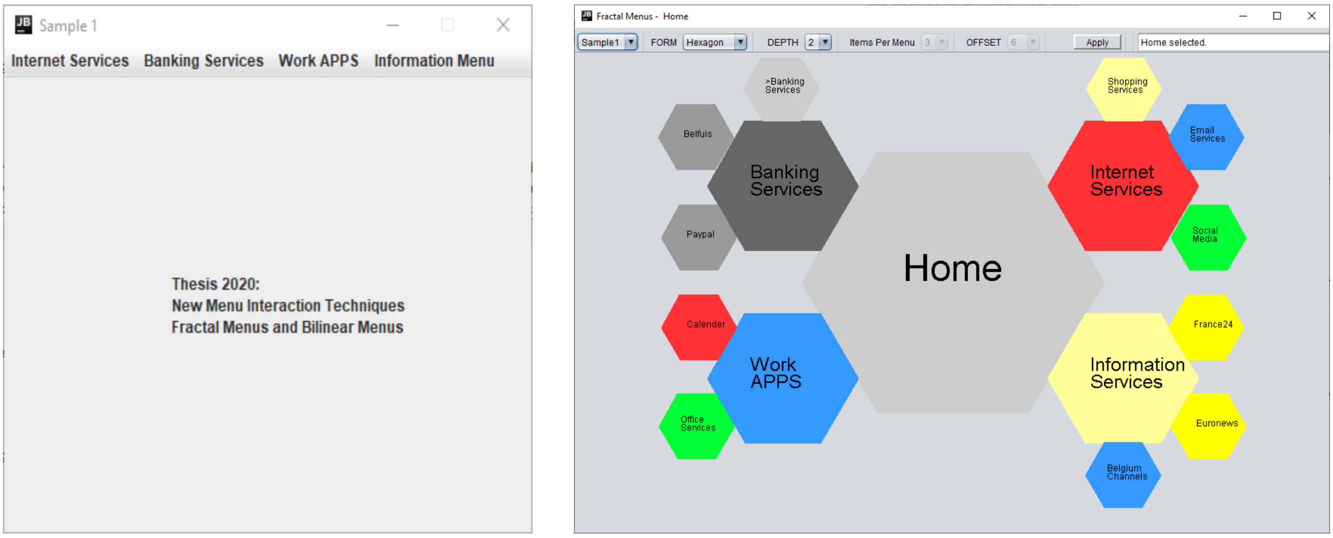


Figure 3.41: Rectangular Fractal Menu V1

Properties

A. **Item Selection** : Our fractal menu design implements the sequential and optimized item selection developed for the rectangular fractal. As with rectangular fractal, right click and left click event are use to respectively zoom into a selected item and out of the selected item. As illustration, say we wish to chose Word.

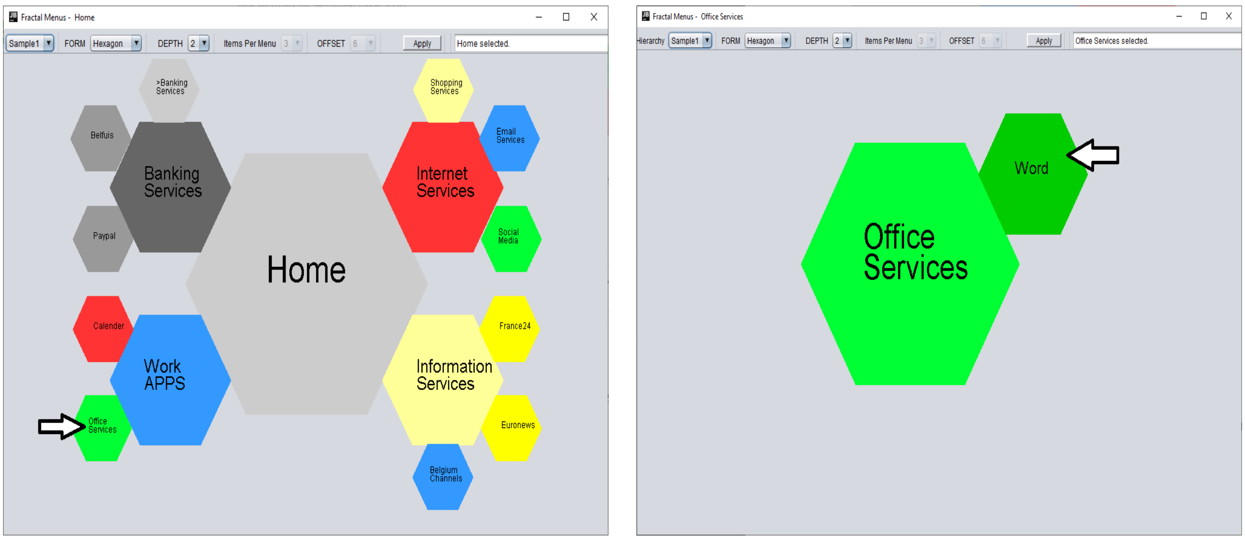


Figure 3.42: Hexagonal Fractal Menu : Item Selection - Example

B. **Depth Modification** : For the hexagonal fractal, to maintain its simplicity, the user can chose between a minimum depth of 1 and maximum depth of 2.

As explained in rectangular fractals, reducing the view depth increases the selection steps and therefore time for items that could have being selected from the view screen using the optimized selection. As example, say we wish to choose Yahoo.

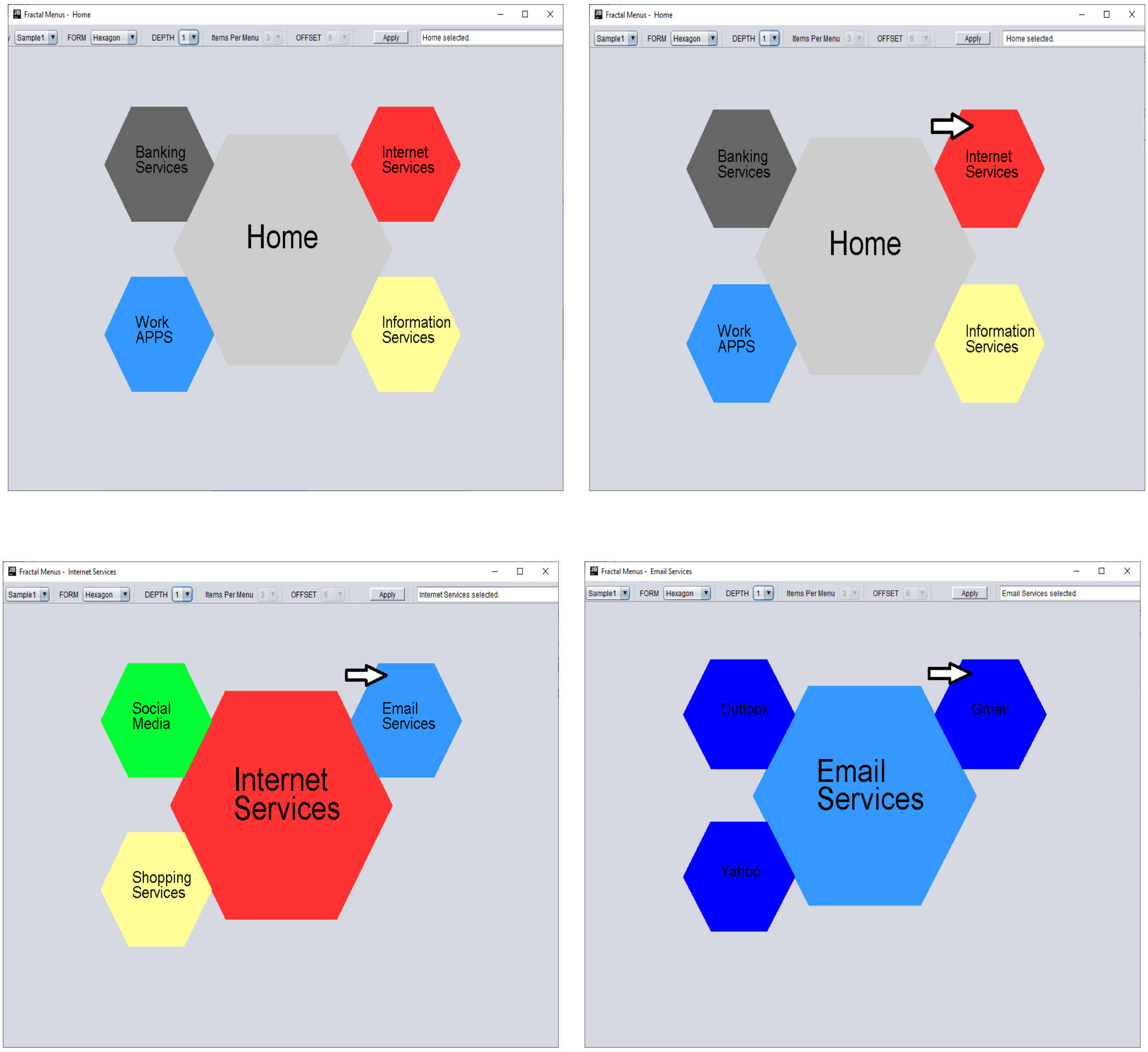


Figure 3.43: Hexagonal Fractal Menu : Item Selection - Example 2

C. For the hexagonal menu, the breath (items per menu) and offset were not implemented.

3.3.6 Characteristics of fractal menus developed

In this section, we review the different characteristics of the fractal menus developed.

A. **Visual Variables**

The first point we look at is its design space by enumerating visual variables[16] our menu has to effectively and efficiently convey a change.

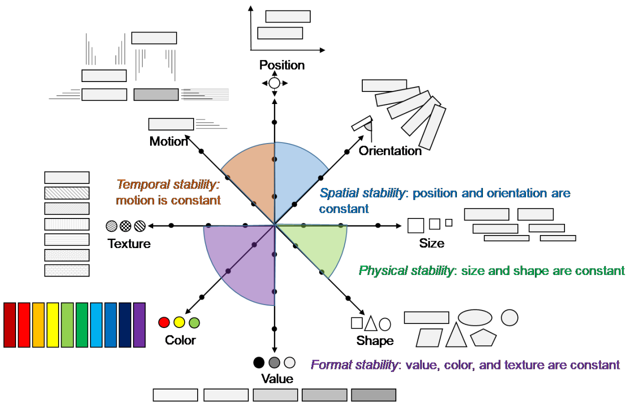


Figure 3.44: The Design Space for Graphical Adaptive Menus. [491]

- Spatial stability : Position and orientation of menu items are variable depending on their view depth.
- Physical stability : Our fractal menu achieves partial physical stability shape are preserved after configurations changes are applied but the size adapted to the view depth.
- Temporal stability : Preserves its position over time while being adapted.
- Format stability : Our fractal menu achieves format stability for each view depth.

B. **The frequency of adaptation**

In their study on exploring the design space for adaptive GUI, Gajos *et al.* [184] found that the frequency of adaptation of a menu appears to have a large impact on the weights people assign to the benefits of adaptability. In fact, the study found that a slow space adaptation resulted in strongly positive results. As a good compromise, our fractal menu applies the chosen options and adapts the menu once the user clicks on apply. The settings are equally saved for the next session.

C. **The selection time (little experiment)**

In 3.4, we present a little experiment we did on the rectangular fractal to evaluate the selection times for a list of menu items. The experiment aims at giving us an idea of selection times in a rectangular fractal menu.

[16] Based on Bertin's semiology.

3.4 Small experiment to evaluate selection times in the rectangular fractal menu

3.4.1 Introduction

We experimented the selection time for our rectangular fractal for different settings of depth and breadth. For our fractal menu, we believe the fractal design will be easier to get familiar with thereby further reducing the search time. We considered two profile for the experiment, our selves already familiar with the menu structure and friends who were unfamiliar with the structure. For each experiment, we repeated the procedure 2 to 3 times for the users unfamiliar with the menu so as to see how the selection times evolved as they got familiar with the menu.

As already mentioned, optimizing the selection time is out of the scope of this thesis. The experiment aims at giving us an idea of selection times for such a menu. A more elaborated and complete experimentation on usability and the selection times of the fractal menu compared to others menu should be the subject of a further study on fractal menus. Nevertheless, We propose in the conclusion different techniques we believe will further optimize these selection times for fractal menus.

For this little experiment, we use the java.time API. The timer is started when the main handler to reach the menu item is clicked and stop once the item its click.

We had three participants for our experiments. Before the experiment, we gave them a general explanation on the fractal menus and how its functions. We then asked them to select a given number of menu items. They selection times for these items we recorded and logged to a file. We repeated the process 2 to 3 times for each configuration still using the same items.

3.4.2 Experiment 1 : Using depth 3 and max 6 items per menu

For this experiment, we set the view depth to 3 and the maximum items per menu to 6. As, a reminder this configuration produces the following fractal:

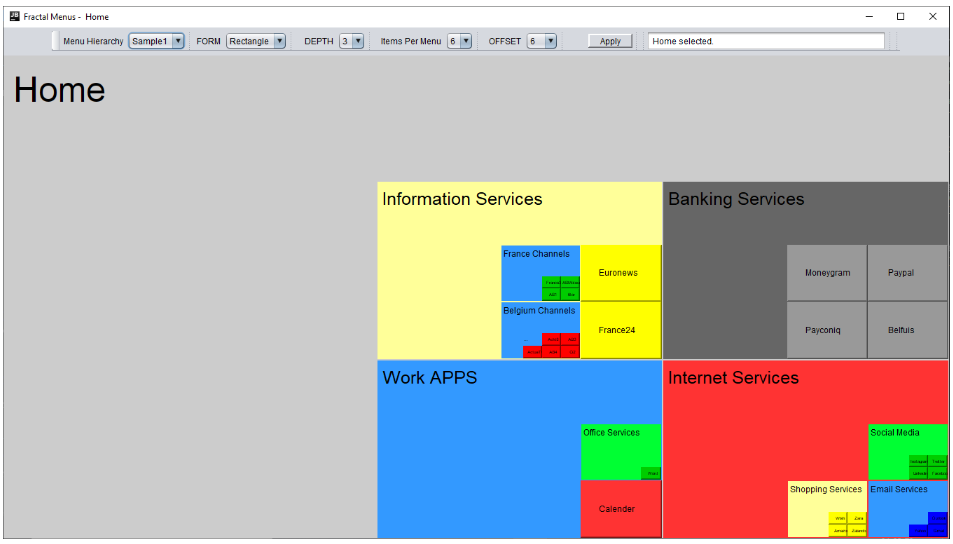


Figure 3.45: Rectangular Fractal : Depth 3 With Max 6 Items Per Menu

The following results were obtained for this experiment using a view depth of 3 and a maximum of 6 items:

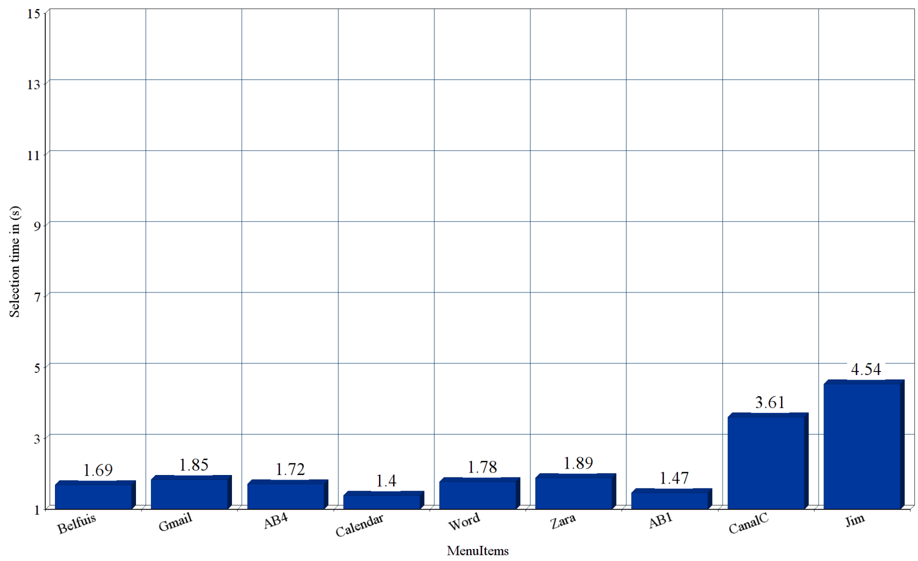


Figure 3.46: Experiment 1 : Selection times for expert users

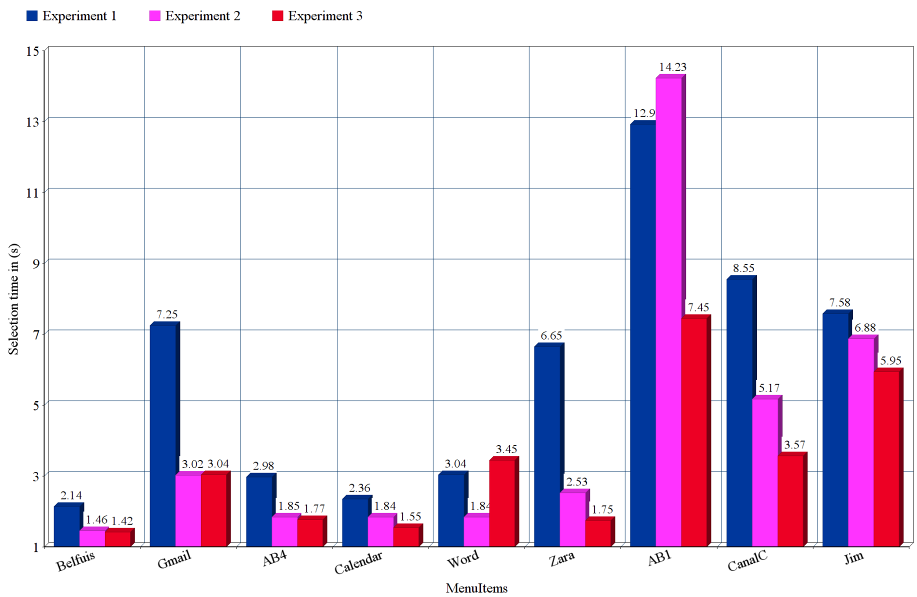


Figure 3.47: Experiment 1 : Average Selection times for unfamiliar users for three repetition of the experiment

As, we can see from the selection times: for the expert users it is relatively low as we are already familiar with the menu structure and know were each item is found. Selection times for user unfamiliar with the menu structure is higher but reduces as he/she gets familiar with the menu. The selection time for some items are smaller as the users easily identify those items i.e they are familiar with them e.g., "Belfius" while unfamiliar items like "AB1" and also deeper in the menu's hierarchy was more difficult to locate for the users.

3.4.3 Experiment 2 : Using depth 1 and max 6 items per menu

For this experiment, we set the view depth to 1 and the maximum items per menu to 6. As, a reminder this configuration produces the following fractal:

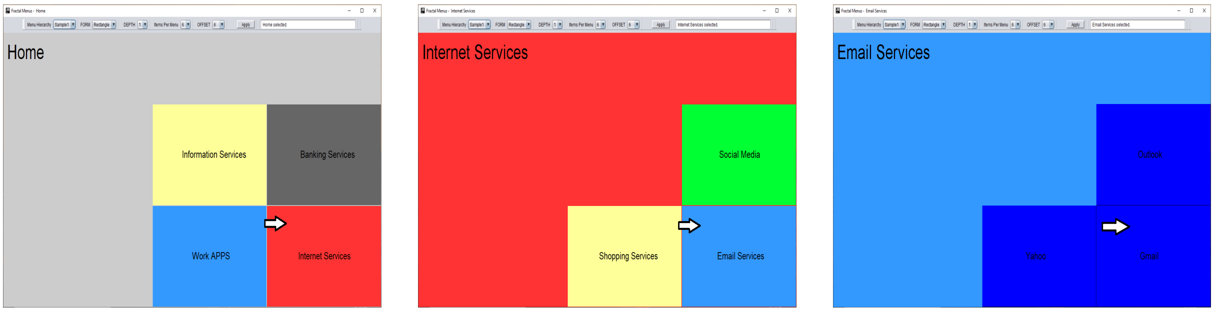


Figure 3.48: Rectangular Fractal Menu : View depth 1 and max 6 items per menus

The following results were obtained for this experiment using a view depth of 1 and a maximum of 6 items:

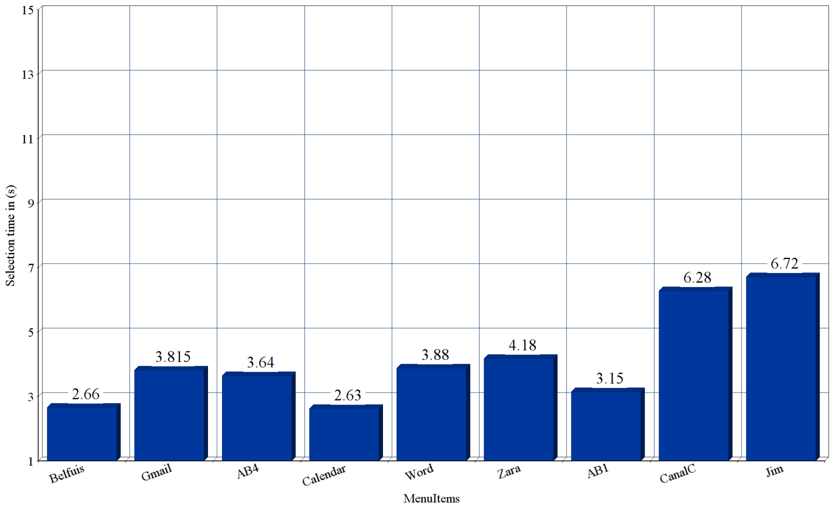


Figure 3.49: Experiment 2 : Selection times for expert user

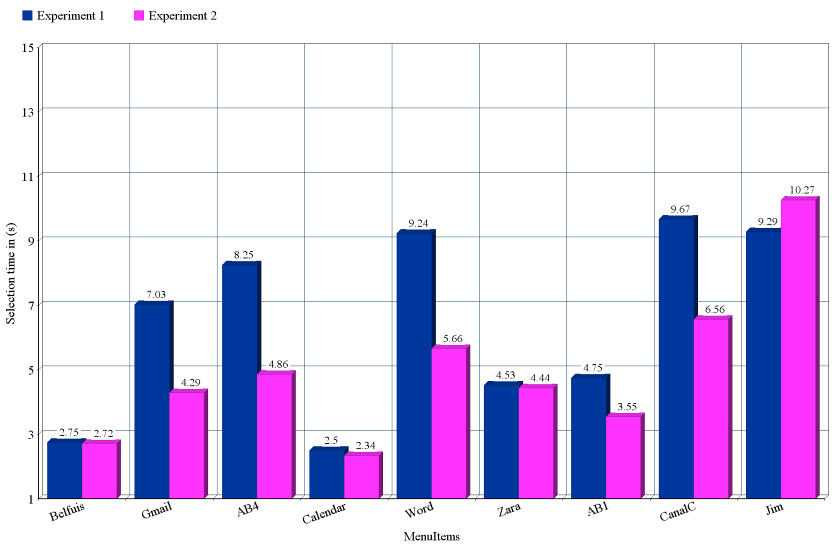


Figure 3.50: Experiment 2 : Average Selection times for unfamiliar users for two repetition of the experiment

From the result, we notice an increase in the selection times for most of the items. This is because, reducing the view depth removes from the viewed screen deeper items than could have being directly selected using the optimized item selection technique described previously. The selection times of items here increases because, their parent component will have to be zoomed in (clicked) in order for them to be visible and clickable.

3.4.4 Experiment 3 : Using Depth 3 and Max 3 Items per menu

For this experiment, we set the view depth to 3 and the maximum items per menu to 3. As, a reminder this configuration produces the following fractal:

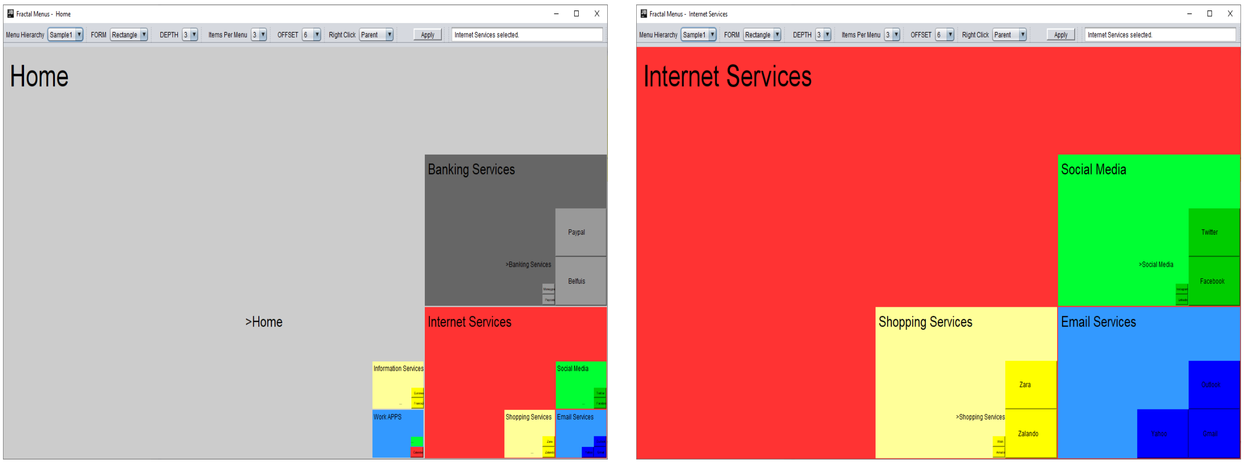


Figure 3.51: Rectangular Fractal : Depth 3 With Max 3 Items Per Menu

The following results were obtained for this experiment using a view depth 3 and max 3 items per menu:

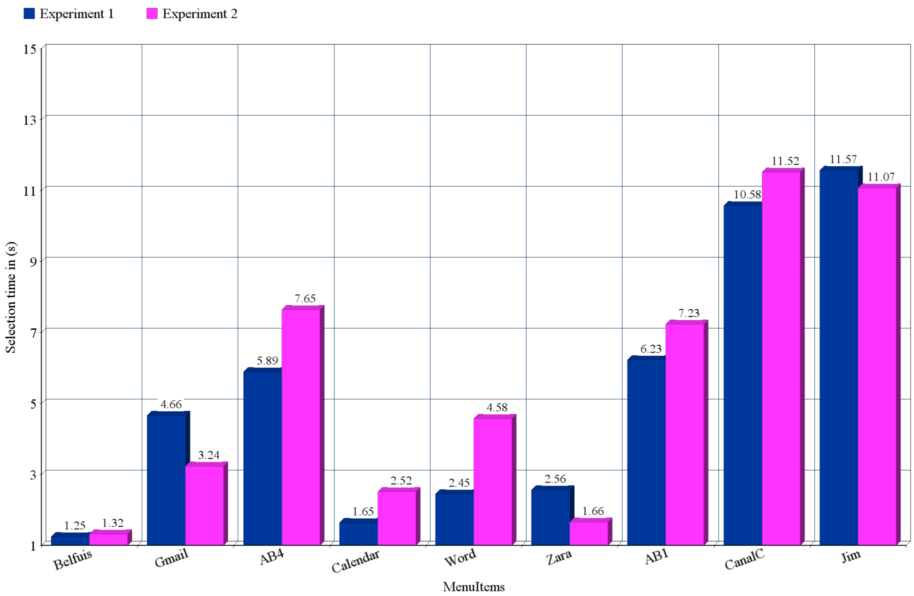


Figure 3.52: Rectangular Fractal selection times for unfamiliar user.

The increase in selection time here is explained due to the fact that reducing the max number of items per view screen puts additional items in a deeper view screen which increases their selection time.

3.4.5 Experiment 4 : Using Depth 1 and Max 3 Items per menu

For this experiment, we set the view depth to 1 and the maximum items per menu to 3. As, a reminder this configuration produces the following fractal:

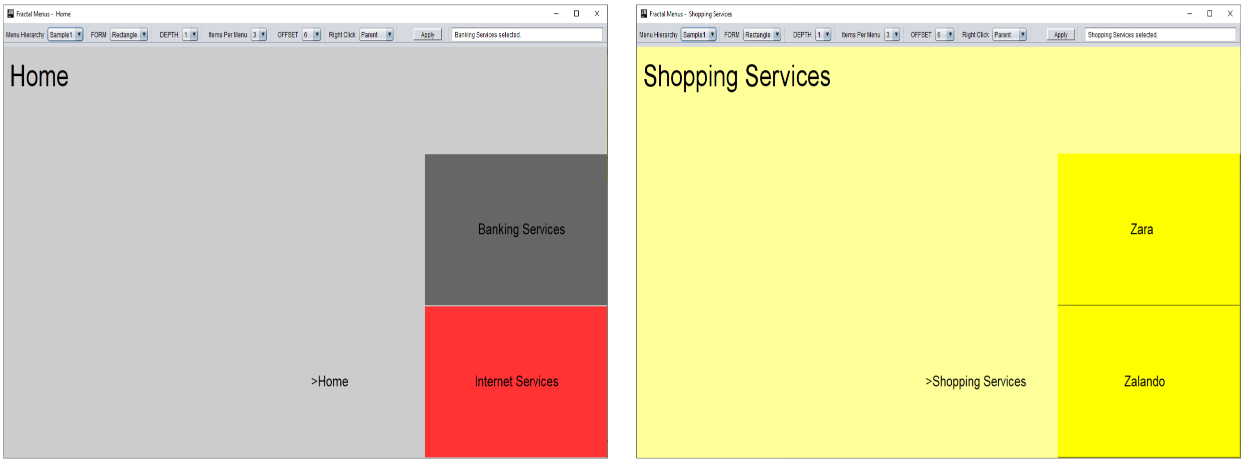


Figure 3.53: Rectangular Fractal : Depth 1 With Max 3 Items Per Menu

The following results were obtained for this experiment using a view depth 1 and max 3 items per menu:

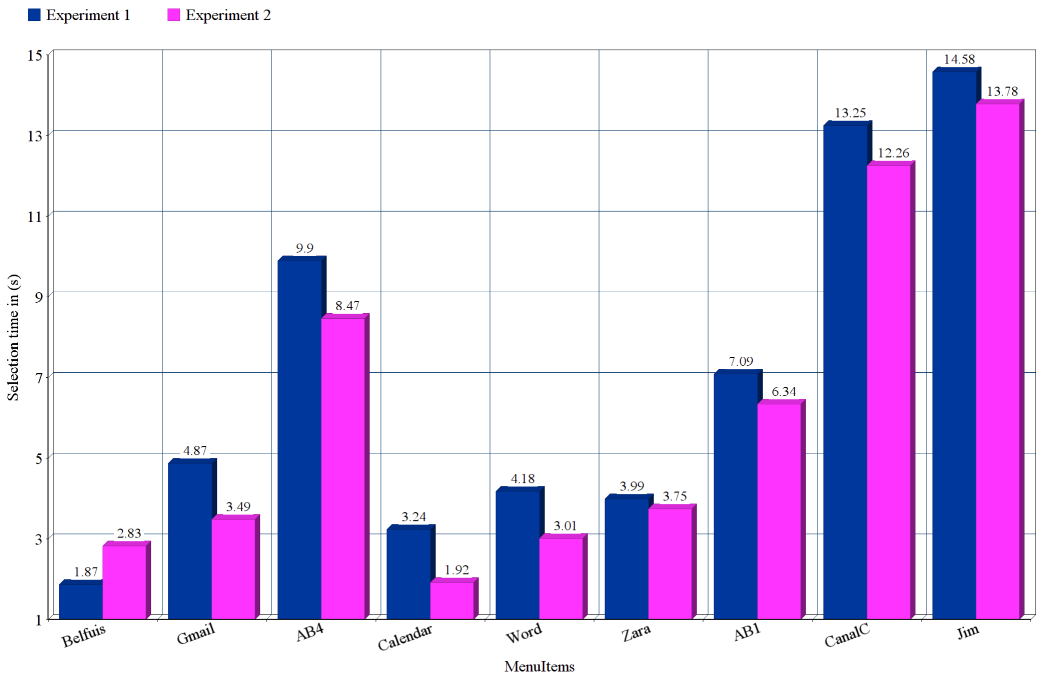


Figure 3.54: Rectangular Fractal selection times for unfamiliar user

Once more here, the reduced view depth and items per menu increases the selection times. These times are ameliorated as the user gets familiar with the menu.

3.5 Conclusion on the Fractal Menu

This is section, we developed two fractal menus namely a rectangular and hexagonal fractal. The developed fractal menus have the following benefits :

A. **Varying resolution**: Our fractal menus are designed to be suitable for varying screen resolution. Via the different options of depth and breath they can be configured to be suitable for a given screen size. For bigger screens, the view depth can be increased such than deeper items can be directly selected through the optimized item selection. For smaller screens, the view depth can be decreased and items per menu reduced to make the menu more usable on these screens.

B. **Some optimizations**: Our fractal's menu is design to do a little optimization of the selection time for menu item through their low fractal dimension which makes the menu less complex and menu item easier to identifier and also due to the configurable layout options such as increasing the items per view and increasing the view depth to directly access deeper items.

Our fractal menus still have some aspects which can be ameliorated or implemented :

A. We can further optimize the selection time for items in the menu by implementing some prediction strategies. For example, most frequently used items that are deeper in the menu can be brought some levels closer to further reduce their selection time especially in smaller screen constraints.

B. Some other interesting fractal shapes can be considered which may present betters properties in terms of space management and item selection.

C. Testing our menu on varying screen sizes and evaluating their resulting usability would very important to better evaluate and fix possible short comings of the implemented fractal menus.

D. Increasing the max items per menu or reducing it below 4 items per menu in our fractal menus results in resizing the icons to create space for additional view items. This results in all icons in all the view screen our of fractal menu having the same size whether the viewed screen has less icons or not. An amelioration will be to resize the icons per view such that, views will less icons than the max items per menu can have their icons size a little bigger. A better space optimization techniques should therefore be implemented such that space is optimize on a per view basis. We could also implement auto-configurations based on the resolution.

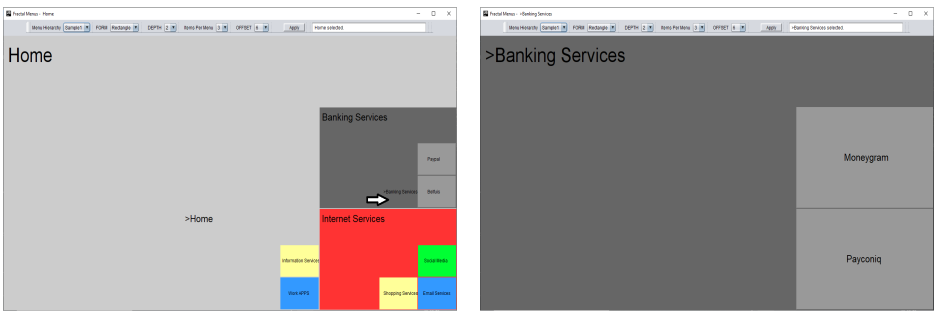


Figure 3.55: Rectangular Fractal : View Breath Modification Optimization Suggestion

Chapter 4

Development of a Bilinear Menu

Based on the SLR conducted in Chapter 2 , this chapter develops a second original multi-platform graphical adaptive menu, *i.e.,* the bilinear menu, based on geometric deformation. The background and the motivations for introducing this menu are first discussed. Then, we provide our definition of this menu and explain its design options which guide the implementation. Finally, an individual evaluation highlights benefits and shortcomings in a local conclusion. In the next chapter we will conduct a conjoint analysis of graphical adaptive menus.

4.1 Definition

We define the bilinear menu as a menu that uses distortion techniques to alter some aspect of the menu in order to provide a better view and access to a desired menu item.

4.2 Historical Background

The first attempts of using distortion in menu viewing and exploration was by Spence and Apperley's Bifocal Display [32] and Furnas' Generalized Fisheye Views [182]. Researchers and menu designers developed new algorithmic solutions of distortion techniques based on these origins, some of well-known examples are: Sarkar and Brown's Graphical Fisheyes [426] and [425], which expand upon Furnas' approach creating spatial reorganizations of visual representations; Hyperbolic Display [289], which uses mathematical function to create detail-in-context presentations; Perspective Wall [321], and Document Lens [414], which uses 3D perspective projection to create detail-in-context presentations; and Elastic Presentation [112], which also uses 3D projection.
High-precision magnification lenses also has been investigated in [33] and [405] but it is independent of the background and it does not take into account what type of contents it should emphasize with a very generic behavior and the zoomed area will take most of the screen space. To combine the ability to enhance the desired area without losing the data that are outside the focusing area due to the lack of screen space, the fisheye has been developed to achieves that. One of the most recent Fisheye Menus designs [70], applies a traditional fisheye graphical visualization techniques to linear menus, that provides an efficient mechanism to select items from long menus. In this design all of the elements are always displayed in a single window that is completely visible but it is known to present some limitations mainly:-

The deformation is only vertical and unidimensional. The fisheye deformation is concave and continuous zones before and after the currently being displayed item are decreasing equally, therefore preventing from having different deformations above the menu item and below the menu item. The deformation is applied on textual menu items only by increasing/decreasing the size font depending on where the cursor is. Finally it cannot handle menu items with icons, shortcuts, or gestures since the fisheye deformation is only applicable to the textual items. Several researchers have investigated methods to eliminate these limitations but most solutions proposed trad-offs in their design to solve one or more limitations, therefore the need has been raised to design an innovated menu that address all these limitations in one single design.

4.3 Motivations for a Bilinear Menu

From our SLR we realised that menu designers are focusing on solving problems related to three factors:

A. Human Factor : This factor involves erroneous selections due to the size of the selecting area being small, or because of difficulties moving the cursor in sharp corners [477] which leads to sub-menus appearing and disappearing. Another human factor is the age [398] which can lead to seeing difficulties and unpleasing user experience by spending too much time in visual search through the menu items.

B. Device's Design Factor : This factor involves the platform, screen size, the input method, and the number of the device functionalities, all of this may lead to rich interface and long menu lists. Some researchers tried to overcome some of these challenges like :

- Icon-Scenario Based Animated Menus [276] to overcome the limitation of screen size in some small devices. This method provides a preview for an icon that took most of the screen size yet the visibility enhanced for one icon by taking most of the screen size for the preview and not leaving any space left for more icons.
- Another solutions focused on the ability of the menus to becomes adaptive to the user usage like Bubbling Menus [477]. This menu eliminate the possibility of selecting an item from a drop-down menu by mistake and highlights the most used items thereby reducing the visual search time for the desired item. But again there are some limitation in the design : the number of items in single drop-down menu remains limited and for deep hierarchical menu list the user will have to move between several sub-menus to reach the desired item.

C. Shape and Layout Factor : This factor is related to the menu shape and layout overview. Researchers and designers tried to manipulate the shape of the menus and layouts to enhanced the menu usability on the expense of the classical look of the menus. That's was a disadvantage for the new users since they need some time to practice and understand the new layouts of the menu. We designed our menu to maintain the classical shape and layout, therefore we use visualization techniques only.

4.4 Bilinear Menus's Requirements

The idea of bilinear menus is to enable the user to interactively change the perspective on the data [471] very much in the sense of what Bertin [77] said:

'A graphic is not "drawn" once and for all; it is "constructed" and reconstructed until it reveals all the relationships constituted by the interplay of the data. The best graphic operations are those carried out by the decision-maker himself.'

Jacques Bertin, was the first to develop the concept of visual variables in his book, "Semiologie Graphique," [249]. Bertin identified seven main categories of visual variables: position, size, shape, value, color, orientation, and texture, these seven Bertin's original variables illustrated in 4.1 have been modified and expanded in number by various cartographers and authors.

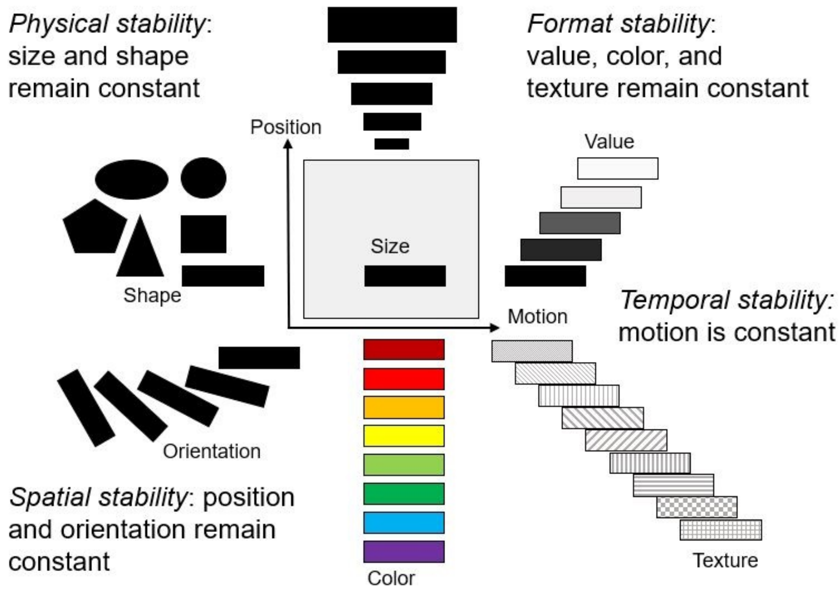


Figure 4.1: Visual Variables

Position

The position of map elements is important in cartography. Absolute location on a map cannot be altered, but the position of labels and information can affect the viewer's perception of a map.

Size

The size of a label or symbol is how much space it occupies on a map. Size differences are relatively easy to recognize, making it a useful variable. The size of symbols can convey information, such as a quantitative amount of something, or can be used to attract a viewer's attention *i.e.,* raising it in the Visual hierarchy. Because geographical features have an actual size on the earth, this cannot always be controlled, and sometimes works against the wishes of a cartographer. For example, it can be difficult to make a world map in which Russia does not stand out.

Shape

A shape is a simple design that is used to symbolize an attribute on a map. They reference a location with a certain attribute. Different shapes generally correspond with different attributes. For example if a map of a city has a red cross on it then, most people will assume that there is some sort of medical services provided at that location. Another example would be a star on the map that denotes that a particular location is of some importance to the reader. Some shapes are simple in nature and thus are more abstract, while other shapes are more pictorial and are easy for the reader to comprehend what is trying to be conveyed. Shapes come in many different varieties and should be used in simple symbology as not to confuse the viewer. Generally, shapes are also easily recognized in contrast with one another. Some aspects of shape are inherent to the phenomenon being portrayed and may not be easily manipulable, especially in line and region symbols, such as the shape of a road or a country. In line and region symbols, shape can also play a role as part of a pattern, such as a region filled with tree icons.

Color

Color is the visual perceptual property corresponding in humans to the categories called red, green, blue, and others. Humans generally perceive three aspects of color: hue (the commonly-named colors of the rainbow), saturation (the intensity or brightness of a color), and value (the lightness or darkness of a color). Choropleth maps often use color value to differentiate between characteristics that are being mapped.

Value

As an aspect of color, value refers to how light or dark an object appears on a map. Value effectively conveys a feeling of "more" and "less," an ordinal measure; this makes it a very useful form of symbology in thematic maps, especially choropleth maps. Value contributes strongly to Visual hierarchy; elements that contrast most with the value of the background tend to stand out most (e.g., black on a white sheet of paper, white on a black computer screen).

Orientation

Orientation refers to the direction labels and symbols are facing on a map. Although it is not used as often as many of the other visual variables, it can be useful for communicating information about the real-world orientation of features. Common examples include wind direction and the direction in which a spring flows.

Texture

Texture refers to the aggregate pattern made up of many individual symbols. For example, a dense network of lines representing streets could collectively convey the concept urban area. An evenly spaced lattice of green dots could mean orchard, while a random distribution of the same green dots could mean forest.

Hence to preserve the variables in a typical standard menu we made design choices to implement the bilinear menu using viewing techniques that are subject to a bilinear transform of the zone containing the menu item. Which it is basically a dynamic process for distorting one region depending on the position of the cursor. When the cursor is over a pull-down menu, the region surrounding the menu item is subject to maximal distortion (based on a degree of interest), but the surrounding regions are also subject to different distortions so as to preserve continuity of viewing between the distorted zone (the menu item) and the rest of the interface (the surrounding zones).

4.5 Inspiration From Photography

The definition of the bilinear menu explain the use of the distortion and viewing techniques to provide a better view and there is no better domain to take an inspirations from than photography. In photography radial distortions are very common because every lens is suffering from distortion (both cheap and high-end lenses). This is due to the physics of light no lens can give a perfect photo that is distortion free. Some photographers use distortion as a way to produce certain photos with natural effect like the fisheye lenses that can produce a barrel distortion for example see figure 4.2.

Figure 4.2: Photo taken by a fisheye lens.[7]

Note that this type of distortion is considered an optical distortion since a special type of lens is needed to produce it. We also have not optical radial distortion obtained by using normal lens with some photographic positioning techniques to produce it, an example for this type is the prospective distortion as illustrated in figure 4.3.

From the figure we noticed that the left part of the car looks disproportionately bigger. The left light seems twice as big than the one on the right, although they are the same size. The car is occupying the majority of the frame and everything in the background looks relatively small.

Photographers can also produce random distortions effects by harnessing the environments to his advantage an example of random non-uniform radial distortion is shown in figure 4.4.

[7] Photo by Anuj Gogari https://loadedlandscapes.com/reviews-best-fisheye-lenses-canon/
[8] Photo courtesy of Nasim Mansurov from https://photographylife.com/what-is-distortion

Figure 4.3: Perspective Distortion photo [8]

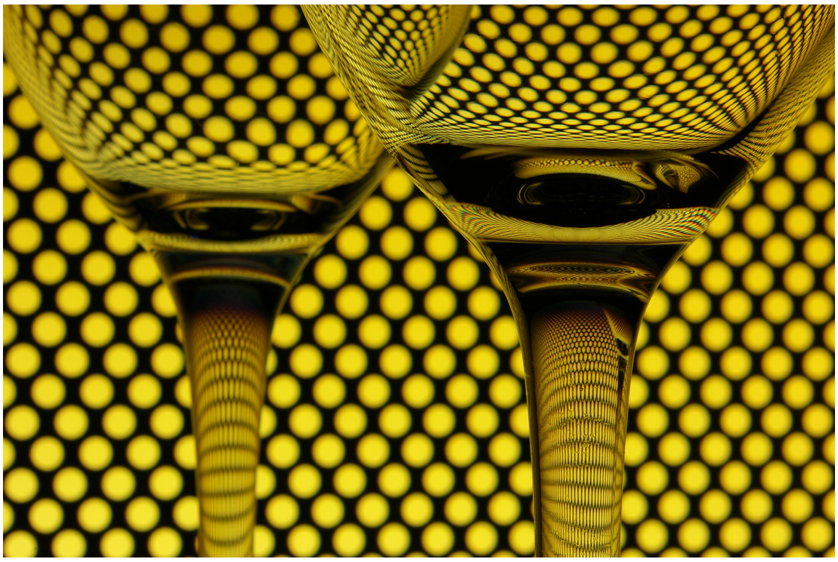


Figure 4.4: Wine glasses create Radial non-uniform distortion of their background.[9]

We will discuss these distortions techniques in details and implement it to create our innovative menu.

4.5.1 Distortion Techniques

In this section we will be discussing the radial distortion techniques used in our bilinear menu and the implementation of DOI (degree of interest) function in these techniques.

4.5.2 Radial Distortion Techniques

Distortion can be irregular or follow many patterns, the most commonly encountered distortions are radially symmetric, arising from the symmetry of a photographic lens.

[9] Photo courtesy of Atoma https://en.wikipedia.org/wiki/Distortion_(optics)#/media/File:Uniformity.jpg

Hence this is an optical distortion which it is a result of the optical design of lenses. These radial distortions can usually be classified as Barrel Distortion or pincushion distortion or mustache distortion as illustrated in figure 4.5, There is a visual similarity between barrel and mustache distortion thus, it is hard to distinguish the difference between these two distortions types unless the distortion degree was sever, hence we chose to focus on the barrel distortion since it is easier to implement and provide better visual. The pincushion distortion does not serve our desired requirements since its tend to make the targeted menu item looks smaller and farther in fact it is the opposite of barrel distortion see figure 4.5.

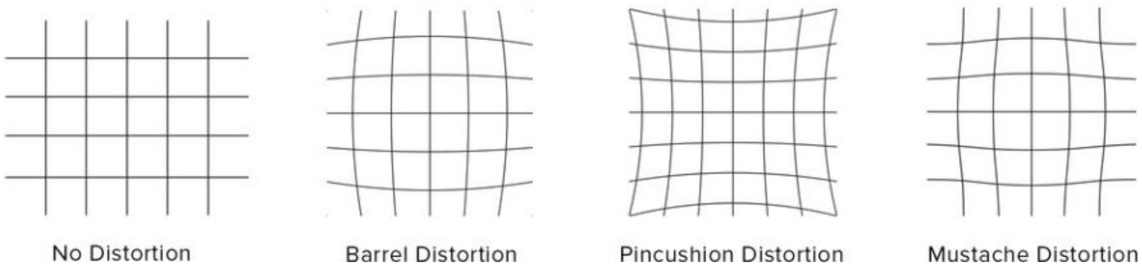


Figure 4.5: Illustration of radial distortion models

4.5.3 Barrel Distortion

In barrel distortion, image magnification decreases with distance from the optical axis. The apparent effect is that of an image which has been mapped around a sphere (or barrel). Fisheye lenses for example utilize this type of distortion as a way to map an infinitely wide object plane into a finite image area. Barrel distortion appears in the middle of the lens and is worst at the wide-angle end of the lens, an example is shown in figure 4.6 of a photo taken by a camera using fisheye lens to produce barrel distortion effect.

Figure 4.6: Barrel distortion photo taken by a fisheye lens for Lena[10]

As illustrated, we notice that the center of the image is distorted in a degree (depends on the type of lens that has been used) resulting in a relatively zoomed/enhanced image compared to the original one.

[10] Photo taken from www.the-pro-photographer.com/how-to-find-the-perfect-portrait-lens-to-avoid-distortion

Barrel Distortion Formula

Inspired by [205] we can have a barrel distortion in computer vision by applying this formula :-

$$r_u = r_d(1 + DOI * r_d^2)$$
$$as \;\; r_u \;\; is \;\; calculated \;\; as : r_u = \sqrt{x_u^2 + y_u^2}$$

where r_u *and* r_d are the distance from the centre of distortion in the undistorted and distorted images respectively as shown in Figure 4.7 and DOI is the distortion parameter, which it is our degree of interest.

DOI of Barrel Distortion

The degree of interest (DOI) must be in range of 4/27 to 3. Where 3 will gives a sever distortion results and anything below 4/27 will give a pincushion distortion which is not desirable in our case as we already explained.

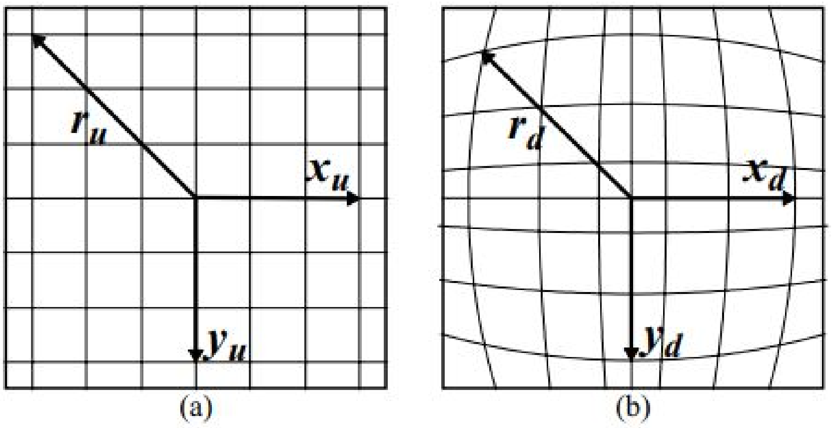


Figure 4.7: Illustration of barrel distortion parameter, (a) the original data,(b) the distorted data

4.5.4 Perspective Distortion

Perspective distortion is another type of distortion that is often seen in images as illustrated in figure 4.8. Unlike optical distortions, it has nothing to do with lens optics and thus, it is not optical. When projecting three dimensional space into a two dimensional image, if the subject is too close to the camera, it can appear relatively large or distorted when compared to the objects in the background.
This is a very normal occurrence and something you can easily see with your own eyes. If were sitting in front of a laptop and you look at your hand, then bring your hand very close to your eyes, it will appear large relatively to the laptop in the background (and the farther your hand is from your laptop, the smaller the laptop will appear relative to your hand). The same effect happens when photographing any subject. For example see figure 4.3.

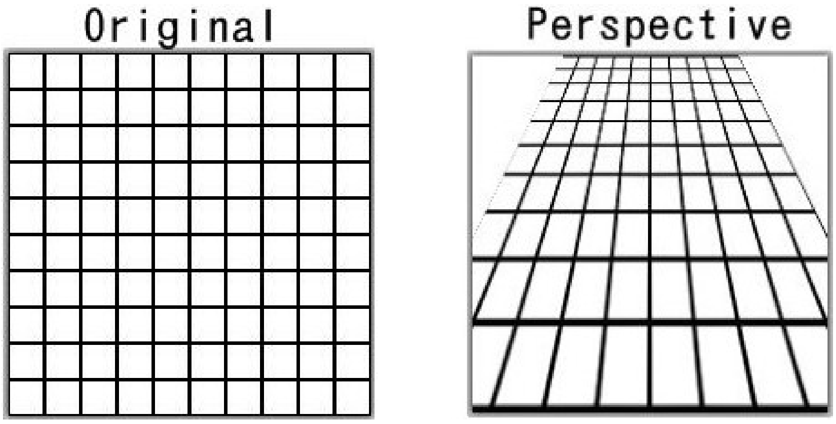


Figure 4.8: Illustration of Perspective distortion model

Perspective Distortion Formula

Inspired by [27] we can have perspective distortion by using Homography Transform by applying this formula :-

$$\begin{bmatrix} x' \\ y' \\ w \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \rightarrow \begin{bmatrix} x' \\ y' \\ w \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ h_{31} & h_{32} & 1 \end{bmatrix} = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

We need to change the image coordinates from two dimensions to three dimensions by adding w where w = 1 and h_{31}, h_{32}, h_{33} *is equal to* $0, 0, 1$ *respectively.*
If we changed h_{31} value the image will have a perspective on the y axis and if we change the value of h_{32} the image will have a perspective on the x axis, note that if we put these parameters equal to zero the resulted image will be distortion free.
After converting to the homogeneous form we proceed by applying these formulas:

$$x' = h_{11}x + h_{12}y + h_{13}$$

$$y' = h_{21}x + h_{22}y + h_{23}$$

$$w' = h_{31}x + h_{32}y + h_{33}$$

finally we convert our new coordinates to be two dimensional again and calculate the new values:

$$x'' = \frac{x'}{w} = \frac{h_{11}x + h_{12}y + h_{13}}{h_{31}x + h_{32}y + h_{33}}$$

$$y'' = \frac{y'}{w} = \frac{h_{21}x + h_{22}y + h_{23}}{h_{31}x + h_{32}y + h_{33}}$$

where x” and y” is our new perspective x,y-coordinates.

DOI of The Perspective Distortion

The DOI here is the scaling factor of our distortion. We can control it by changing h_{31} or h_{32} which can take a positive or negative value depending on the desired direction of the distortion. The values of these parameters should be between -0.0099 and 0.0099.

4.6 Implementation of a Bilinear Menu

In this section we are going to describes the area of interest (AOI) then we describe the methods and the techniques that has been used to implement the bilinear menu.

4.6.1 Area of Interest

The area of interest abbreviated as (AOI) is the most important part of our bilinear menu implementation. We are going to describe it in details to provide a good overview.

Definition

The AOI is a lightweight tool acting like an Interactive Lenses for Visualization [471] to solve a localized visualization problem by temporarily altering a selected part of the visual representation of the data.

Interactive Lenses for Visualization

As Tominski described in [471], lens techniques are lightweight on-demand tools, illustrated in figure 4.9, a lens is a parameterizable selection according to which a base visualization is altered. Typically, a lens is added to a visualization interactively to solve a specific localized problem. A key characteristic is that the lens effect (*i.e.,*. the modification of the visualization) is transient, that is, the visualization returns to its original state once the lens is dismissed.

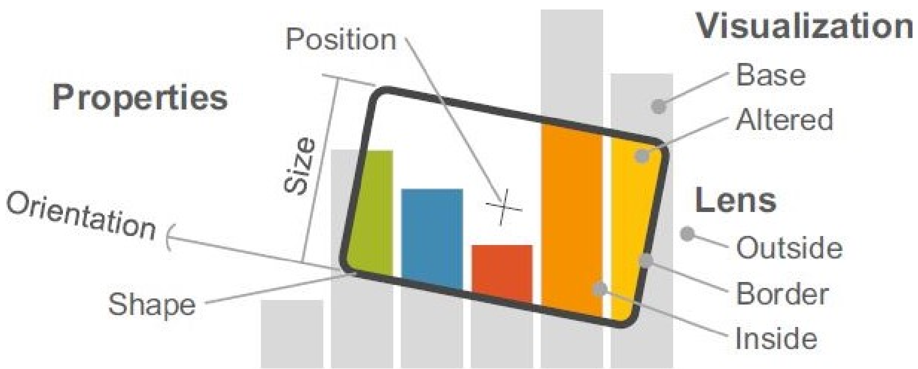


Figure 4.9: Schematic depiction of an interactive lens [11]

In order to actually implement such an approximation we decided to choose a rectangular shape lens to define our (AOI) since most of the shapes in the user interfaces and menus are rectangular, to achieves that we need to define the height (h) and width (w) of our rectangle as our lens bounds, these parameters can be changed according to the user customization. With our rectangular parameters defined, we can now raster[1] all the data within the rectangle and then process it with our algorithm.

As illustrated in figure 4.10 the green dots are the area of interest coordinates and the red dot is our target point (*i.e.,* pixel we want to process). In computer visions when we want to enlarge or do any kind of transformation for an image that required scaling we

[11] Figure taken from[471]

[1] Raster graphics, also called bitmap graphics, a type of digital image that uses tiny rectangular pixels, or picture elements, arranged in a grid formation to represent an image.

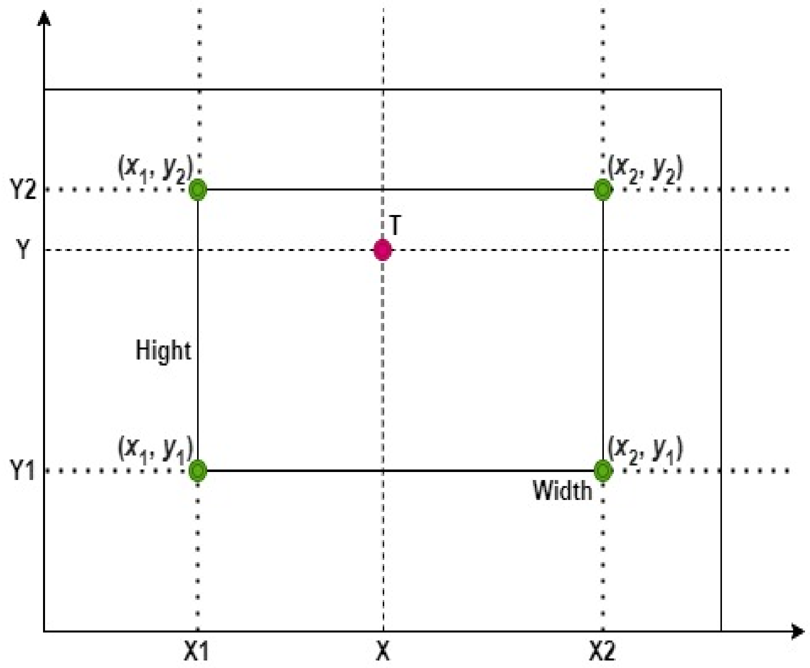


Figure 4.10: illustration of area of interest

have to perform an interpolation to achieve that.

In more details our AOI is a 3x3 rectangular grid on the target plane, the central cell of the grid is a scaled-up image of a rectangle on the source plane which is the focus area, the ratio of this scaling is the DOI. All 8 other areas are distorted with a bilinear interpolation so that the four outer sides of the 3x3 grid are not distorted, and the neighboring cells are seamlessly aligned. The positions of the grid and the focus area depend on which type of distortion is used.

4.6.2 Interpolation

In this section, we will define interpolation, interpolation forms and how we used it to make our bilinear menu.

4.6.3 Definition

In the mathematical field of numerical analysis, interpolation is a type of estimation, a method of constructing new data points within the range of a discrete set of known data points [445].

In engineering and science, one often has a number of data points, obtained by sampling or experimentation, which represent the values of a function for a limited number of values of the independent variable. It is often required to interpolate(*i.e.,* estimate the value of that function for an intermediate value of the independent variable), hence it is used in computer vision for the computation of pixel color values between the pixels that already exist.

4.6.4 Image Interpolation Methods

There are three methods of interpolation :- the nearest neighbour method [214], Bilinear Interpolation Method [204] and bicubic method [186]. The neighbour method is the simplest one to implement but the result images where with obvious pixelation-edges that break up curves into steps or jagged edges see 4.11(b), on the other hand the bilinear method is delivers more visually satisfying results comparing to the nearest neighbour method see 4.11(c). The bilinear interpolation reduces pixelation by filtering the surrounding pixels and smooth out the jagged edges giving the image a smoother look. The bilinear method is based on the color values from only the four surrounding pixels. These pixels are sampled to provide the color value for the new pixel added during interpolation. The contrast between the jagged edges produced by the nearest neighbor method is reduced because of averaging neighboring values together [214].
The bicubic method 4.11(d) takes a step further in interpolation by taking the sixteen pixels around each individual pixel and analysing these pixels to use it for interpolation. We found that bicubic is resources consuming due to a larger calculations so we chose the bilinear method as our interpolation method since it delivers an acceptable image result and a relatively smoother performance.

Figure 4.11: (a) original image of lena, (b) nearest neighbour method, (c) bilinear method, (d) bicubic method

4.6.5 Bilinear Interpolation Method

In mathematics, bilinear interpolation is an extension of linear interpolation for interpolating functions of two variables (e.g., x and y) on a rectilinear 2D grid. Bilinear interpolation is performed using linear interpolation first in one direction, and then again in the other direction. Although each step is linear in the sampled values and in the position, the interpolation as a whole is not linear but rather quadratic in the sample location. Bilinear interpolation is one of the basic re-sampling techniques in computer vision and image processing, where it is also called bilinear filtering or bilinear texture mapping.

Bilinear Interpolation Algorithm

To illustrate the bilinear interpolation algorithm, suppose that we want to find the value of the unknown function f(Q) at the point (x, y).
It is assumed that we know the value of f(Q) at the four points Q11 = (x1, y1), Q12 = (x1, y2), Q21 = (x2, y1), and Q22 = (x2, y2), you can consult figure 4.10 and imagine these green dots are our f(Q) points.

We used another formula for the bilinear interpolation (longer version), you can find it in the appendix C but you can have the same result by using this alternative formula.
The solution to the interpolation problem is :

$$f(x, y) \approx b11f(Q11) + b12(Q12) + b21f(Q21) + b22f(Q22)$$

where the coefficients are found by solving :

$$\begin{bmatrix} b_{11} \\ b_{12} \\ b_{21} \\ b_{22} \end{bmatrix} = \left(\begin{bmatrix} 1 & x_1 & y_1 & x_1y_1 \\ 1 & x_1 & y_2 & x_1y_2 \\ 1 & x_2 & y_1 & x_2y_1 \\ 1 & x_2 & y_2 & x_2y_2 \end{bmatrix}^{-1} \right)^{T} \begin{bmatrix} 1 \\ x \\ y \\ xy \end{bmatrix}$$

The implementation of the formula delivered a slow performance due to the high number of calculations, since for each pixel of the target image, we calculate the x,y coordinates of the intended pixel on the original image, but these can be floats, they represent a point somewhere between 2x2 neighboring pixels. The trick that made it work much faster is that we do rounding conversion to integers then we do the bilinear interpolation.

4.6.6 System Design and Life Cycle

When the bilinear menu generator get executed, the first thing will do is read the data underling the mouse pointer within the area of interest and put it in a buffer continuously. These buffered data taken from the area of interest which subjected to the predefined parameters by the user. The system monitor will observe the mouse pointer, it will create a new empty buffer and trigger the data interpolation if the pointer where pointing at a menu item. Then the interpolated data will be sent to the distortion function to generate the desired effect corresponded to the predefined DOI.
All these processed data will be held in the second buffer then will be sent to the draw function to draw it in the new window, in another case if the mouse pointer is pointing to anything but a menu it will draw the same image in the new window without interpolation and distortion.
The reason for creating new window is for preserving the data of the original pixel from being modified, a system flow diagram is illustrated in figure 4.12.
The system monitor not only manages the triggering of the interpolation and distortion but also it is handling the mouse movements in the new window to mimic the original mouse in the original window perfectly.

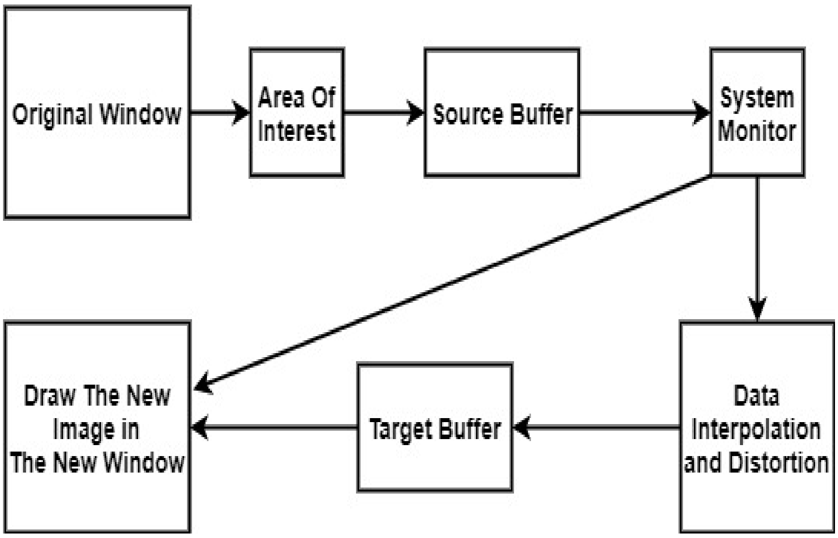


Figure 4.12: System diagram

4.7 Bilinear Menu's design option

In this section, we describe the distortion effects and techniques that has been implemented in the bilinear menu and the design.

4.7.1 Introduction

Bilinear menu consists of two types of effects that has been derived from the barrel and perspective distortions effects.

4.7.2 The First Technique

The first technique is a combination of Barrel Distortion technique and a Perspective Distortion technique see figure 4.13. The highlighted area with blue dots is the area where the two distortions are combined and it is our area of focus that contains the targeted item or items.

The first technique is illustrated in figure 4.14. As illustrated in the figure the image got distorted and the targeted menu item CNN in our case now looks closer and bigger and the next item got distorted relatively too. When the mouse pointer leaves the menu and hovers over anything else, the menu is restored to it's original appearance as shown in (a).

4.7.3 The Second Technique

The second technique combines two Perspective Distortion, one starting from the top and another from the bottom of the frame. Both meeting in the center which will be our area of focus, at that point (the center). Before the two perspective distortion meet each other they both halt and preview a normal view like a normal zoomed image.

The size of the meeting area is variable according to the users desires so he can choose the number of items that can be enhanced every time he moves the mouse pointer on the menu.

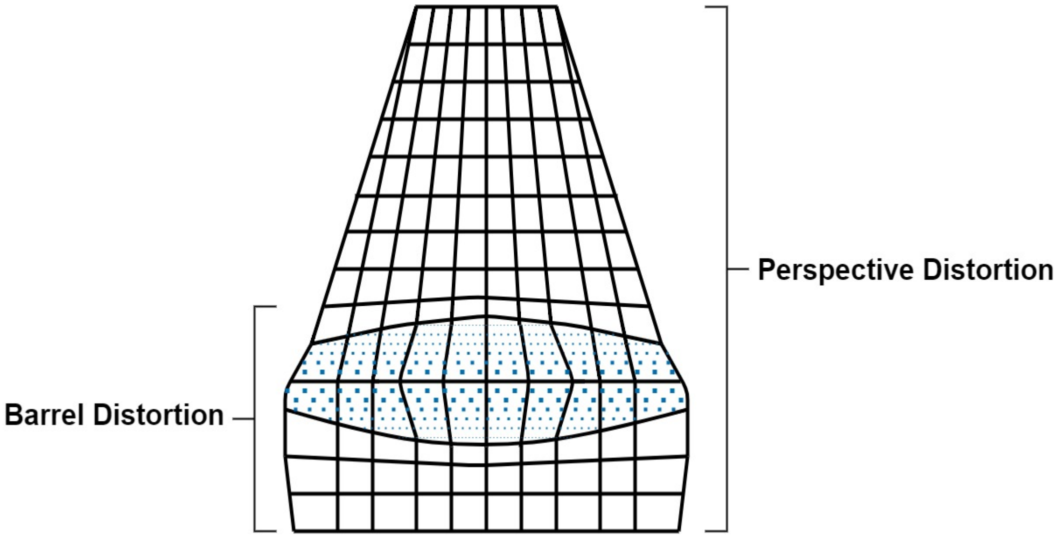


Figure 4.13: Illustration for the usage of barrel and perspective distortions

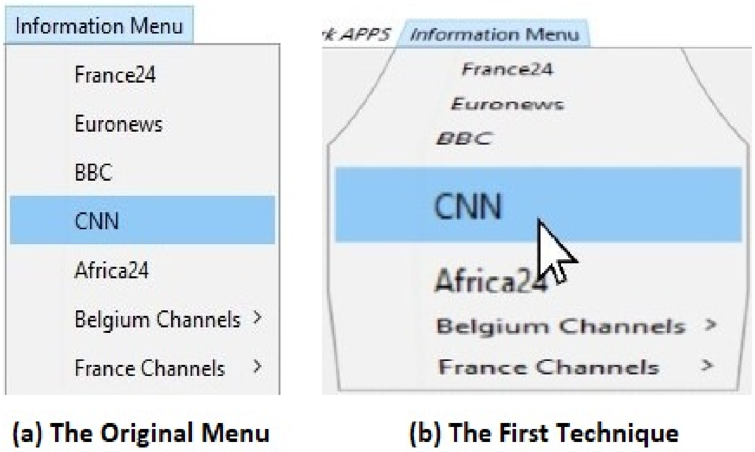


Figure 4.14: Bilinear Menu First Technique in Action

The degree of the perspective distortion will be modified according to our DOI, the meeting points of the two perspectives will focus on the targeted item or items to enhance the visibility within the menus and facilitate the item selection. See figure 4.15, note that the blue dots in the figure represent the meeting area.

For better visualization of the second technique see figure 4.16 the bilinear menu in action.

As illustrated in the figure the image got distorted and the targeted menu item CNN in our case now looks closer and bigger.
Note that, if you compare the result with first technique you will not notice a big difference at glance but if you look closer you will notice that the menu title remains the same when compared with the original menu and that is not the case in the first technique. Another difference is that the bottom part got distorted equally as the upper part of the image and that's also not the same as the first technique. Finally as the first technique, if the mouse pointer leave the menu and hover over anything else the menu will be restored to

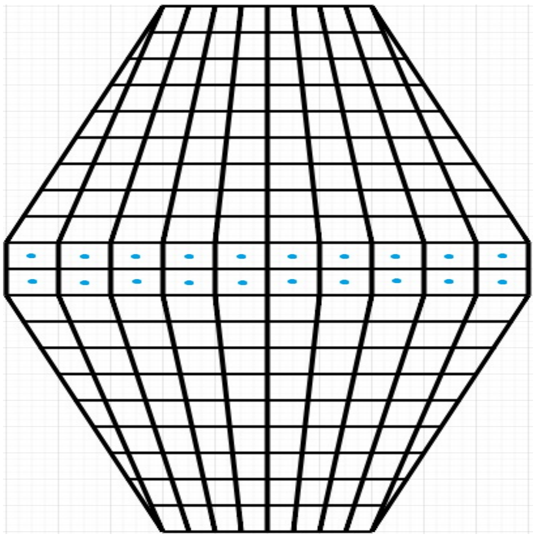


Figure 4.15: Illustration of The Double Perspective Distortion Model

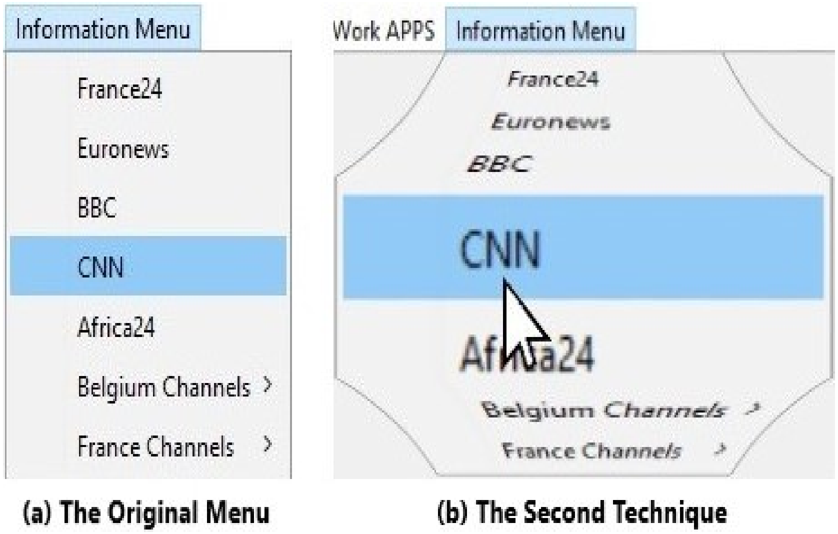


Figure 4.16: Bilinear Menu First Technique in Action

its original view as shown in (a).

4.7.4 Algorithm of Bilinear Menu

The algorithm of the bilinear menu generator is quite complex since it uses two buffers, two screens, and does many mathematical calculations, we tried our best to represent it in a way to have a general overview for the program by describing the main functions.

Algorithm 5 Bilinear Menu

Result: Distort The Image

initialization;

target color (⟨x,y⟩) = source color (position on source plane(⟨x,y⟩))

position on source plane (⟨x,y⟩) =

if *left side < x < the zoomed left and top side < y < zoomed top* **then**
    bilinear(⟨x,y⟩), x goes left side to zoomed left, y goes top side to zoomed top, corners are ⟨left side,top side⟩,⟨zoomed left,top side⟩,⟨left side,zoomed top⟩,⟨original left,original top⟩;
end
if *zoomed left < x < zoomed right and top side < y < zoomed top* **then**
    bilinear(⟨x,y⟩, x goes left side to zoomed left, y goes top side to zoomed top, corners are ⟨zoomed left, top side⟩,⟨zoomed right, top side⟩, ⟨original left, original top⟩ ⟨original right, original top⟩)
end
if *top right* **then**
    bilinear(⟨x,y⟩, ... **if** *center left* **then**
        bilinear(⟨x,y⟩, ...
    end
    if *center center* **then**
        bilinear(⟨x,y⟩, ...
    end
    if *center right* **then**
        bilinear(⟨x,y⟩, ...
    end
    if *bottom left* **then**
        bilinear(⟨x,y⟩, ...
    end
    if *bottom center* **then**
        bilinear(⟨x,y⟩, ...
    end
    if *bottom right* **then**
        bilinear(⟨x,y⟩, ...
    end
else
    ⟨x,y⟩
end

source color(⟨x,y⟩) =

if *both x and y are round numbers* **then**
    screen color(⟨x,y⟩)
else
    bilinear(⟨x,y⟩, x goes ⌊x⌋ to ⌈x⌉, y goes ⌊y⌋ to ⌈y⌉, corners are screen color($\langle \lfloor x \rfloor, \lceil y \rceil \rangle$), $screen\ \ color(\langle \lceil x \rceil), \lceil y \rceil \rangle), screen\ \ color(\langle \lfloor x \rfloor, \lceil y \rceil \rangle)$ $, screen\ \ color(\langle \lceil x \rceil, \lceil y \rceil \rangle)$)**end**
    bilinear(⟨x,y⟩, x goes x0 to x1, y goes y0 to y1, corners are v00, v10, v01, v11) = (v11*(x-x0)*(y-y0) + v01*(x1-x)*(y-y0) + v10*(x-x0)*(y1-y) + v00*(y-y0)*(y-y1)) / (x1-x0)*(y1-y0)

4.8 Bilinear Menu Development

In this section we illustrate the development of the bilinear menu version by version.

4.8.1 Introduction

The first attempt to implement the bilinear menu were with HTML, JavaScript and JQuery since it was used intensively with menu designs, but we found out that its not possible in the time we wrote this thesis to manipulate menu items inside canvas. There are few hacks and workarounds but were unpractical and will not gives the desired results. Then we switched to java since it is a mature language with flexibility, we developed it using eclipse IDE tool.

4.8.2 BilinearMenu v1.0

In the first version we implemented The First Technique only, with changeable parameters such as height and width of area of interest and DOI for the barrel distortion. The performance was fast since the image interpolation is simple due to the use of the nearest neighbour interpolation method but generate poor quality images with obvious pixelation-edges.
The implementation was simple, the mouse pointer hover over any thing and the image will get distorted see figure 4.17 and 4.18.

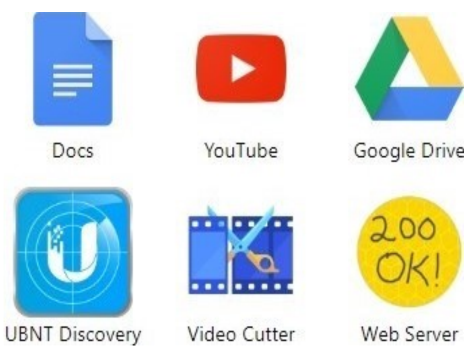


Figure 4.17: The Original Image

Figure 4.18: The Distorted Image

The three changeable parameters in the bottom of figure 4.18 can be controlled by the user to change the distortion level and the size of the area of interest.

4.8.3 BilinearMenu v2.0

The second version was aiming to improve the first version by adding a demo menu within the program so the demonstration become easier and improve the image quality by implementing the Bilinear Interpolation Method instead of the nearest neighbour interpolation method. The images becomes much more enhanced and smooth but the overall performance degraded significantly due to the large number of continues calculations caused by the bilinear method.

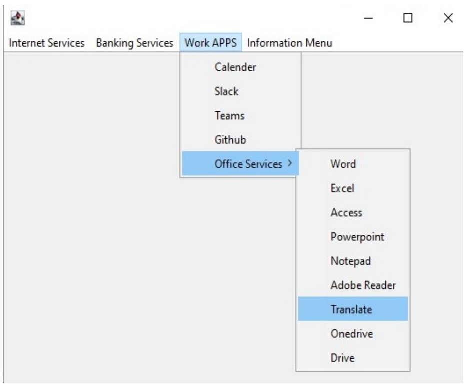


Figure 4.19: The Original Image

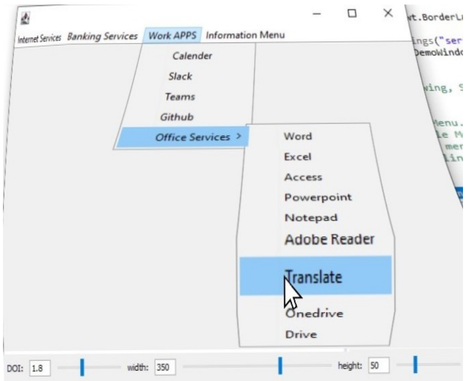


Figure 4.20: The Distorted Image

The figure 4.19 show the demo window before getting distorted in figure 4.20.

4.8.4 BilinearMenu v3.0

The third version was targeting the performance. We successfully enhanced the performance of the program to become much more responsive, by rounding the variables to integers instead of floats numbers. This made the calculations relatively faster.
We made some improvements in the source code like fixing some bugs related to the area of interest.
We added a new important feature that made the program detect what is under the mouse pointer. One way to activate the distortion is by clicking on the menu, the reason for that to make the distortion works only when it is needed see figure 4.21 and 4.22.

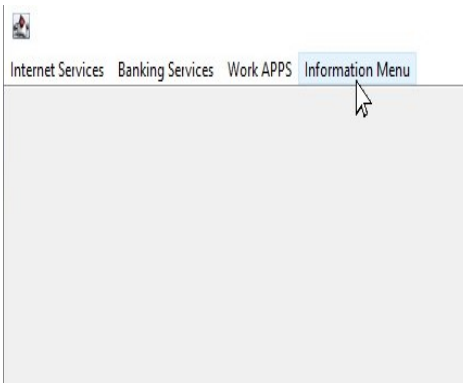


Figure 4.21: (a) The Menu Before Clicking.

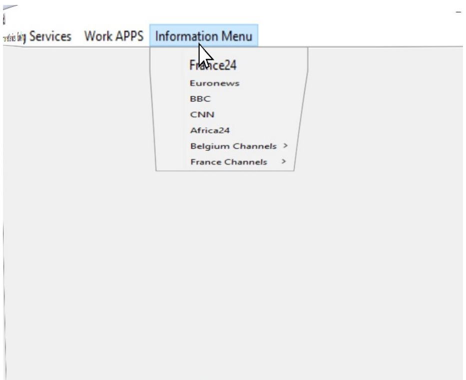


Figure 4.22: (b) The Menu Distorted After Clicking.

Another way to trigger the distortion effect is when the mouse pointer is hovering above a menu items otherwise the distortion will deactivate when the mouse leaves the menu list area and it will show a normal view without any enhancement or effects see figures 4.23, 4.24, 4.25 and 4.26.

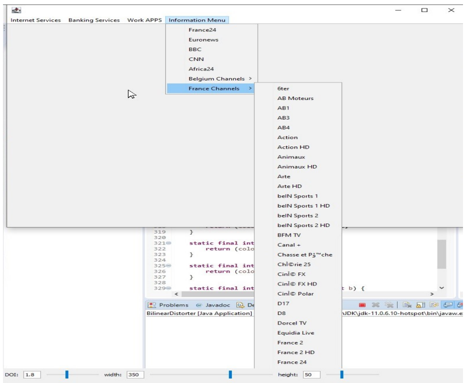


Figure 4.23: (a) The Mouse Pointer Outside The Menu Area.

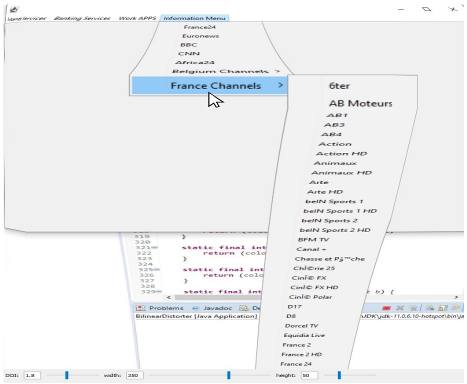


Figure 4.24: (b) The Mouse Pointer Pointing on a Menu Item.

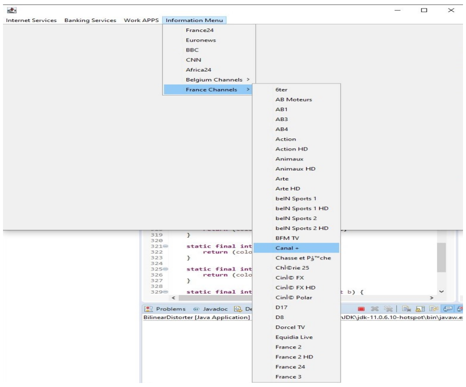


Figure 4.25: (a) The Mouse Pointer Outside The Menu Area.

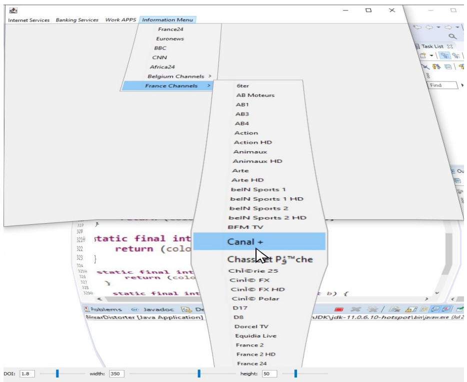


Figure 4.26: (b) The Mouse Pointer Pointing on a Menu Item.

4.8.5 BilinearMenu v4.0

In the fourth version we added the second technique to the program, with the ability to choose one of the two distortions techniques from the panel, we did some modifications to the variables of the second distortion we replaced the height and width of AOI with "number of items", which it is can be changed between 1 to 4 see figures 4.28, 4.30 and 4.31 for illustration the item selection.
In this figure 4.28 we chose to enhance one item each time, hence the menu item CNN is enhanced and focused in the middle, the next menu item Africa24 got enhanced relatively compared to the other items, note that only the next item is enhanced not the item before CNN which it is BBC, the reason for this is that since the user already pass it then he will be interested more likely in the next item more than the ones he already saw or ignore.

The second technique is relatively faster and more responsive than the first one due to less calculations, and we made some optimizations helped in that matter, the height of the AOI in the second distortion become constant and its relevant to the number of items that has been selected. The width of AOI is the width of the menu.
The AOI is aligned to the actually open menu and the mouse cursor, the horizontal center of AOI is the horizontal center of the menu, the vertical center of AOI is the mouse cursor's Y position, the left and right edge of the 3x3 grid is the left and right side of the source plane, the size of the central cell is the size of AOI multiplied by the DOI, the center point

of the central cell is the same as the center point of the AOI, the top and bottom edge of the 3x3 grid is the top and bottom of the menu.
We also made some changes to the first distortion we replaced the height with "number of items" since it will provide easier handling for the panel and for the resulted distorted images.

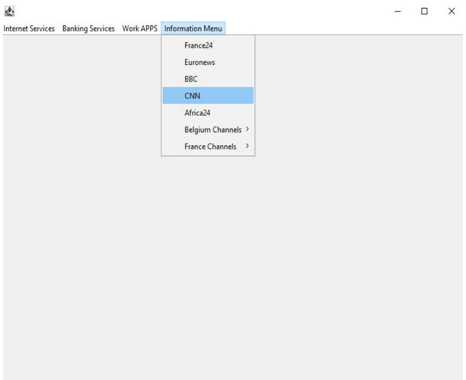


Figure 4.27: (a) The Original Menu

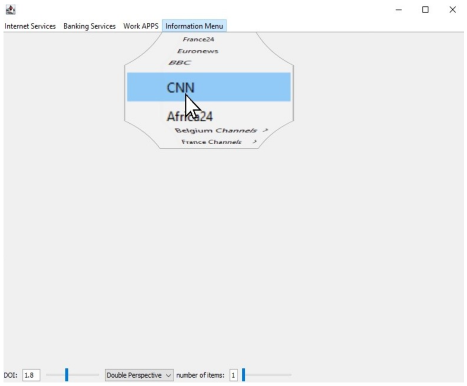


Figure 4.28: (b) The Menu Distorted

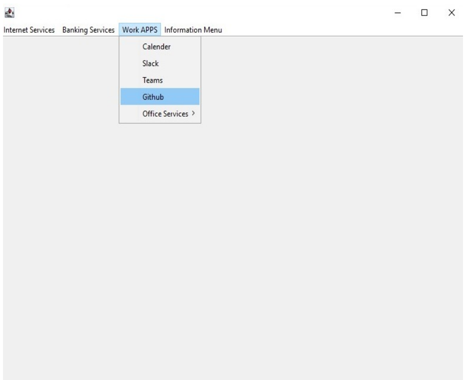


Figure 4.29: (a) The Original Menu

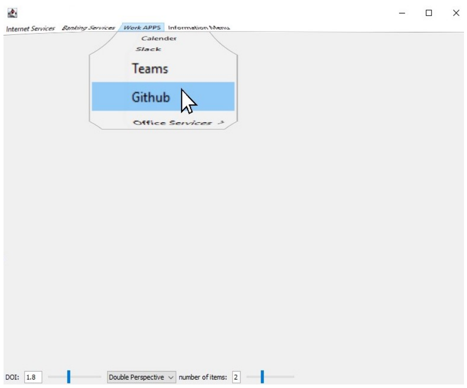


Figure 4.30: (b) The Mouse Pointer Pointing on a Menu Item.

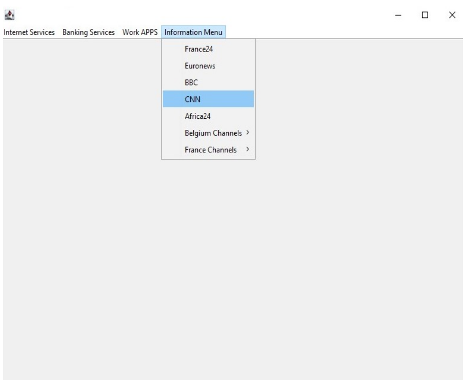


Figure 4.31: (a) The Original Menu

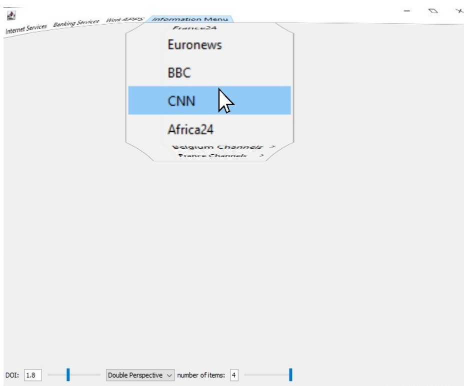


Figure 4.32: (b) The Mouse Pointer Pointing on a Menu Item.

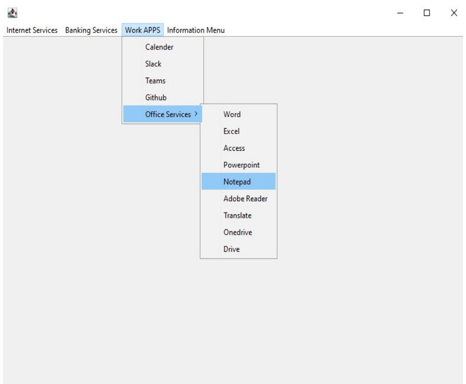


Figure 4.33: (a) The Original Menu

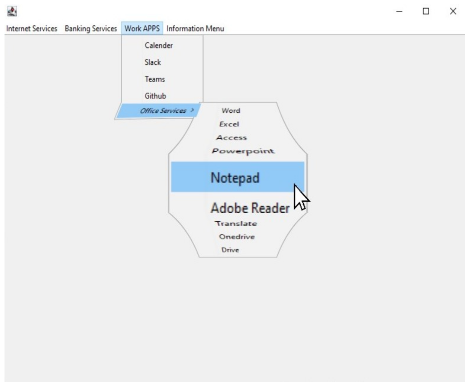


Figure 4.34: (b) The Mouse Pointer Pointing on a Menu Item.

4.9 Conclusion on The Bilinear Menu

The bilinear menu lays the foundation for a great origin for a lot of possible enhancements for menu visuals and menu selection. It is designed to be cross platform and with possibility to be experimented on wearable devices like smart glasses since these kind of devices come with small screen size and low processing powers due to the small design of such devices, our latest version of the bilinear menu require less calculations and that will lead to less resources consuming. The design is very easy to use no need for practicing and demonstration to the new users.

The two developed distortion techniques delivers a great way to visualize the menu and facilitate the menu selection with selectable options, comparing it to earlier menu types our results showing a better looking that enhance and shorting the visual search time and easier selection from other menus see figure 4.35.

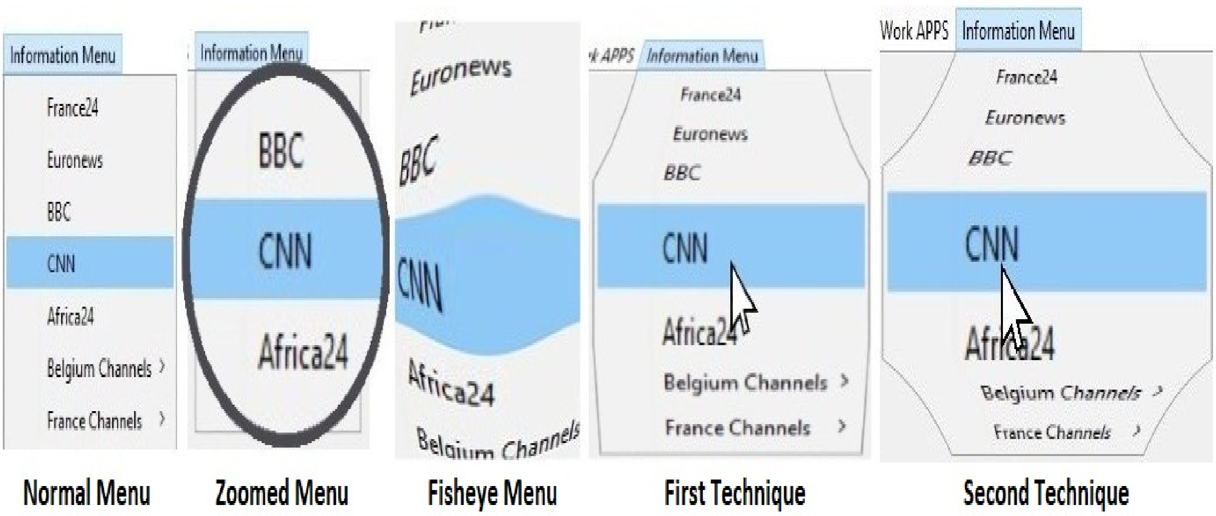


Figure 4.35: Comparison With Another Type of Menus

There is a very interesting design currently under development zoomablebook which provides a zoomable text book that can contain a big amount of data in one page and the user can zoom to the desired section. The more the user zooms the more data he sees. Its similar to google maps overviews, but with the advantage of having text, photos and videos.

We found that the closest related work to our menu is by (Agarwal, B and Stuerzlinger, Wolfgang) [17], in his work he present WidgetLens, a novel adaptive widget magnification system, which improves access to and interaction with graphical user interfaces, see 4.36 for illustrations the widgetlens.

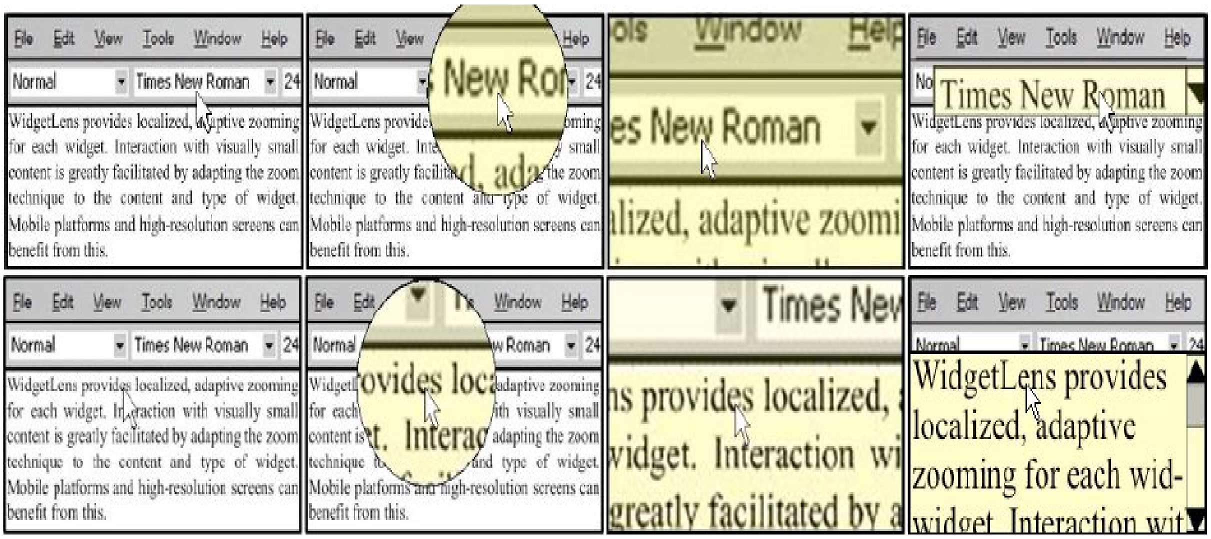


Figure 4.36: Illustrations The Widgetlens1 [12]

As illustrated the widget lens is acting like a zooming lens enhancing some parts of the selected data, the widget worked well by enhancing the desired text but the rest of the text got covered. By looking at the figure you can noticed that the enhanced part consumes more screen space to the expense of the rest of the text. Finally, there is no connection between the enhanced text and the original one which is not the case with our bilinear menu see figure 4.35. You can noticed that in the two techniques we enhanced the selected menu items without losing the space of the screen and not at the expense of other menu items.
Another Illustrations of the widget lens in figure 4.37.

Again the word "Documents" is enhanced but its covering another menu items, which it is not the case in the bilinear menu.

But our implementation is far from being perfect and suffers from some issues:

When the bilinear menu generator is launched, its splits the screen in two, one for the original image and the second one is for the distorted image. The reason for this is because we wanted to prevent the data from the original screen get bloated by the distorted data, that's will lead to unwanted distortion. But by doing some tests we successfully fixed it on the OS side. We tested our implementation on Linux since its a more

[12] Figure taken from [17]
[13] Figure from [17]

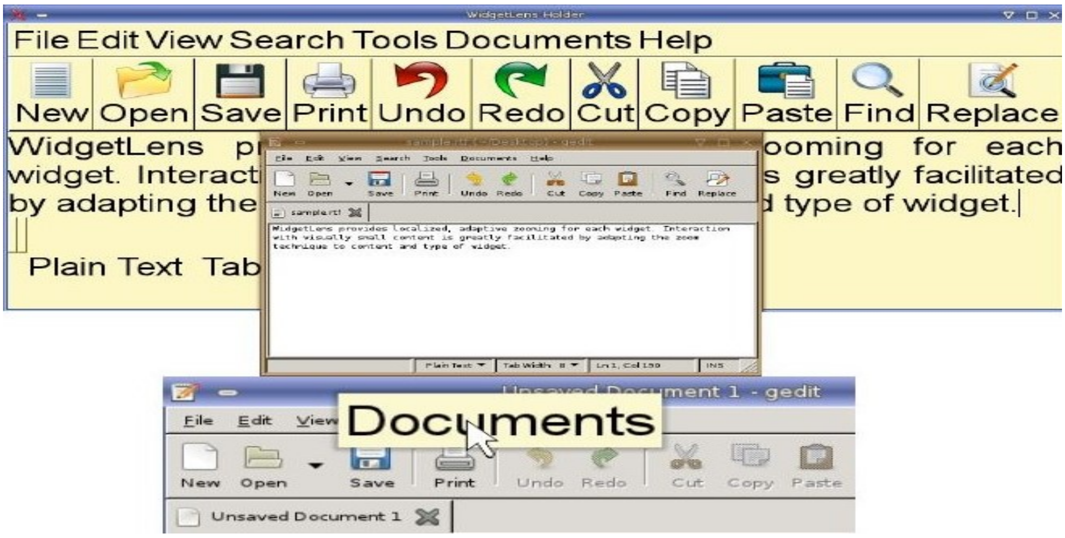


Figure 4.37: Illustrations The Widget lens [13]

controllable OS and we managed to fix the two screen issue. We also tested our implementation on some windows versions we found that it is not possible to control the screens in some versions like windows 10 but on the other hand we successfully did it on windows vista.

In general the bilinear menu delivers an original menu that is easy to use and with enhanced interface that has enhancement opportunities for future work.

Chapter 5

A Conjoint Analysis of Graphical Adaptive Menus

The fractal menu developed in Chapter 3 and the bilinear menu developed in Chapter 4 were new innovative menu interaction techniques we implemented. In this chapter, we conduct a conjoint analysis of existing graphical adaptive menus and in a local conclusion, we compare the results with ours implemented menus.

5.1 Introduction to the Conjoint Analysis

As pointed out in the Systematic Literature Review in Chapter 2, there are many graphical menus in general. This set is so large that it is almost impossible to compare them all on the same basis. An ideal comparison between these menus would consist on implementing them in a consistent framework and ask a large amount of participants to use all of them through extensive experiments determining the best selection time, the best error rate, etc. Since this ideal comparison is an utopia, we chose to conduct a conjoint analysis for determining the visual preference of graphical adaptive menus only by participants.

The *conjoint analysis* [200] is a survey-based statistical technique for determining how participants would value different attributes (*e.g.,* a feature, a function, a behavior) of different stimuli to identify the best ones. For instance, a conjoint analysis could be conducted to ask participants which advertisement they prefer from a visual point of view. This technique is largely used in marketing, sociology, etc.

In order to conduct a conjoint analysis of existing graphical adaptive menus, we will rely on AB4Web [497], an online software that performs a particular form of conjoint analysis where participants' responses to questions are used in order to compute preference measures. Each participant is presented with a series of paired tests (*i.e.,* with two options, left and right), and selects which one is the most preferred. When there is no preference, a draw is applicable. Each time a participant clicks on a variant or on the "It's a draw" push button, AB4Web records these selections and computes two measures:

A. **The Number of Presentations** (NumPresentations) represents the total amount of times that a particular menu is included in a complete testing and, thus, presented to participants. If a particular menu is randomly selected 10 times in 15 paired tests, then NumPresentations is simply 10.

B. **The Preference Percentage** (PreferencePercentage) represents the ratio between the number of times that a particular menu was preferred by participants during a complete testing and . In the previous example, if a particular menu was preferred by participants for 10 times over other menus in 15 paired tests, then PreferencePercentage is $10/15 = 66.6\%$. Inversely, the dislike percentage is $(15 - 10)/15 = 33.3\%$.

5.2 A Review of Existing Graphical Adaptive Menus

Graphical Adaptive Menus (GAMs) are those graphical menus that exhibit some adaptive behavior with respect to the context of use (user, device, environment). As for any adaptivity, there are some potential benefits such as an increased performance [201], but also disadvantages such as a varying menu [294]. For example, the selection time could be reduced in a hierarchical menu [387] or on a small screen [295]. *Spatial instability* is provoked when the initial menu is dynamically changed [184], thus potentially confusing the end user. Various graphical adaptive menus exist that partially satisfy this property. This section reviews work related to GAMs in chronological order [490]. Fig. 5.1 presents an overview of most of the menus discussed in this section.

Static menus are considered as the initial non-adaptive versions of the graphical menus, as we can find them in classical interactive applications. The menu is static in that all items are determined at design-time and physically implemented in the user interface

Probability-based menus [451] sort menu items in decreasing order of their probability of selection: the more likely a menu item is to be selected, the higher it appears in the list of items. The probability can be computed in various ways, including by machine learning techniques based on frequency, recency, etc. [195].

Frequency-based menus [346] sort menu items in decreasing order of their selection frequency, depending on the end user's actions. There are several manifestations of the frequency depending how it can be computed at run-time. For example, the **K-Menu** [299] enables the end user to select a menu item by typing its keyword in a search field: the system then presents to the user the most frequent menu items based on this keyword. The **G-Menu** [495] does the same job by gesture instead of by typing: the end user draws the first letters of any menu item on the screen surface (*e.g.,*, of a smartphone or a tablet, anywhere on screen) and the system responds by presenting the most frequent items corresponding. The **T-Menu** does the same job by touching and the **GK-Menu** performs item selection both by keyboard and gesture [495].

Split menus [432] combine a prediction window (an area with a small list of items, typically 2-3) and a static menu. Menu items are initially presented by probability or frequency, or any combination of them. Any menu item appears in either the static part or in the predicted window, which may confuse users who constantly oscillate between the two areas to find their item. The split menu was implemented for three conditions [165]: static (top four items remain static), adaptable (top four items can be moved up and down by end users), and adaptive (top four items are predicted according to user's recently and frequently used items). Static menus were found more efficient than both adaptable and adaptive menus, but adaptable menus were favored in terms of satisfaction.

Split menus with replication [184] address the problem of split menus: the prediction window remains as stated before but the second part remains unaltered, thus enabling end users to always refer to the static menu as they know it. This part never changes. When this second part contains an important amount of items, it could become reduced and scrollable either via a scroll bar (**split menu with scroll bar**) or with an arrow bar (**split menu with arrow bar**) [69].

Smart menus [251] initially display the most commonly selected items and all the available items by clicking on the arrow at the bottom of the smart menu to expand it. Smart menus track how often an end user invokes each item, in order to predict frequently used and recently used menu items. Smart menus provide beginners with a starting guided path toward learning a new user interface.

Gapped menus [169] restore the spatial stability to smart menus. Gapped menus present the static menu with only predicted items, leaving a blank space as a gap for unpredicted items. Clicking on the arrow at the bottom of the gapped menu displays the entire static menu.

Bolding menus [387] present menus with frequent items in bold font. Since there is only one level of bolding, a menu item is either presented in bold if frequent or not if infrequent. There is no middle.

Highlighting menus [387] highlight frequent menu items. For example, Gajos et al. [184] highlight predicted items by colouring their background in pink, whic is called a **pinked menu**. A study analyzed the effect of menu size on user satisfaction of five menus with different highlighting [23]:

- An adaptable menu where participants could move up and down predicted items;
- An adaptive double split menu divided into a section applying a frequency-based scheme, a section apply a recently-used scheme, and the unpredicted items;
- An adaptive/adaptable bolding menu where participants could move predicted items up and down after 50 selections;
- An adaptive/adaptable minimized menu, which is a smart menu divided into a section by frequency, a section by recency, and the rest of items to be displayed on demand with a moving facility; and a mixed-initiative menu [234], which is a bolding menu letting the participant to choose between frequency and recency. Indeed, te most frequent items are not necessarily the most recent, and vice versa.

In small menus, the minimized condition was the most preferred menus, followed by the adaptable, and bolding menus. The adaptive split and mixed-initiative menus were the least preferred menus. In large menus, the mixed-initiative was the most preferred, followed by the minimized menu; the adaptable menu was the least preferred followed by the adaptive split menu.

Square Menus [20] re-layout menu items into squares to enlarge their selection area for better performance. They propose the Search, Decision, and Pointing (SDP) model to improve the pointing performance, especially for experts. It is a promising solution compared to traditional linear menus and to pie menus.

Morphing menus [129] change the font size of each menu item depending on its prediction: the more important the menu item is, the larger its font size becomes. While morphing menus preserve the ordering of items, they facilitate selecting accurately-predicted items, but complexify selecting items with low or inaccurate prediction. Like for square menus, it is expected that the item selection is made faster when the activation area is enlarged. The enlarged activation area and zero delays improve item selection by up to 29% in comparison to traditional methods. However, cascading menus are not an easy way to deal with hierarchies on tiny screens, they need motor abilities.

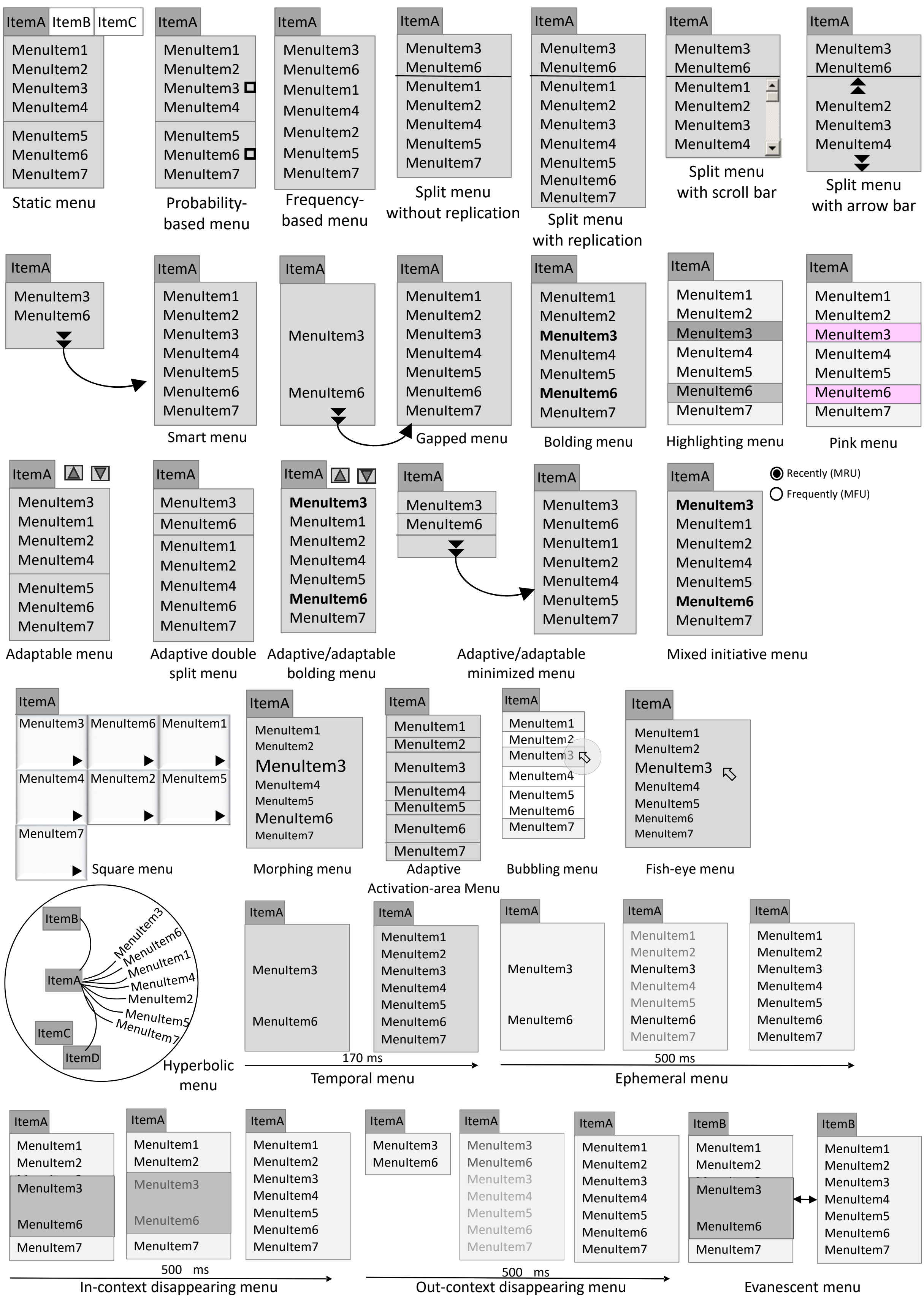


Figure 5.1: Repertoire of menus investigated.

Adaptive Activation-Area Menus (AAMUs) [465] are a form of adaptive morphing menus where the activation area for predicted items is dynamically resized for menu navigation. AAMUs, whether they are used in isolation or combined with **Force-field menus** [20], outperformed the static menu.

Bubbling menus [476] are pull-down menus where frequently used items are selected by directly jumping to them one by one. The bubble cursor dynamically changes its size as the cursor moves and selects the target within the closest distance. Directional mouse-gesture techniques accelerate reaching predicted items.

Fish-eye menus [69] display menu items with a font size that increases or decreases depending on the distance with respect to cursor position: the closer the item is from the cursor, the larger its font size becomes. Similarly to AAMUs [465] and Bubbling menus [476], they are able to increase the activation area of any item.

Hyperbolic menus [290] exploit a Focus+Context technique for displaying large hierarchies of menus and sub-menus. It displays several hierarchy levels at once according to a hyperbolic tree which minimizes screen usage. When a particular node of the hyperbolic menu is activated, the items of its sub-menu are expanded and arranged in a hyperbolic way. Selecting any item in this sub-menu collapses the former node and expands a new one, and so forth.

Temporal menus [295] display menu items in two stages: at opening, only predicted items, and after a delay of 170 ms, non-predicted items appear. This menu maintains spatial stability, thus helping the end user to maintain a mental model of the menu.

Ephemeral menus [168] is an adaptive temporal menu where the gradual onset was used in order to display non-predicted items. At opening the menu, user finds predicted items and after a delay of 500ms remaining items appear gradually. Lee and Yoon [295] examined a **dynamic menu**, where important items appear immediately when a menu is opened (abrupt onset) and those outside of the subset appear after a delay.

In-Context Disappearing (ICD) [?] have another display scheme: at opening the static menu, the end user finds a superposition of the full list of items with the prediction window. This window contains 2-3 items, like in the split menu, and disappears gradually. The presentation of predicted items in the prediction window remains homogeneous, thus preserving spatial stability.

Out-of-Context Disappearing (OCD) approach is the inverse [?]: at opening, the prediction window is immediately displayed with the predicted items, like in a split menu; after 500 msec [168], the complete menu is gradually displayed from the back, thus replacing the prediction window.

Evanescent menus [?] are GAMs where the prediction window is superimposed to the static menu and then progressively made transparent to reveal the entire menu.

Step-by-Step Menus [88] display at each level of the hierarchy the prediction window and offers to select the menu item leading to the next level of the target path that is the most likely to be selected.

Shortcut Menus [?] display the target item in a prediction window at the root level of the hierarchy. A comparison between two GAMs, *i.e.,* shortcut menus [?] vs. step-by-step menus [88], shows that the hierarchical navigation in smartphones is improved: step-by-step menus preserve the consistency with the static menu through level-by-level navigation to reach the predicted item. Shortcut menus directly moves the end user to the predicted item in its very right location, thus reducing if not eliminating the effort for navigating in the menu hierarchy.

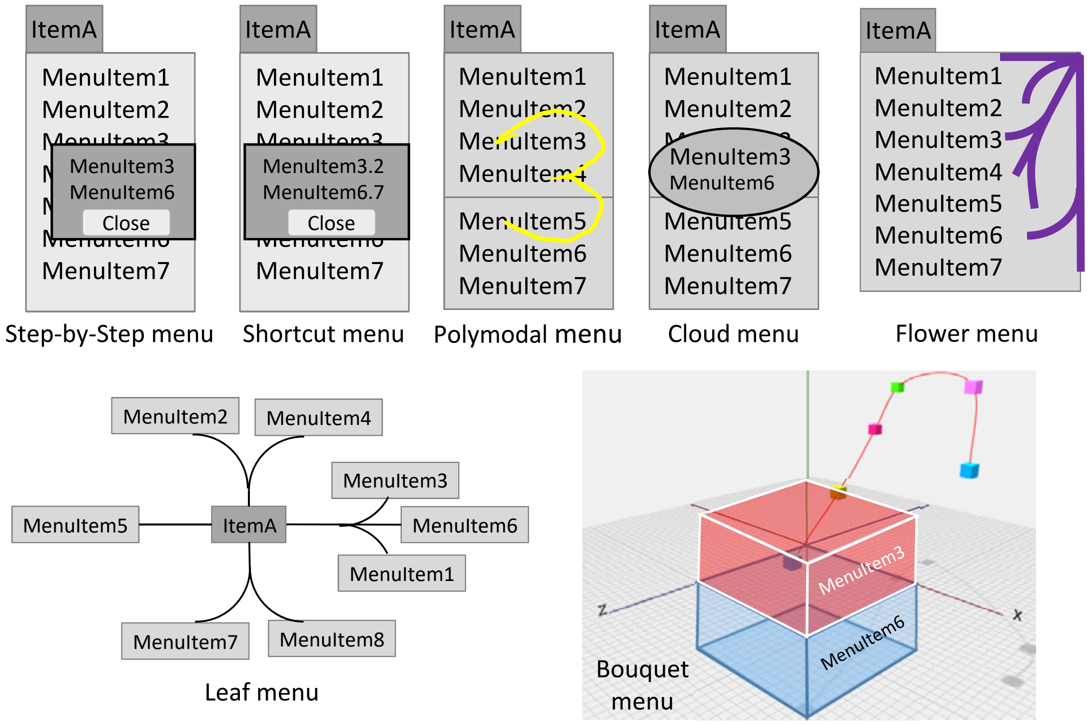


Figure 5.2: Repertoire of menus: other menus.

Polymodal Menus [91] are graphical adaptive menus in which any menu item can be selected graphically (by pointing), vocally (by voice recognition), tactilely (by touching), gesturally (by issuing a gesture representing the menu), or any combination of them. Predicted menus are rendered by graphical or vocal prediction window.

Cloud Menus [489, 490] are an adaptive split menu where the predicted items are arranged in a word cloud with a location consistent with their corresponding position in the static menu and a font size depending on their prediction level. Cloud menus reduce item selection time and error rate when prediction is correct without penalizing it when prediction is incorrect, compared to two baselines: a non-adaptive static menu and an adaptive linear menu.

Dual-motion Menus [89] are adaptive split menus that constantly and dynamically performs adaptivity by promoting the most predicted item and demoting the least predicted item either in the menu itself or in a separated prediction window, depending on the need for spatial stability. A promoted item which is no longer subject to any selection progressively becomes demoted and vice versa, therefore inducing a dual motion of items. Promotion and demotion are therefore the underlying mechanisms.

Flower menus [52] extend marking menus with opportunity to draw straight lines or curved ones into the eight cardinal directions of a compass, which can optionally be terminated by bended, cusped, and pig-tail endings.

Wavelet Menus [177] promote with a circular and linear layout, while **Leaf Menus** [419] are optimized for item accessibility.

Bouquet Menus [105] consist of a marking menu offering flicks and marks from an origin towards the directions of the eights octants of a cube, thus generalizing the Flower Menu [52] into three dimensions.

Stacked half-pie menus [226] display menu items as circles in half pie on a tabletop surface. This interaction technique tends to make this design unlimited in terms of menu depth and breadth while still maintaining the initial form of the menu. This menu is limited for small screen devices like smartphones where there is not enough space on the screen. In addition, the navigation in the pie menu may be a constraint for novice users.

PocketMenu [395] exploited the idea of changing the modality for menu selection: menu items are laid out along the border of the touch smartphone within the hand comfort zone, tactile features guide the hierarchical navigation, a vibro-tactile feedback with speech allows identifying the items non-visually. This interaction technique is particularly useful for end users with visual disabilities.

We now review GAMs according to their Bertin's visual variable [490].

5.2.1 Position-Changing Menus

GAMs where the position of predicted items changes depending on their prediction are the primary form of adaptivity namely in probability-based menus [451], frequency-based menus [346], split menus with replication [184] or without [432] as their most representative examples. These menus are criticized for endangering their spatial stability, which may confuse end users with perpetually changing positions, thus preventing them from building a permanent mental model based on their layout. Split menus with replication partially escape from this drawback: the end user may first check whether the desired item belongs to the prediction window and, if not, browse the normal menu. The static part is position-invariant whereas the prediction window is not. Other suggested forms of position-changing menus are (Fig. 5.3a): the **pushpin menu** (a split menu where predicted items can become permanently placed by locking them with a pushpin) and the **time-based menu** (a multi-split menu where predicted items are sorted in chronological time). This induces a new category of split menus where different portions could contain different split areas according to different prediction scheme: single split, double split (as in Fig. 5.1), or multi-split menu. Later in this exploration, other families of variable-changing menu will inevitably affect the position and thus the spatial stability: menus with layout and/or a selection area modified by the prediction. Adaptive activation-area menus [465] are representative as they change the selection area depending on the prediction of each item. Similarly, morphing menus [129] or square menus [20], enlarge this zone by resizing it as a rectangle or a square.

5.2.2 Orientation-Changing Menus

Any technique changing the orientation of predicted items falls in this category. A typical example concerns hyperbolic menus [290], which are not a graphical adaptive menu per se. But the visualization technique can expand sub-trees unveiling predicted items and collapse sub-trees containing unpredicted items. Since items are distributed along a hyperbola, their orientation changes as the tree is expanded or collapsed. This visual variable has never been subject to any investigation as far as we know, although it has some potential to be further examined. For instance, label orientation, i.e., horizontal, vertical, angular, could be considered for emphasizing a predicted item, especially in cloud menus [489]. When a **localized menu** is adapted according to the end user's culture and language, items can be flipped automatically [267] between a Left-to-Right (LTR) format as used in Western cultures and a Right-To-Left (RTL) format as used in Arabian languages, or from Top-To-Bottom (TTB) to Bottom-To-Top (BTT), thus giving rise to a **flippable menu**. Similarly to twisting icons [396], **twisting menus** briefly change the orientation of predicted items by twisting them with respect to the horizontal, while in **rotating menus**, predicted items are subject to a 360° rotation. The speed and the frequency to

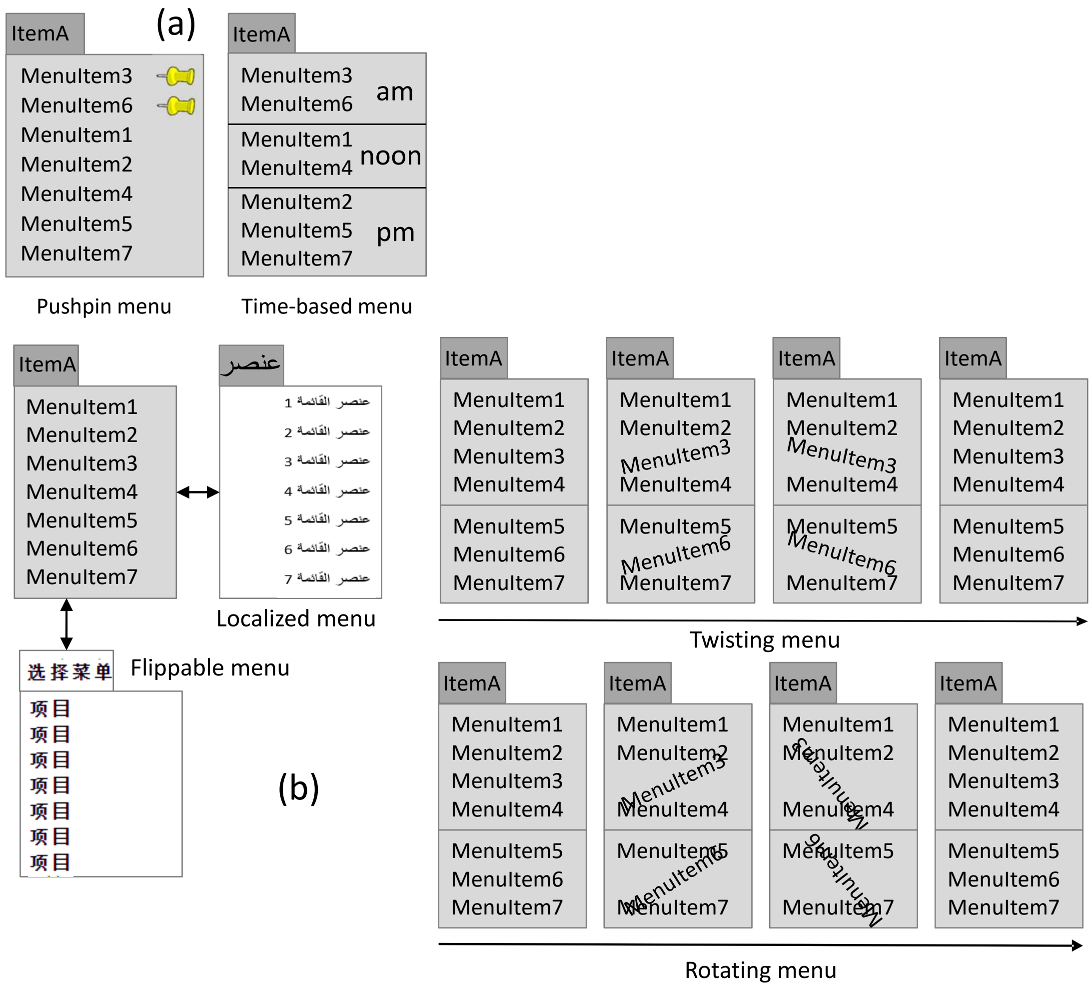


Figure 5.3: Menus classified by visual variable: position-changing (a) or orientation-changing (b).

be used for these animations are yet to be determined, although we know that the speed should last more or less 500 msec and the frequency should not repeat the same animation too many times in order not to induce any boredom.

5.2.3 Size-Changing Menus

Four types of actions can modify the menu size: modifying the selection area (such as in morphing menus [129]), adding another menu part (such as in split menus with replication [184], adding another user interface element for shortcutting the menu hierarchy (such as in shortcut menus [?], and translating/localizing their labels (e.g., translating an English item produces a longer item in French and even longer in German). Size can be further decomposed into four sub-variables depending on how many dimensions are considered: line (for a one-dimensional change), surface (for a two-dimensional change), multi-surface (for a $2D1/2$ change when a surface is projected onto another one), and volume (for a three-dimensional change). The **pulsing menu** (Fig. 5.4a), inspired by the pulsing icons [396], pulse forward predicted items, thus changing their size until they return to their initial state. Pulsing is perceived less intrusively than strong animations found in a rotating menu.

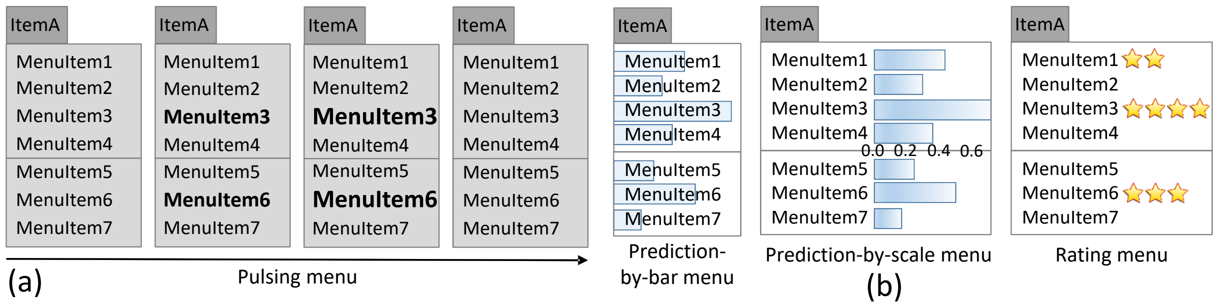


Figure 5.4: Menus by visual variable: size-changing (a) and shape-changing (b).

5.2.4 Shape-Changing Menus

While the rectangle remains the predominant shape for menus, other shapes have been considered, but not necessarily for adaptivity. For instance, square menus [20] delineate a rectangular area for each menu item appearing in a squared menu to improve item selection. **Radial menus** [313] present menu items according to a (semi-)circular layout, showing that for different breadths and depths, they can be superior to their equivalent cascading menu counterpart. What has not been investigated so far for adaptive menu are shape-changing menus: depending on the amount of predicted items and the prediction scheme, the menu could gracefully evolve from one shape to another that is more suitable for displaying the items as they are: circle, oval, square, rectangle, pentagon, hexagon, heptagon, octagon, parallelogram, trapezium, etc. Some of these shapes have been particularly exploited for gesture-based menus. Shape-changing menus should be possible to highlight predicted items when hovering for instance: the shape of the menu bar would change from a square to a somewhat rounded shape or other shape when you hover over them, thus revealing items only on demand. The menu shape is also affected when the prediction window is moved onto a separate area, which makes it more distinguishable by end users [508], as in split menus with replication [184]. This prediction area is itself subject to shape-changing: line (e.g., for expressing the frequency of a menu item in frequency-by-line menus), histogram, square, rectangle, circle as in cloud menus. Fig. 5.4b suggests three shapes to convey the likelihood of predicted items: a bar superimposed to the item in the **prediction-by-bar menu**, a bar juxtaposed to the item with a scale such as a histogram in the **prediction-by-scale menu**, or a rating scale in the **rating menu**. The last two menus considerably increase the menu width.

5.2.5 Value-Changing Menus

Any user interface highlighting technique can be applied to predicted items in value-changing menus: bold, italics, underscores, boxes, capitalization, single or double quotation marks, alternate fonts, emphasis techniques in computer text, or any combination of the preceding. Color is considered in the next family. Highlighting menus [31] highlight predicted items by contrast, which is viable since up to three levels of contrast are usually distinguishable by end users. Bolding menus [31] apply a bold font to predicted items. These two menus were introduced for a comparison with split menus, adaptable menus, and traditional menus on a desktop [165]. This study showed that the adaptable menu outperformed the other menus in terms of overall performance and subjective satisfaction. The split menu was estimated sub-optimal, especially when the predicted frequency changed. The bolding menu was not significantly better than when working with variations in the traditional menus, but was preferred by end users since less sensitive

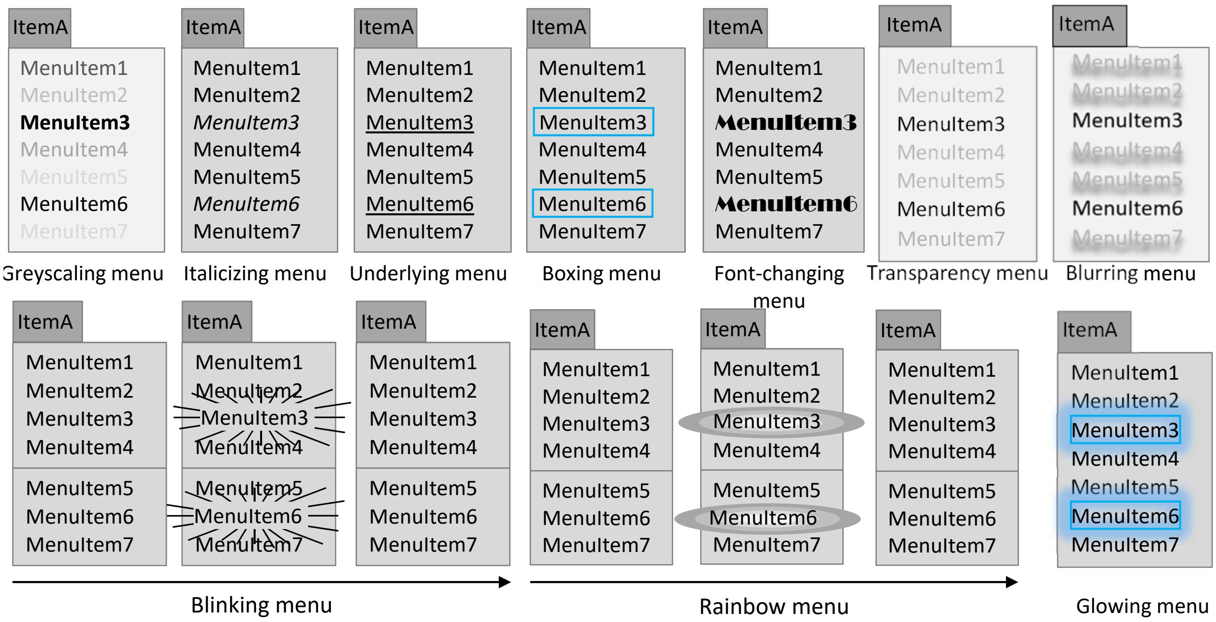


Figure 5.5: Menus by visual variable: value-changing menus.

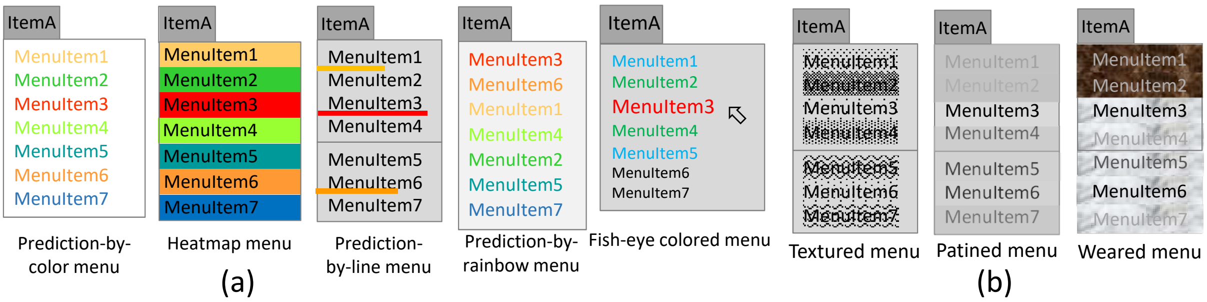


Figure 5.6: Menus by visual variable: color-changing (a) and texture-changing (b).

to the variations of prediction that its counterparts. Fig. 5.5 suggests several forms of value-chaining menus: the **greyscaling menu** where items are greyscaled depending on their prediction (the more greyscaled, the less predicted), the **italicizing menu** where predicted items are formatted in italics, the **underlying menu** where labels of predicted items are underlined, the **boxing menu** where predicted items are surrounded by a visual frame, or the **font-changing menu**, where a different font is used for representing predicted items as opposed to unpredicted ones. This technique heavily depends on the font legibility (e.g., sans-serif fonts are more legible on screen than serif fonts) and their recognizability (some popular fonts can be recognized but not all). Many other visual effects could be envisioned but their effectiveness and efficiency is not demonstrated yet: unpredicted items could be subject to a transparency stencil in a **transparency menu**, blurred in a **blurring menu**, subject to blinking with a small rate in a **blinking menu**, animated with a rainbow in a **rainbow menu**, or glowing to make them more salient in a **glowing menu** as in Phosphor Widgets [63].

5.2.6 Color-Changing Menus

Color-changing menus [396] were proposed in order to reduce visual search time: frequently used items are highlighted by changing their background or foreground (font) color or both. The study compared color menus to fish-eye menus [69]: on smartphones, color menus require a lot of concentration from the user, especially when predicted item is located at the bottom of the screen. The user must scroll the window to see the item. For instance, a **prediction-by-color** menu (Fig. 5.6a) would display all items with a color code associated to the prediction level, while in a **prediction-by-rainbow menu**, menu items would be sorted and displayed according to a rainbow color scheme. We suggest that a **heatmap menu** color its items in a heatmap depending on their frequency of use. Contrarily to split menus where frequent menu items are first presented or to morphing menus where frequent items are enlarged, heatmap menus do not change their layout. The perception of the heat map may induce some usability problems, since rainbow scales are not always correctly interpreted. Heatmap menus could be presented in two ways: the heatmap shows all dots generated by interaction on a menu item (e.g., a mouse click, a finger touch, an eye saccade captured by an eye tracker), or on the entire item region by aggregating the dots and attribute them to the menu item that actually generated them.. Since visual variables are independent of each other, more than one variable could be used to reinforce the adaptivity when critical. For instance, a prediction-by-color menu sorts items in decreasing order of prediction rendered on a rainbow scale: both position and color are altered. A **fish-eye colored menu** plays mainly with size (since the size of items depends on the cursor position), but also with position because the change of size implies a change of absolute position, but not a change in the item ordering.

5.2.7 Texture-Changing Menus

A **textured menu** results from applying any change of the texture of predicted and/or unpredicted items without changing the rest of its format. Patina [331] dynamically creates a heatmap depicting how frequently a user interface element, such as a menu item or an icon, is used. The Patina is overlaid to the initial user interface, for instance a toolbar, an icon palette. This inspires a **patined menu**, where menu items are overlaid with a transparency texture depicting their usage: the more predicted, the more visible. Instead of progressively hiding unpredicted items, which may prevent end users from appropriately locating them, we can also introduce a **weared menu**, which change their texture based on their usage frequency and recency.

5.2.8 Motion-Changing Menus

Although the time dimension expresses a fourth sub-variable of position (*i.e.,* add t after x, y, and z), *time* is typically involved in the motion variable, especially in information visualization. Motion could be even further decomposed into sub-variables, such as: direction, speed, frequency, rhythm, flicker, trails, and style. Transposing temporal menu to smartphone is not obvious because all items cannot be displayed on a single screen. Any predicted item located on the subsequent screens requires a cognitive effort to explore the whole set of items. All menu types relying on animation of predicted items fall in this category: ephemeral menus, In-Context Disappearing menus, Out-of-Context Disappearing menus, Evanescent menus.

5.2.9 Fractal menus

A fractal menu is a responsive adaptative multi-platform context menu where items are organized to increase its effectiveness and efficiency and where selection depends on the user's preference and usage.

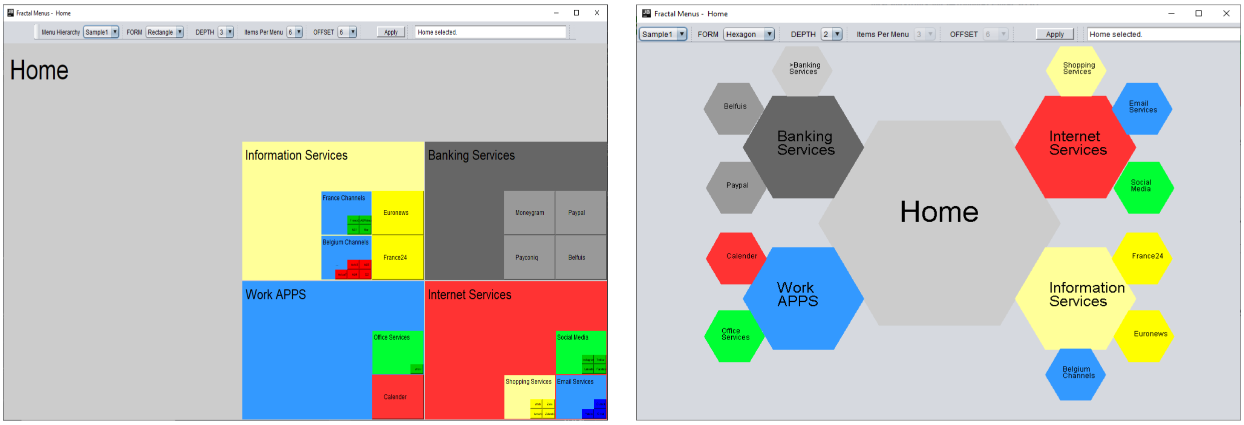


Figure 5.7: Rectangular and Hexagonal Fractal Menus

Fractal menus implement optimized selection techniques to directly select deeper items, configurations to modify the menu's view depth or breath(items) per view screen.

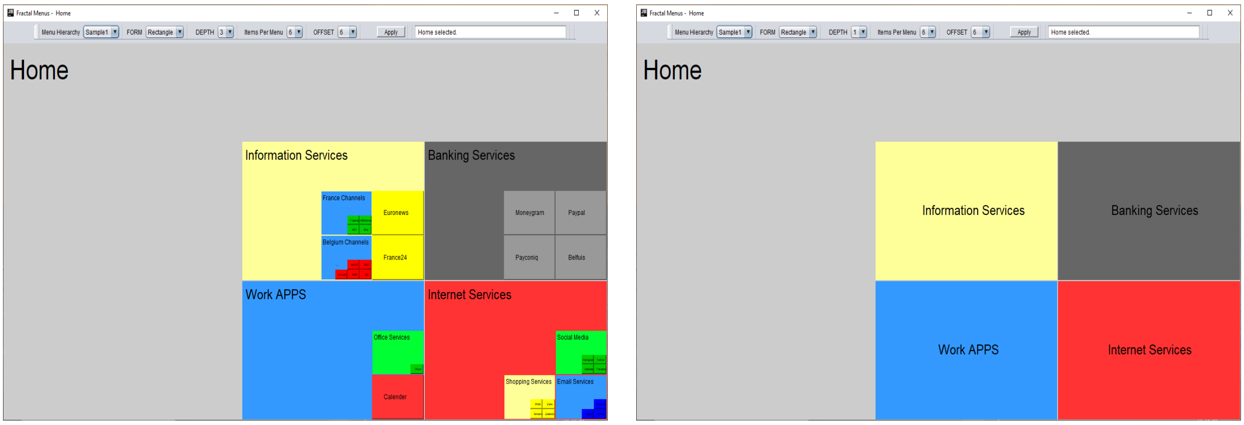


Figure 5.8: Rectangular Fractal Menu Depth Modification

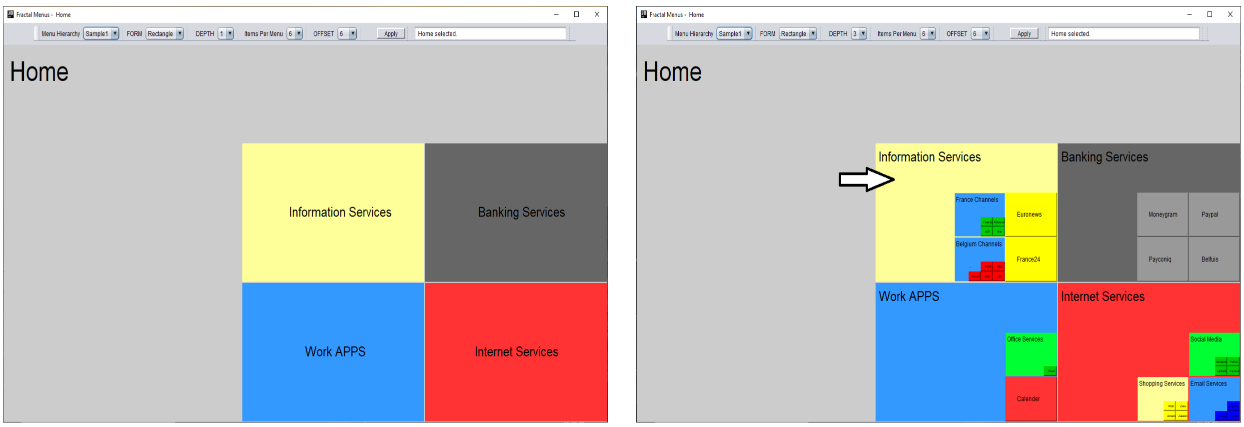


Figure 5.9: Rectangular Fractal Menu Breath Modification

5.2.10 Bilinear menus

Representing enlarged and enhanced view for the item or items on which the mouse is pointing. The menu stays original but the rendering is changed. The menu is restored to its original view if the mouse pointer leaves the activation area. The techniques uses configurable values the user can change to suit his needs like the dimensions of the activation area or the number of items enhanced per render.

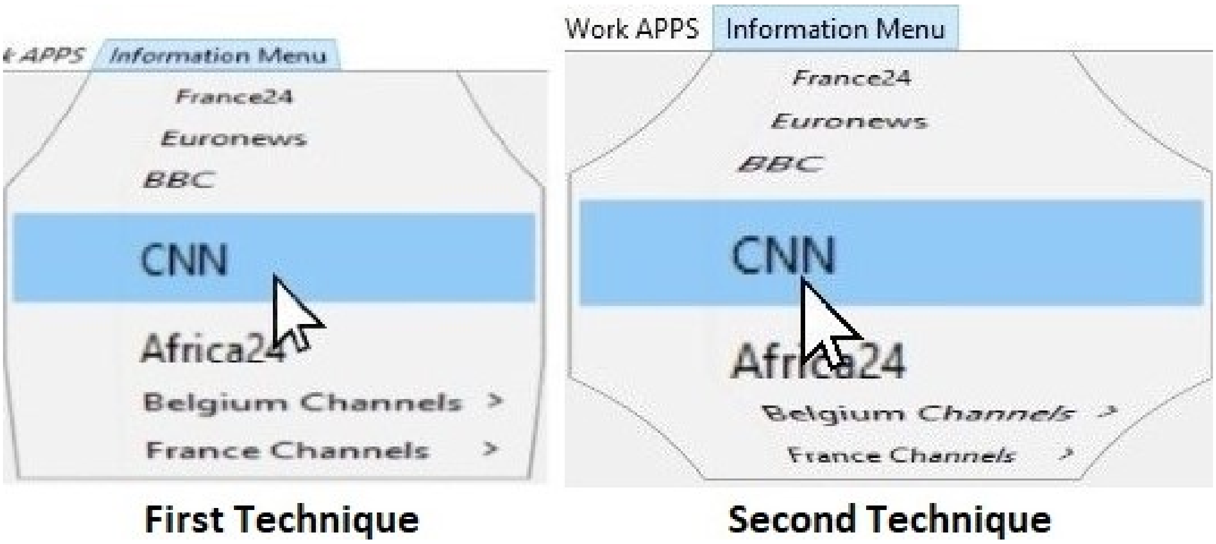


Figure 5.10: Bilinear menus

5.3 A Preliminary Conjoint Analysis

To get a first idea of how end users appreciate GAMs, a conjoint analysis is conducted to determine their preference. Preference is often compared to performance [365]: end users sometimes prefer user interfaces that are not necessarily performant and they also become efficient with user interfaces they are not satisfied with. In the previous section, 49 GAMs were identified. Unlike other research experiments, we did not choose to present one menu at a time and asking each participant to evaluate it separately: such a repetitive task would lead to participant's fatigue. Instead, we chose to conduct a conjoint analysis with several advantages: participants do not need to remember the assessment given to previous menus; this avoids inconsistent and conflicting results; the assessment is simplified by avoiding each participant to evaluate different measures for each menu; instead of evaluating qualitative measures on a Likert scale and/or quantitative measures for one menu at a time, the participants are presented with a pair of two GAMs to be compared.

5.3.1 Method

Apparatus

We used AB4Web [490] where two GAMs were randomly selected in a pair from the pool of 49 menus and presented individually to the user so as to select by clicking on the most preferred one from a visual viewpoint. For every positive selection, one point is given to the global score of the menu. If the participant is undecided, there is the possibility to click on the "It is a draw" push button, no point is assigned and two new menus are further presented. Since the pool comprises 49 graphical adaptive menus, a complete comparison would generate $49^2 = 2,401$ menu pairs to be compared per participant, which is of course prohibitive. Therefore, for each participant, the system randomly generates 50 menu pairs. In order to prevent responses biases, the menus are also displayed in a random order and each menu pair is verified as being unique per participant to avoid any duplicate.

Stimuli

For each of the 49 graphical adaptive menus in the pool, a high resolution GIF image was created depicting the menu's behavior based on Figs. 5.1, 5.2, 5.3, and 5.6. For each menu involving some visual effect such as an animation, an animated GIF file was produced respecting the guidelines issued, such as 170 msec for the temporal menu [295], 500 msec for the ephemeral menu [168]. For new menus, similar timing was chosen and the animation was repeated in a loop.

Procedure

The conjoint analysis requested each respondent to compare 50 randomly generated pairs of menus and select the most preferred. No time constraint was imposed to participants. Each experiment takes approximately ten minutes.

Participants

Participants were recruited from a mailing list maintained at Université catholique de Louvain. No compensation was offered to volunteers. The experiment took place remotely via the AB4Web system with the fifty menu pairs. Before starting the experiment, the participants were asked to provide some personal information for statistics such as year of birth, gender. Eighty-seven participants conducted the experiment from nine different countries speaking different languages (*e.g.,* English, French, German, Dutch, Spanish, Portuguese, Romanian, Arabian). 166 persons participated to this conjoint analysis. Thus, the amount of data generated by this experiment is 166 participants $\times$ 50 menu pairs = 8,300 samples.

5.3.2 Results and Discussion

Table 5.1, continued in Table 5.2, reproduces the two measures for all 49 GAMs sorted in decreasing order of their preference percentage. Fig. 5.11 graphically depicts the distribution of these GAMS in decreasing order of their preference percentage with the number of presentations in parentheses. From these tables and graphic, we make the following observations:

- The greyscaling menu is the winner of the conjoint analysis with a maximal preference percentage of 77%, which means that this menu was preferred by participants in 77% of cases when compared to all other 48 menus in the pool. This menu was very much appreciated because the way it highlights predicted items is the simplest one and the most stable: greying unpredicted items and keeping predicted items in normal font.
- The rotating menu is the looser of the conjoint analysis with a minimal percentage of 9%, which means that this menus was disliked by participants in almost 91% of the cases compared to all other 48 menus in the pool. This menu was very much depreciated because it highlights items in the most "violent" way, *i.e.,* by rotating

Ranking	**GAM**	NumPresent.	PreferencePer.
1	Greyscaling menu	190	77%
2	Transparency menu	242	72%
3	Highlighting menu	215	70%
4	Rating menu	228	68%
5	Underlying menu	218	64%
6	Pushpin menu	208	63%
7	Fish-eye menu	203	59%
8	Boxing menu	193	59%
9	Bolding menu	186	55%
10	Split menu with replication	211	55%
11	Morphing menu	221	53%
12	Prediction-by-bar menu	214	50%
13	Smart menu	219	49%
14	Patined menu	199	49%
15	Frequency-based menu	190	49%
16	Probability-based menu	213	48%
17	Prediction-by-line menu	199	48%
18	Font-changing menu	213	47%
19	Blurring menu	212	46%
20	Split menu with scroll bar	198	42%
21	Split menu without replication	192	42%
22	Glowing menu	223	42%
23	Bubbling menu	208	41%
24	Pulsing menu	207	40%
25	Prediction-by-scale menu	178	39%
26	Fish-eye colored menu	220	39%
27	Blinking menu	215	39%
28	Ephemeral menu	195	38%
29	Time-based menu	204	38%
30	Cloud menu	206	37%
31	Step-by-step menu	207	35%
32	Rainbow menu	228	35%
33	Split menu with arrow bar	217	33%
34	Twisting menu	192	32%
35	In-context disappearing menu	234	32%
36	Temporal menu	228	32%
37	Square menu	202	31%
38	Italicizing menu	223	30%
39	Out-context disappearing menu	205	29%
40	Tree menu	220	27%

Table 5.1: Results of the conjoint analysis: menus in decreasing order of their preference percentage (1/2).

predicted items, thus making them them difficult to read and too much salient with respect to others.

- Since GAMs were randomly selected in each pair, they were not presented in an equal way over the whole experiment. Indeed, the number of presentations is varying (M=208.2, Md=208, SD=14.10), although AB4Web attempts to balance the distribution while guaranteeing that any participant is presented with a pair of GAMs only once (no repetition).

- Only twelve GAMs were preferred by participants with a percentage above or equal to 50%, which represents a fourth ($\frac{12}{49}$=24.49%) of the entire pool. This suggests that most other GAMs are not preferred for at least one reason. Fig. 5.11 really conveys the point where the dislike percentage exceeds the preference percentage, at the level of "'Prediction-by-bar menu" (#12). The preference percentage is generally low (M=41.57%, Md=39%, SD=14.82%) and rapidly decreases after this threshold.

- If we admit a threshold of 30% instead of 50%, then 39 GAMs belong to this confidence interval, finishing with the "Italicizing menu" (#39). This represents roughly 80% of the GAM pool.

- The six most preferred graphical adaptive menus all preserve spatial stability, being both position-invariant and orientation-invariant. The fish-eye menu (#7) is the first one appearing in the list with position variance. If we exclude this fish-eye menu,then we can go until the split with replication menu (#10), which somewhat preserves the same stability. Many other menus with spatial stability are also in the top list. This indicates a strong preference for spatial stability.

- These fourteen first menus mostly play with value-changing capabilities. Out of these fourteen menus, the rating menu (#4) and the pushpin menu (#6) are the only two instances introducing a shape-changing (with the rating bar and the pushpin, respectively) while the fish-eye (#7) and the morphing menu (#11) are the only one with size-changing after them. This reveals a preference for menus which also preserve physical stability, with shape and size after.

- Surprisingly, the patined menu (#14) is the first occurrence of a texture-changing menu, long before any other of the same category (*e.g.,* the weared menu appears at the #38 place).

- In the category of motion-changing menus, the smart menu (#13) is well placed contrarily to previously expressed criticism [251]. But this menu is mostly a two-state menu and is the first one below the 50% barrier with a preference of 49%. Among all genuine motion-changing menus, the ephemeral menu is the great winner (#28), followed by the cloud menu (#30). Other members of this category come long after: ICD (#35), temporal menus (#36), OCD (#39), and evanescent menus (#44) and with low preference percentages. ICD seems to be preferred over its inverse technique, *i.e.,* OCD, because the motion produces an appearance in the already existing menu, which is more appreciated than making the predicted items standing out first, and then making the context appearing.

- Color-changing menus are not appreciated at all, not because participants do not like or perceive colours, but because they cannot attach any interpretation. The

Ranking	GAM	NumPresent.	PreferencePer.
41	Heatmap menu	190	27%
42	Polymodal menu	181	27%
43	Weared menu	226	26%
44	Evanescent menu	201	26%
45	Prediction-by-color menu	212	25%
46	Prediction-by-rainbow menu	199	23%
47	Hyperbolic menu	218	22%
48	Flower menu	191	18%
49	Rotating menu	199	9%

Table 5.2: Results of the conjoint analysis: menus in decreasing order of their preference percentage (2/2).

first instance is the prediction-by-line menu (#17), probably because the color is supplemented by the bar length indicating the prediction. In other menus, the color coding scheme is not favored not because people cannot differentiate colors (studies show that the human being is capable of distinguishing up to 9 colors without any trouble), but because they cannot easily associate the color to a value, even in the prediction-by-rainbow menu (#46) or in the heatmap menu (#41). These color coding schemes seem to be more appropriate for visualization (*e.g.,* to show how frequent items are globally) than for selecting items.

- After the split menu with replication (#10), the cloud menu (#30) is the first GAM instance with a separate prediction window, as opposed to a close one, as in split with replication (#10), which is related to the observation that people prefer a prediction close to their focus of interest [508].

- Menus with unusual shapes are ranked low, such as the square menu (#37), the tree menu (#44), the hyperbolic menu (#47), adn the flower menu (#48). This suggests that shape-changing menus as a first attempt to convey adaptivity is not very much appreciated at least visually. In the experiment, participants only saw the behavior of the menu, but did not navigate through them, thus limiting their perception of potential advantages, such as easy browsing on a small screen.

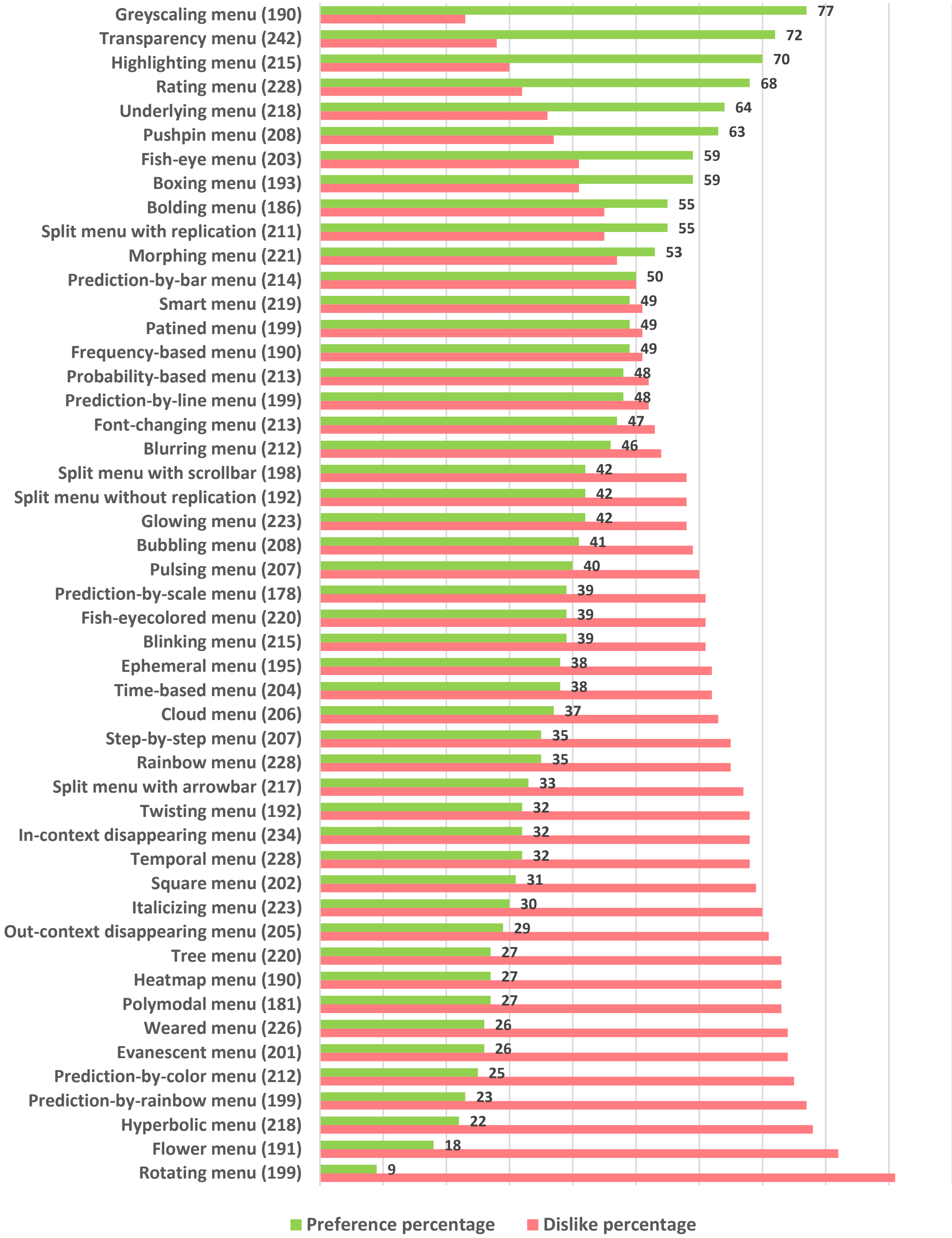


Figure 5.11: Distribution of Graphical Adaptive Menus in decreasing order of their preference percentage. Note that the number of presentations is in parentheses.

Chapter 6

Final Conclusion

In this thesis, we did a SLR in Chapter 2 to assess existing adaptive menus selection techniques and motivate the creation of our two innovative menu selection techniques. In chapter 3, we presented a first multi-platform innovative menu selection technique we created which we called fractal menus, we motivated its design, described its implemented and evaluated some of its characteristics. In a local conclusion, we described its advantages, limitation and possible improvements. In chapter 4, we presented a second innovative menu selection technique we created with its design motivated, its implementation and evaluation described. In chapter 5, we presented a conjoint analysis of existing graphical adaptive menus and the results of an online study that was performed.

6.1 Benefits and Shortcomings

In this section, we recall the different benefits and shortcoming of our SLR, the innovative menu selection techniques we created and conjoint analysis we performed for this thesis.

SLR

The systematic literature review 2 helped us gather information on existing menu interaction techniques and what is being investigated or ameliorated and helped us motivate the creation of our innovative menu selection techniques.

Its main shortcomings was that, we did not perform a full classification of the collected papers. But all the material has being collected and is available in our online collection for such a classification which can be the basis for a further work.

Fractal Menu

The Fractal menu design we implemented proposed a new multi-platform menu interaction technique adaptable for varying screen resolution and which is user adaptable. Its design not only facilitates its deployment on varying resolution but its effectiveness and efficiency can equally be optimize through configurable aspects such as the menu's breath and depth.

For its shortcomings, we did not conduct a usability study with an experiment controlling various independent variables like shape, layout, depth, breath for varying screen resolution and platforms. We did not develop other fractal shapes that could have being interesting for fractal menus. We did not perform a complete efficiency experiment of the fractal menus to evaluate selection times and compare it to those of others menus.

Bilinear Menus

For the bilinear menus, the main advantages the bilinear menu propose was a new menu viewing technique that is adaptable for varying screen resolutions and which can be tweaked to suit the users likes and needs. The bilinear menu enhance the visibility of the menu contents and narrow the possibility for clicking on undesired item by mistake.

For its shortcomings :

- We did not conduct any usability study with an experiment controlling various independent variables like shape, layout.
- The bilinear menu have the screen splitting issue which its not controllable on all operation systems.
- The image quality is not superb because we favored performance over quality.

Conjoint Analysis

Our conjoint analysis helped us evaluate users preference with respect to various adaptative menu designs. One if its limitation is that, due to time constraint we did not include our two developed menus in the study.

6.2 Future work

For the SLR, a full classification of our collection should be done. For our fractal and bilinear menus, a usability study should be done to evaluate the menus effectiveness and efficiency on varying screen resolutions and plate-forms. They study should equally compare their effectiveness and efficiency to those of other menus forms to highlight their advantages and disadvantages. For the menus efficiency evaluation, the study should lay emphasis on evaluation item selection times especially for the fractal menus and these times should be compared with those of other menus.

For the fractal menu in particular, additional techniques should be implemented to ameliorate the selection times of item : *e.g.,* rearranging the menu items in terms of their frequency of use as is the case in frequency base menus. This rearrangement should be such than most frequently used items shouldn't be deeper in the menu structure. Also, additional fractal menus forms should be implemented and evaluated.

For the bilinear menu a new distortions techniques could be implement relatively easily since the interpolation methods have already being implemented. Improving the performance and the rendering of the results can also be the basis for a future work.

Another conjoint analysis including our developed method should be performed.

6.3 Conclusion

The fractal and bilinear menus developed in this thesis are original multi-platform innovative menu interaction techniques. The menu handling for both differs, with the fractal menu delivering a new layout of menu items to favour varying screen resolution and proposes new interaction technique for selection items and the biliner menu delivering new viewing techniques for the menu without changing the original menu layout but only the rendering is changed. Both techniques gives a great foundation for future improvements and further developments.

The source code for both implementations is available on the following GitHub repository : https://github.com/gillesngongang/thesis-UCL.

List of Figures

List of Tables

Bibliography

[1]

[2] Constraint and Compound Widgets. In Theo Pavlidis, editor, *Fundamentals of X Programming: Graphical User Interfaces and Beyond*, Series in Computer Science, pages 173–196. Springer US, Boston, MA, 2002.

[3] Adding Menus and Toolbars. In Scott Mattocks, editor, *Pro PHP-GTK*, pages 231–255. Apress, Berkeley, CA, 2006.

[4] Basic Widgets. In Andrew Krause, editor, *Foundations of GTK+ Development*, pages 75–110. Apress, Berkeley, CA, 2007.

[5] Building Site Navigation with the Spry Menu Bar. In David Powers, editor, *The Essential Guide to Dreamweaver CS3 with CSS, Ajax, and PHP*, pages 183–206. Apress, Berkeley, CA, 2007.

[6] The Menu System. In John K. VanDyk and Matt Westgate, editors, *Pro Drupal Development*, pages 31–48. Apress, Berkeley, CA, 2007.

[7] *47th Hawaii International Conference on System Sciences, HICSS 2014, Waikoloa, HI, USA, January 6-9, 2014*. IEEE Computer Society, 2014.

[8] Hausdorff dimension. https://en.wikipedia.org/wiki/Hausdorff_dimension, 2020. [Online: accessed 20-Dec-2019].

[9] Hausdorff dimension. https://en.wikipedia.org/wiki/Mandelbrot_set, 2020. [Online: accessed 20-Dec-2019].

[10] Hausdorff dimension. https://en.wikipedia.org/wiki/Fractal_dimension, 2020. [Online: accessed 20-Dec-2019].

[11] Lebesgue covering dimension. https://en.wikipedia.org/wiki/Lebesgue_covering_dimension, 2020. [Online: accessed 11-Jan-2020].

[12] Hybesis H.urna. Generating fractals using lindenmayer systems (https://medium.com/analytics-vidhya/generating-fractals-using-lindenmayer-systems-6214dddbe223). https://medium.com/@urna.hybesis/generating-fractals-using-lindenmayer-systems-6214dddbe223, 2018. [Online: accessed 18-Dec-2019].

[13] Ahmad Abdel-Hafez, Yue Xu, and Nan Tian. Item Reputation-Aware Recommender Systems. In *Proceedings of the 16th International Conference on Information Integration and Web-Based Applications & Services*, iiWAS '14, pages 79–86, New York,

NY, USA, 2014. Association for Computing Machinery. event-place: Hanoi, Viet Nam.

[14] J. Aceituno, S. Malacria, P. Quinn, N. Roussel, A. Cockburn, and G. Casiez. The design, use, and performance of edge-scrolling techniques. *International Journal of Human-Computer Studies*, 97:58 – 76, 2017.

[15] Takamasa Adachi, Seiya Koura, Fumihisa Shibata, and Asako Kimura. Forearm Menu: Using Forearm As Menu Widget on Tabletop System. In *Proceedings of the 2013 ACM International Conference on Interactive Tabletops and Surfaces*, ITS '13, pages 333–336, New York, NY, USA, 2013. ACM. event-place: St. Andrews, Scotland, United Kingdom.

[16] Anand Agarawala and Ravin Balakrishnan. Keepin' It Real: Pushing the Desktop Metaphor with Physics, Piles and the Pen. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '06, pages 1283–1292, New York, NY, USA, 2006. ACM. event-place: Montréal, Québec, Canada.

[17] B Agarwal and W Stuerzlinger. Widgetlens: A system for adaptive content magnification of widgets.

[18] Christopher Ahlberg and Ben Shneiderman. The Alphaslider: A Compact and Rapid Selector. In *Conference Companion on Human Factors in Computing Systems*, CHI '94, pages 226–, New York, NY, USA, 1994. ACM. event-place: Boston, Massachusetts, USA.

[19] David Ahlstroem, Rainer Alexandrowicz, and Martin Hitz. Improving Menu Interaction: A Comparison of Standard, Force Enhanced and Jumping Menus. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '06, pages 1067–1076, New York, NY, USA, 2006. ACM. event-place: Montréal, Québec, Canada.

[20] David Ahlström, Andy Cockburn, Carl Gutwin, and Pourang Irani. Why It's Quick to Be Square: Modelling New and Existing Hierarchical Menu Designs. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pages 1371–1380, New York, NY, USA, 2010. ACM. event-place: Atlanta, Georgia, USA.

[21] K. Al-Omar and D. Rigas. Comparison of Adaptive, Adaptable and Mixed-Initiative Menus. In *2009 International Conference on CyberWorlds*, pages 292–297, September 2009.

[22] K. Al-Omar and D. Rigas. A user performance evaluation of personalised menus. In *2009 Second International Conference on the Applications of Digital Information and Web Technologies*, pages 104–109, August 2009.

[23] Khalid Al-Omar and Dimitrios Rigas. The effect of size of personalised menus on user satisfaction. In *Proceedings of the 11th WSEAS International Conference on Mathematical Methods and Computational Techniques in Electrical Engineering*, MMACTEE'09, pages 322–327, Stevens Point, Wisconsin, USA, 2009. World Scientific and Engineering Academy and Society (WSEAS).

[24] Anna Alfieri. L-system fractals: an educational approach by new technologies. 07 2015.

[25] Donald Wayne Allison, Steven Carl Williamson, and Walter Benjamin Herrington. *Interactive program guide navigator menu system*. Google Patents, July 2001.

[26] Nourah A. ALRossais. Integrating Item Based Stereotypes in Recommender Systems. In *Proceedings of the 26th Conference on User Modeling, Adaptation and Personalization*, UMAP '18, pages 265–268, New York, NY, USA, 2018. Association for Computing Machinery. event-place: Singapore, Singapore.

[27] Tao D Alter. 3d pose from 3 corresponding points under weak-perspective projection. Technical report, MASSACHUSETTS INST OF TECH CAMBRIDGE ARTIFICIAL INTELLIGENCE LAB, 1992.

[28] Humberto Lidio Antonelli, Rodrigo Augusto Igawa, Renata Pontin De Mattos Fortes, Eduardo Henrique Rizo, and Willian Massami Watanabe. Drop-Down Menu Widget Identification Using HTML Structure Changes Classification. *ACM Trans. Access. Comput.*, 11(2):10:1–10:23, June 2018.

[29] Humberto Lidio Antonelli, Elias Adriano N. da Silva, and Renata Pontin M. Fortes. A Model-driven Development for Creating Accessible Web Menus. *Procedia Computer Science*, 67:95 – 104, 2015.

[30] Humberto Lidio Antonelli, Willian Massami Watanabe, and Renata Pontin de Mattos Fortes. Adapting Web Menus to Mobile Devices for Elderly Interactions. In *Proceedings of the 8th International Conference on Software Development and Technologies for Enhancing Accessibility and Fighting Info-Exclusion*, DSAI 2018, pages 47–54, New York, NY, USA, 2018. Association for Computing Machinery. event-place: Thessaloniki, Greece.

[31] Liat Antwarg, Talia Lavie, Lior Rokach, Bracha Shapira, and Joachim Meyer. Highlighting items as means of adaptive assistance. *Behaviour & Information Technology*, 32(8):761–777, 2013.

[32] M. D. Apperley, I. Tzavaras, and R. Spence. A BIFOCAL DISPLAY TECHNIQUE FOR DATA PRESENTATION. In D.S. Greenaway and E.A. Warman, editors, *Eurographics Conference Proceedings*. The Eurographics Association, 1982.

[33] Caroline Appert, Olivier Chapuis, and Emmanuel Pietriga. High-precision magnification lenses. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 273–282, 2010.

[34] Udo Arend, Klaus-Peter Muthig, and Jens Wandmacher. Evidence for global feature superiority in menu selection by icons. *Behaviour & Information Technology*, 6(4):411–426, 1987.

[35] Leena Arhippainen, Tapani Rantakokko, and Marika Tähti. Navigation with an adaptive mobile map-application: User experiences of gesture- and context-sensitiveness. In Hitomi Murakami, Hideyuki Nakashima, Hideyuki Tokuda, and Michiaki Yasumura, editors, *Ubiquitous Computing Systems*, pages 62–73. Springer, Berlin, Heidelberg, 2005.

[36] Alexandre Armengol-Urpi and Sanjay E. Sarma. Sublime: a hands-free virtual reality menu navigation system using a high-frequency SSVEP-based brain-computer interface. In *VRST*, pages 1:1–1:8. ACM, 2018.

[37] L. M. Arnold. Item Selection from Menus: The Influence of Menu Organization, Query Interpretation, and Programming Experience on Selection Strategies. *SIGCHI Bull.*, 21(1):81–85, August 1989.

[38] James D. Arthur. A descriptive/prescriptive model for menu-based interaction. *International Journal of Man-Machine Studies*, 25(1):19 – 32, 1986.

[39] James D. Arthur. Toward a formal specification of menu-based systems. *Journal of Systems and Software*, 7(1):73 – 82, 1987.

[40] W. David Ashley and Andrew Krause. Dynamic User Interfaces. In W. David Ashley and Andrew Krause, editors, *Foundations of PyGTK Development: GUI Creation with Python*, pages 317–344. Apress, Berkeley, CA, 2019.

[41] Stacey Ashlund, Kevin Mullet, Austin Henderson, Erik Hollnagel, and Ted N. White, editors. *Human-Computer Interaction, INTERACT '93, IFIP TC13 International Conference on Human-Computer Interaction, 24-29 April 1993, Amsterdam, The Netherlands, jointly organised with ACM Conference on Human Aspects in Computing Systems CHI'93, Adjunct Proceedings*. ACM, 1993.

[42] Saïd Assar, Oscar Pastor, and Haralambos Mouratidis, editors. *11th International Conference on Research Challenges in Information Science, RCIS 2017, Brighton, United Kingdom, May 10-12, 2017*. IEEE, 2017.

[43] Siddhartha Asthana, Pushpendra Singh, and Amarjeet Singh. Exploring Adverse Effects of Adaptive Voice Menu. In *CHI '13 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '13, pages 775–780, New York, NY, USA, 2013. Association for Computing Machinery. event-place: Paris, France.

[44] Takumi Azai, Shuhei Ogawa, Mai Otsuki, Fumihisa Shibata, and Asako Kimura. Selection and Manipulation Methods for a Menu Widget on the Human Forearm. In *Proceedings of the 2017 CHI Conference Extended Abstracts on Human Factors in Computing Systems*, CHI EA '17, pages 357–360, New York, NY, USA, 2017. Association for Computing Machinery. event-place: Denver, Colorado, USA.

[45] Takumi Azai, Mai Otsuki, Fumihisa Shibata, and Asako Kimura. Open Palm Menu: A Virtual Menu Placed in Front of the Palm. In *Proceedings of the 9th Augmented Human International Conference*, AH '18, pages 17:1–17:5, New York, NY, USA, 2018. ACM. event-place: Seoul, Republic of Korea.

[46] Takumi Azai, Syunsuke Ushiro, Junlin Li, Mai Otsuki, Fumihisa Shibata, and Asako Kimura. Tap-Tap Menu: Body Touching for Virtual Interactive Menus. In *Proceedings of the 24th ACM Symposium on Virtual Reality Software and Technology*, VRST '18, New York, NY, USA, 2018. Association for Computing Machinery. event-place: Tokyo, Japan.

[47] Felipe Bacim, Regis Kopper, and Doug A. Bowman. Design and evaluation of 3D selection techniques based on progressive refinement. *International Journal of Human-Computer Studies*, 71(7):785 – 802, 2013.

[48] Huidong Bai, Lei Gao, Jihad El-Sana, and Mark Billinghurst. Free-Hand Interaction for Handheld Augmented Reality Using an RGB-Depth Camera. In *SIGGRAPH Asia 2013 Symposium on Mobile Graphics and Interactive Applications*, SA '13, New York, NY, USA, 2013. Association for Computing Machinery. event-place: Hong Kong, Hong Kong.

[49] Gilles Bailly. Techniques de Menus: Description, Développement, Evaluation. In *Proceedings of the 19th Conference on l'Interaction Homme-Machine*, IHM '07, pages 217–220, New York, NY, USA, 2007. Association for Computing Machinery. event-place: Paris, France.

[50] Gilles Bailly, Alexandre Demeure, Eric Lecolinet, and Laurence Nigay. MultiTouch Menu (MTM). In *Proceedings of the 20th Conference on l'Interaction Homme-Machine*, IHM '08, pages 165–168, New York, NY, USA, 2008. Association for Computing Machinery. event-place: Metz, France.

[51] Gilles Bailly, Eric Lecolinet, and Laurence Nigay. Wave Menus: Improving the Novice Mode of Hierarchical Marking Menus. In Cécilia Baranauskas, Philippe Palanque, Julio Abascal, and Simone Diniz Junqueira Barbosa, editors, *Human-Computer Interaction – INTERACT 2007*, Lecture Notes in Computer Science, pages 475–488, Berlin, Heidelberg, 2007. Springer.

[52] Gilles Bailly, Eric Lecolinet, and Laurence Nigay. Flower Menus: A New Type of Marking Menu with Large Menu Breadth, Within Groups and Efficient Expert Mode Memorization. In *Proceedings of the Working Conference on Advanced Visual Interfaces*, AVI '08, pages 15–22, New York, NY, USA, 2008. ACM. event-place: Napoli, Italy.

[53] Gilles Bailly, Eric Lecolinet, and Laurence Nigay. Visual Menu Techniques. *ACM Comput. Surv.*, 49(4), December 2016.

[54] Gilles Bailly, Jörg Müller, and Eric Lecolinet. Design and evaluation of finger-count interaction: Combining multitouch gestures and menus. *International Journal of Human-Computer Studies*, 70(10):673 – 689, 2012.

[55] Gilles Bailly and Antti Oulasvirta. Toward Optimal Menu Design. *Interactions*, 21(4):40–45, July 2014.

[56] Gilles Bailly, Antti Oulasvirta, Timo Kötzing, and Sabrina Hoppe. Menuoptimizer: Interactive optimization of menu systems. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology*, UIST '13, pages 331–342, New York, NY, USA, 2013. ACM.

[57] Gilles Bailly, Antti Oulasvirta, Timo Kötzing, and Sabrina Hoppe. MenuOptimizer: interactive optimization of menu systems. In *UIST*, pages 331–342. ACM, 2013.

[58] Gilles Bailly, Robert Walter, Jörg Müller, Tongyan Ning, and Eric Lecolinet. Comparing Free Hand Menu Techniques for Distant Displays Using Linear, Marking and Finger-Count Menus. In *INTERACT (2)*, volume 6947 of *Lecture Notes in Computer Science*, pages 248–262. Springer, 2011.

[59] Nikola Banovic, Frank Chun Yat Li, David Dearman, Koji Yatani, and Khai N. Truong. Design of Unimanual Multi-finger Pie Menu Interaction. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*, ITS '11, pages 120–129, New York, NY, USA, 2011. ACM. event-place: Kobe, Japan.

[60] W. Barfield, C. Rosenberg, and G. Levasseur. The use of icons, earcons, and commands in the design of an online hierarchical menu. *IEEE Transactions on Professional Communication*, 34(2):101–108, June 1991.

[61] Surajit Das Barman, Mahamudul Hasan, and Falguni Roy. A Genre-Based Item-Item Collaborative Filtering: Facing the Cold-Start Problem. In *Proceedings of the 2019 8th International Conference on Software and Computer Applications*, ICSCA '19, pages 258–262, New York, NY, USA, 2019. Association for Computing Machinery. event-place: Penang, Malaysia.

[62] Amy Battles and Robb Cazier. *Camera menu system*. Google Patents, November 2004.

[63] Patrick Baudisch, Desney Tan, Maxime Collomb, Dan Robbins, Ken Hinckley, Ken Hinckley, Maneesh Agrawala, Shengdong Zhao, and Gonzalo Ramos. Phosphor: Explaining transitions in the user interface using afterglow effects. In *Proceedings of the 19th Annual ACM Symposium on User Interface Software and Technology*, UIST '06, pages 169–178, New York, NY, USA, 2006. ACM.

[64] Jens Bauer and Achim Ebert. Mobile Devices for Virtual Reality Interaction. A Survey of Techniques and Metaphors. In Guido Brunnett, Sabine Coquillart, Robert van Liere, Gregory Welch, and Libor Váša, editors, *Virtual Realities: International Dagstuhl Seminar, Dagstuhl Castle, Germany, June 9-14, 2013, Revised Selected Papers*, Lecture Notes in Computer Science, pages 91–107. Springer International Publishing, Cham, 2015.

[65] Jens Bauer, Achim Ebert, Oliver Kreylos, and Bernd Hamann. Marking Menus for Eyes-Free Interaction Using Smart Phones and Tablets. In *CD-ARES*, volume 8127 of *Lecture Notes in Computer Science*, pages 481–494. Springer, 2013.

[66] P. Bayman, S. Civanlar, and W. B. Whitten. Usage-sensitive menu design with Huffman coding. In *[1989] Proceedings of the Twenty-Second Annual Hawaii International Conference on System Sciences. Volume II: Software Track*, volume 2, pages 436–437 vol.2, January 1989.

[67] Michel Beaudouin-Lafon. Designing interaction, not interfaces. In *Proceedings of the Working Conference on Advanced Visual Interfaces*, AVI '04, pages 15–22, New York, NY, USA, 2004. ACM.

[68] Jongmin Beck, Sung Ho Han, and Jungchul Park. Presenting a Submenu Window for Menu Search on a Cellular Phone. *Int. J. Hum. Comput. Interaction*, 20(3):233–245, 2006.

[69] Benjamin B. Bederson. Fisheye Menus. In *Proceedings of the 13th Annual ACM Symposium on User Interface Software and Technology*, UIST '00, pages 217–225, New York, NY, USA, 2000. Association for Computing Machinery. event-place: San Diego, California, USA.

[70] Benjamin B. Bederson. Fisheye Menus. In BENJAMIN B. BEDERSON and BEN SHNEIDERMAN, editors, *The Craft of Information Visualization*, Interactive Technologies, pages 299 – 307. Morgan Kaufmann, San Francisco, 2003.

[71] Mohammed Belatar and François Coldefy. Sketched Menu: A Tabletop-Menu Technique for GUI Object Creation. In *Proceedings of the 2nd ACM SIGCHI Symposium on Engineering Interactive Computing Systems*, EICS '10, pages 77–86, New York, NY, USA, 2010. Association for Computing Machinery. event-place: Berlin, Germany.

[72] Mohammed Belatar and François Coldefy. Sketched menus and iconic gestures, techniques designed in the context of shareable interfaces. In *ITS*, pages 143–146. ACM, 2010.

[73] BenMeadowcroft. Pie menus a brief examination (http://www.benmeadowcroft.com/reports/piemenu/). http://www.benmeadowcroft.com/reports/piemenu/. [Online: accessed 18-Mai-2020].

[74] Michael Bernard and Chris Hamblin. Cascading versus indexed menu design. *Usability News*, 5(1), 2003.

[75] C. Bernius. The ATLAS trigger menu: Design, performance and monitoring. In *2012 IEEE Nuclear Science Symposium and Medical Imaging Conference Record (NSS/MIC)*, pages 1382–1385, October 2012.

[76] Jacques Bertin. *Sémiologie graphique*. Mouton/Gauthier-Villars, Paris, France, 1967.

[77] Jacques Bertin. *Graphics and graphic information processing*. Walter de Gruyter, 2011.

[78] Matthias Bezold and Wolfgan Minker. *Adaptive Multimodal Interactive Systems*. Springer, Berlin, Germany, 2011.

[79] Matthias Bezold and Wolfgang Minker. A framework for adapting interactive systems to user behavior. *J. Ambient Intell. Smart Environ.*, 2(4):369–387, December 2010.

[80] Pradipta Biswas and Pat Langdon. A New Interaction Technique Involving Eye Gaze Tracker and Scanning System. In *Proceedings of the 2013 Conference on Eye Tracking South Africa*, ETSA '13, pages 67–70, New York, NY, USA, 2013. Association for Computing Machinery. event-place: Cape Town, South Africa.

[81] G. Blasko and S. Feiner. A menu interface for wearable computing. In *Proceedings. Sixth International Symposium on Wearable Computers,*, pages 164–165, October 2002.

[82] G. Blasko and S. Feiner. An interaction system for watch computers using tactile guidance and bidirectional segmented strokes. In *Eighth International Symposium on Wearable Computers*, volume 1, pages 120–123, October 2004.

[83] François Bodart, Anne-Marie Hennebert, Jean-Marie Leheureux, Isabelle Provot, Jean Vanderdonckt, and Giovanni Zucchinetti. *Key Activities for a Development Methodology of Interactive Applications*, pages 109–134. Springer London, London, 1996.

[84] David Bonnet and Caroline Appert. SAM: The Swiss Army Menu. In *Proceedings of the 23rd Conference on l'Interaction Homme-Machine*, IHM '11, New York, NY, USA, 2011. Association for Computing Machinery. event-place: Sophia Antipolis, France.

[85] Fredrik Boström, Petteri Nurmi, Patrik Floréen, Tianyan Liu, Tiina-Kaisa Oikarinen, Akos Vetek, and Péter Boda. Capricorn - an Intelligent User Interface for Mobile Widgets. In *Proceedings of the 10th International Conference on Human Computer Interaction with Mobile Devices and Services*, MobileHCI '08, pages 327–330, New York, NY, USA, 2008. ACM. event-place: Amsterdam, The Netherlands.

[86] A. Bouraoui. Component based development of non-visual applications using braille-speech widgets. In *2007 IEEE/ACS International Conference on Computer Systems and Applications*, pages 84–91, May 2007.

[87] Sara Bouzit, Gaëlle Calvary, Denis Chêne, and Jean Vanderdonckt. Automated evaluation of menu by guidelines review. In Iftene and Vanderdonckt [239], pages 11–21.

[88] Sara Bouzit, Gaëlle Calvary, Denis Chêne, and Jean Vanderdonckt. A design space for engineering graphical adaptive menus. In *Proceedings of the 8th ACM SIGCHI Symposium on Engineering Interactive Computing Systems*, EICS '16, pages 239–244, New York, NY, USA, 2016. ACM.

[89] Sara Bouzit, Gaëlle Calvary, Denis Chêne, and Jean Vanderdonckt. Interface adaptivity by widget promotion/demotion. In José Ignacio Panach, Jean Vanderdonckt, and Oscar Pastor, editors, *Proceedings of the ACM SIGCHI Symposium on Engineering Interactive Computing Systems, EICS 2019, Valencia, Spain, June 18-21, 2019*, pages 18:1–18:6. ACM, 2019.

[90] Sara Bouzit, Gaëlle Calvary, Joëlle Coutaz, Denis Chêne, Éric Petit, and Jean Vanderdonckt. The PDA-LPA design space for user interface adaptation. In Assar et al. [42], pages 353–364.

[91] Sara Bouzit, Gaëlle Calvary, Denis Chundefinedne, and Jean Vanderdonckt. Polymodal Menus: A Model-Based Approach for Designing Multimodal Adaptive Menus for Small Screens. *Proc. ACM Hum.-Comput. Interact.*, 1(EICS), June 2017.

[92] Sara Bouzit, Gaëlle Calvary, Denis Chundefinedne, and Jean Vanderdonckt. Interface Adaptivity by Widget Promotion/Demotion. In *Proceedings of the ACM SIGCHI Symposium on Engineering Interactive Computing Systems*, EICS '19, New York, NY, USA, 2019. Association for Computing Machinery. event-place: Valencia, Spain.

[93] Sara Bouzit, Gaëlle Calvary, Denis Chêne, and Jean Vanderdonckt. A comparison of shortcut and step-by-step adaptive menus for smartphones. In *Proceedings of the 30th International BCS Human Computer Interaction Conference: Fusion!*, HCI '16, pages 1–12, Poole, United Kingdom, July 2016. BCS Learning & Development Ltd.

[94] Sara Bouzit, Denis Chêne, and Gaelle Calvary. From appearing to disappearing ephemeral adaptation for small screens. In *Proceedings of the 26th Australian Computer-Human Interaction Conference on Designing Futures: the Future of Design*,

OzCHI '14, pages 41–48, Sydney, New South Wales, Australia, dec 2014. Association for Computing Machinery.

[95] Sara Bouzit, Denis Chêne, and Gaëlle Calvary. Evanescent Adaptation on Small Screens. In *Proceedings of the Annual Meeting of the Australian Special Interest Group for Computer Human Interaction*, OzCHI '15, pages 62–68, Parkville, VIC, Australia, December 2015. Association for Computing Machinery.

[96] D. A. Bowman and C. A. Wingrave. Design and evaluation of menu systems for immersive virtual environments. In *Proceedings IEEE Virtual Reality 2001*, pages 149–156, March 2001.

[97] Ralph Allan Bradley and Milton E. Terry. Rank analysis of incomplete block designs: I. the method of paired comparisons. *Biometrika*, 39(3/4):324–345, 1952.

[98] Russell J. Branaghan, Christine M. Covas-Smith, Kenneth D. Jackson, and Craig Eidman. Using knowledge structures to redesign an instructor–operator station. *Applied Ergonomics*, 42(6):934 – 940, 2011.

[99] Peter Brandl, Jakob Leitner, Thomas Seifried, Michael Haller, Bernard Doray, and Paul To. Occlusion-Aware Menu Design for Digital Tabletops. In *CHI '09 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '09, pages 3223–3228, New York, NY, USA, 2009. Association for Computing Machinery. event-place: Boston, MA, USA.

[100] Nicolas Bremard, Laurent Grisoni, and Bruno De Araujo. Interaction Events in Contactless Gestural Systems: From Motion to Interaction. In *Proceedings of the 2014 International Workshop on Movement and Computing*, MOCO '14, pages 166–169, New York, NY, USA, 2014. Association for Computing Machinery. event-place: Paris, France.

[101] Stephen Brewster, Joanna Lumsden, Marek Bell, Malcolm Hall, and Stuart Tasker. Multimodal "eyes-Free" Interaction Techniques for Wearable Devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '03, pages 473–480, New York, NY, USA, 2003. Association for Computing Machinery. event-place: Ft. Lauderdale, Florida, USA.

[102] Robert Bridle and Eric McCreath. Predictive menu selection on a mobile phone. In *Proceedings Workshop W7 on mining spatio-temporal data*, ECML/PKDD'2005, pages 327–329, 2005.

[103] Robert Bridle and Eric McCreath. Inducing shortcuts on a mobile phone interface. In *Proceedings of the 11th International Conference on Intelligent User Interfaces*, IUI '06, pages 327–329, New York, NY, USA, 2006. ACM.

[104] James W. Brown. Controlling the Complexity of Menu Networks. *Commun. ACM*, 25(7):412–418, July 1982.

[105] Nicolas Burny, Suzanne Kieffer, Nathan Magrofuoco, Jorge Luis Perez Medina, Paolo Roselli, and Jean Vanderdonckt. Feedup, feedback, and feedforward in curve mid-air 3d gestures. In M. Giordano, O. Georgiou, B. Dzidek, L. Corenthy, Jin Ryong Kim, S. Subramanian, and Stephen Brewster, editors, *Proceedings of CHI '18 Workshop on Mid-Air Haptics for Control Interfaces.* 2018.

[106] Gaëlle Calvary, Joëlle Coutaz, David Thevenin, Quentin Limbourg, Laurent Bouillon, and Jean Vanderdonckt. A unifying reference framework for multi-target user interfaces. *Interacting with Computers*, 15(3):289–308, 2003.

[107] Kip Canfield. Priming intelligent split menus with text corpora for computerized patient record data-entry. *International Journal of Bio-Medical Computing*, 39(2):263 – 273, 1995.

[108] Xiang Cao and Ravin Balakrishnan. VisionWand: Interaction Techniques for Large Displays Using a Passive Wand Tracked in 3D. In *Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology*, UIST '03, pages 173–182, New York, NY, USA, 2003. ACM. event-place: Vancouver, Canada.

[109] Antonio Capobianco and Caroline Essert. Study of Performances of "Haptic Walls" Modalities for a 3D Menu. In Astrid M. L. Kappers, Jan B. F. van Erp, Wouter M. Bergmann Tiest, and Frans C. T. van der Helm, editors, *Haptics: Generating and Perceiving Tangible Sensations*, Lecture Notes in Computer Science, pages 152–159, Berlin, Heidelberg, 2010. Springer.

[110] C. Carey-Smith, D. Powley, and K. Carey-Smith. An adaptable health screening questionnaire. In *Proceedings 1993 The First New Zealand International Two-Stream Conference on Artificial Neural Networks and Expert Systems*, pages 259–260, November 1993.

[111] M. S. T. Carpendale. *Considering Visual Variables as a Basis for Information Visualisation*. Science Research Publications. University of Calgary, Faculty of science, Calgary, Canada, 2001.

[112] M Sheelagh T Carpendale and Catherine Montagnese. A framework for unifying presentation space. In *Proceedings of the 14th annual ACM symposium on User interface software and technology*, pages 61–70, 2001.

[113] Giandomenico Caruso, Elia Gatti, and Monica Bordegoni. Study on the Usability of a Haptic Menu for 3D Interaction. In *INTERACT (2)*, volume 6947 of *Lecture Notes in Computer Science*, pages 186–193. Springer, 2011.

[114] Leonardo Cella, Stefano Cereda, Massimo Quadrana, and Paolo Cremonesi. Deriving Item Features Relevance from Past User Interactions. In *Proceedings of the 25th Conference on User Modeling, Adaptation and Personalization*, UMAP '17, pages 275–279, New York, NY, USA, 2017. Association for Computing Machinery. event-place: Bratislava, Slovakia.

[115] Jonathan Chaboissier and Frédéric Vernier. CentralMenu: Menu PartagÉ Sur Table Interactive. In *Proceedings of the 22Nd Conference on L'Interaction Homme-Machine*, IHM '10, pages 213–216, New York, NY, USA, 2010. ACM. event-place: Luxembourg, Luxembourg.

[116] Joyce Chai, Jimmy Lin, Wlodek Zadrozny, Yiming Ye, Margo Budzikowska, Veronika Horvath, Nanda Kambhatla, and Catherine Wolf. Comparative evaluation of a natural language dialog based system and a menu driven system for information access: a case study. In *Content-Based Multimedia Information Access-Volume*

2, pages 1590–1600. LE CENTRE DE HAUTES ETUDES INTERNATIONALES D'INFORMATIQUE DOCUMENTAIRE, 2000.

[117] Jaime Chapinal Cervantes, Francisco Luis Gutiérrez Vela, and Patricia Paderewski Rodrundefinedguez. Natural Interaction Techniques Using Kinect. In *Proceedings of the 13th International Conference on Interacción Persona-Ordenador*, INTERACCION '12, New York, NY, USA, 2012. Association for Computing Machinery. event-place: Elche, Spain.

[118] Debaleena Chattopadhyay and Davide Bolchini. Touchless Circular Menus: Toward an Intuitive UI for Touchless Interactions with Large Displays. In *Proceedings of the 2014 International Working Conference on Advanced Visual Interfaces*, AVI '14, pages 33–40, New York, NY, USA, 2014. Association for Computing Machinery. event-place: Como, Italy.

[119] Ankit chaudhary. Experience in Item Based Recommender System. In *Proceedings of the 30th Annual ACM Symposium on Applied Computing*, SAC '15, pages 1112–1114, New York, NY, USA, 2015. Association for Computing Machinery. event-place: Salamanca, Spain.

[120] Xiang "Anthony" Chen. Body-Centric Interaction with Mobile Devices. In *Proceedings of the Sixth International Conference on Tangible, Embedded and Embodied Interaction*, TEI '12, pages 385–386, New York, NY, USA, 2012. Association for Computing Machinery. event-place: Kingston, Ontario, Canada.

[121] Xiang "Anthony" Chen, Julia Schwarz, Chris Harrison, Jennifer Mankoff, and Scott Hudson. Around-Body Interaction: Sensing & Interaction Techniques for Proprioception-Enhanced Input with Mobile Devices. In *Proceedings of the 16th International Conference on Human-Computer Interaction with Mobile Devices & Services*, MobileHCI '14, pages 287–290, New York, NY, USA, 2014. Association for Computing Machinery. event-place: Toronto, ON, Canada.

[122] Kelvin Cheng and Masahiro Takatsuka. Initial Evaluation of a Bare-Hand Interaction Technique for Large Displays Using a Webcam. In *Proceedings of the 1st ACM SIGCHI Symposium on Engineering Interactive Computing Systems*, EICS '09, pages 291–296, New York, NY, USA, 2009. Association for Computing Machinery. event-place: Pittsburgh, PA, USA.

[123] D. B. Chertoff, R. Byers, and J. J. LaViola. Poster: Evaluation of menu techniques using a 3D game input device. In *2009 IEEE Symposium on 3D User Interfaces*, pages 139–140, March 2009.

[124] Dustin B. Chertoff, Ross W. Byers, and Joseph J. LaViola. An Exploration of Menu Techniques Using a 3D Game Input Device. In *Proceedings of the 4th International Conference on Foundations of Digital Games*, FDG '09, pages 256–262, New York, NY, USA, 2009. Association for Computing Machinery. event-place: Orlando, Florida.

[125] Chiung-Hui Chiu, Chiao-Hua Chuang, and Hsieh-Fen Hsiao. An interface design for a structured computer-mediated communication tool for elementary school students: pull-down versus explicit menus. *BJET*, 37(2):303–306, 2006.

[126] Jacek Chodak. Menu interfacefor people with physical disabilities based on EOG. *Bio-Algorithms and Med-Systems*, 6(12-S):53–54, 2010.

[127] J. Choi, Y. Kim, and G. J. Kim. Usability of one handed interaction methods for hand-held projection-based augmented reality. In *2011 10th IEEE International Symposium on Mixed and Augmented Reality*, pages 233–234, October 2011.

[128] Jinhyuk Choi and Gerard J. Kim. Usability of one-handed interaction methods for handheld projection-based augmented reality. *Personal and Ubiquitous Computing*, 17(2):399–409, February 2013.

[129] Andy Cockburn, Carl Gutwin, and Saul Greenberg. A Predictive Model of Menu Performance. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '07, pages 627–636, New York, NY, USA, 2007. Association for Computing Machinery. event-place: San Jose, California, USA.

[130] K. Coninx, F. Van Reeth, and E. Flerackers. A hybrid 2D/3D user interface for immersive object modeling. In *Proceedings Computer Graphics International*, pages 47–55, June 1997.

[131] Jessica Conradi, Bjoern Nord, and Thomas Alexander. Menu Styles of Mobile Devices and Their Influence on Gaze Behavior While Walking. In Christopher Marc Schlick, Sönke Duckwitz, Frank Flemisch, Martin Frenz, Sinem Kuz, Alexander Mertens, and Susanne Mütze-Niewöhner, editors, *Advances in Ergonomic Design of Systems, Products and Processes*, pages 275–288, Berlin, Heidelberg, 2017. Springer.

[132] N. Cooharojananone, K. Taohai, and S. Phimoltares. A New Design of ATM Interface for Banking Services in Thailand. In *2010 10th IEEE/IPSJ International Symposium on Applications and the Internet*, pages 312–315, July 2010.

[133] Gennaro Costagliola, Mattia De Rosa, and Vittorio Fuccella. A technique for improving text editing on touchscreen devices. *Journal of Visual Languages & Computing*, 47:1 – 8, 2018.

[134] Ph. Courcoux and M. Semenou. Preference data analysis using a paired comparison model. *Food Quality and Preference*, 8(5):353 – 358, 1997. Third Sensometrics Meeting.

[135] Jian Cui and Alexei Sourin. Mid-air interaction with optical tracking for 3D modeling. *Computers & Graphics*, 74:1 – 11, 2018.

[136] James E Cullinan. *Menu system*. Google Patents, March 2007.

[137] Bill Cureton. Devguide — The Open Windows G.U.I. Builder. In Bill Cureton, editor, *Software Engineering on Sun Workstations®*, pages 277–296. Springer US, New York, NY, 1993.

[138] Tom Van Cutsem. Session 4 - ifs fractals. http://soft.vub.ac.be/~tvcutsem/teaching/wpo/grafsys/ex4/les4.html, 2008 - 2009. [Online: accessed 19-Dec-2019].

[139] R. Dachselt and A. Hübner. A Survey and Taxonomy of 3D Menu Techniques. In *Proceedings of the 12th Eurographics Conference on Virtual Environments*, EGVE'06, pages 89–99, Goslar, DEU, 2006. Eurographics Association. event-place: Lisbon, Portugal.

[140] Raimund Dachselt, Mathias Frisch, and Markus Weiland. FacetZoom: A Continuous Multi-Scale Widget for Navigating Hierarchical Metadata. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, pages 1353–1356, New York, NY, USA, 2008. Association for Computing Machinery. event-place: Florence, Italy.

[141] Raimund Dachselt and Anett Hübner. Three-dimensional menus: A survey and taxonomy. *Computers & Graphics*, 31(1):53 – 65, 2007.

[142] Girish Dalvi. Development of an Intuitive User-Centric Font Selection Menu. In Dinesh Katre, Rikke Orngreen, Pradeep Yammiyavar, and Torkil Clemmensen, editors, *Human Work Interaction Design: Usability in Social, Cultural and Organizational Contexts*, IFIP Advances in Information and Communication Technology, pages 144–153, Berlin, Heidelberg, 2010. Springer.

[143] A. van Dam. Beyond WIMP. *IEEE Computer Graphics and Applications*, 20(1):50–51, January 2000.

[144] A. I. Danilenko and M. V. Goubko. Semantic-aware optimization of user interface menus. *Automation and Remote Control*, 74(8):1399–1411, August 2013.

[145] K. Das and C. W. Borst. An evaluation of menu properties and pointing techniques in a projection-based VR environment. In *2010 IEEE Symposium on 3D User Interfaces (3DUI)*, pages 47–50, March 2010.

[146] Kaushik Das and Christoph W. Borst. VR Menus: Investigation of Distance, Size, Auto-scale, and Ray Casting vs. Pointer-Attached-to-Menu. In George Bebis, Richard Boyle, Bahram Parvin, Darko Koracin, Ronald Chung, Riad Hammoud, Muhammad Hussain, Tan Kar-Han, Roger Crawfis, Daniel Thalmann, David Kao, and Lisa Avila, editors, *Advances in Visual Computing*, Lecture Notes in Computer Science, pages 719–728, Berlin, Heidelberg, 2010. Springer.

[147] M. M. Davis, J. L. Gabbard, D. A. Bowman, and D. Gracanin. Depth-based 3D gesture multi-level radial menu for virtual object manipulation. In *2016 IEEE Virtual Reality (VR)*, pages 169–170, March 2016.

[148] Adrien Delaye, Rafik Sekkal, and Eric Anquetil. Continuous Marking Menus for Learning Cursive Pen-based Gestures. In *Proceedings of the 16th International Conference on Intelligent User Interfaces*, IUI '11, pages 319–322, New York, NY, USA, 2011. ACM. event-place: Palo Alto, CA, USA.

[149] Charles-Eric Dessart, Vivian Genaro Motti, and Jean Vanderdonckt. Animated transitions between user interface views. In *Proceedings of the International Working Conference on Advanced Visual Interfaces*, AVI '12, pages 341–348, New York, NY, USA, 2012. ACM.

[150] Fabian Di Fiore, Peter Vandoren, and Frank Van Reeth. Multimodal Interaction in a Collaborative Virtual Brainstorming Environment. In Yuhua Luo, editor, *Cooperative Design, Visualization, and Engineering*, Lecture Notes in Computer Science, pages 47–60, Berlin, Heidelberg, 2004. Springer.

[151] O. Dianita, C. Lin, and T. Wijayanto. A Study on the Visual Menu Design Using Pinch Gestures on Touchscreens. In *2018 4th International Conference on Science and Technology (ICST)*, pages 1–5, August 2018.

[152] Nem Khan Dim, Kibum Kim, and Xiangshi Ren. Designing motion marking menus for people with visual impairments. *International Journal of Human-Computer Studies*, 109:79 – 88, 2018.

[153] Romain Dizier and Zacharie Kerger. *GestMan : towards an online gesture management system*. PhD Thesis, UCL - Ecole polytechnique de Louvain, 2018.

[154] Emmanuel Dubois, Mathieu Raynal, and Marcos Serrano. Taking Advantage of Rolling Gestures on a Multidimensionnal Mouse for Interacting with a Menubar. In *Proceedings of the 29th Conference on L'Interaction Homme-Machine*, IHM '17, pages 83–92, New York, NY, USA, 2018. ACM. event-place: Poitiers, France.

[155] Emmanuel Dubois, Marcos Serrano, and Mathieu Raynal. Rolling-Menu: Rapid Command Selection in Toolbars Using Roll Gestures with a Multi-DoF Mouse. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, pages 367:1–367:12, New York, NY, USA, 2018. ACM. event-place: Montreal QC, Canada.

[156] David J. Duke and Angel R. Puerta, editors. *Design, Specification and Verification of Interactive Systems'99, Proceedings of the Eurographics Workshop in Braga, Portugal, June 2-4, 1999*. Springer, 1999.

[157] Bruno Dumas, María Solórzano, and Beat Signer. Design guidelines for adaptive multimodal mobile input solutions. In Rohs et al. [417], pages 285–294.

[158] Rana Mohamed Eisa, Yassin El-Shanwany, Yomna Abdelrahman, and Wael Abouelsadat. LINEUp: List Navigation Using Edge Menu. In Patrick Horain, Catherine Achard, and Malik Mallem, editors, *Intelligent Human Computer Interaction*, Lecture Notes in Computer Science, pages 86–106, Cham, 2017. Springer International Publishing.

[159] C. Essert and A. Capobianco. Comparative study of the performances of several haptic modalities for a 3D menu. In *2010 IEEE Virtual Reality Conference (VR)*, pages 265–266, March 2010.

[160] Alan W Ezekiel and Bradford A Christian. *Automated system and method for dynamic menu construction in a graphical user interface*. Google Patents, April 1997.

[161] Shamal Faily, Nan Jiang, Huseyin Dogan, and Jacqui Taylor, editors. *HCI 2016 - Fusion! Proceedings of the 30th International BCS Human Computer Interaction Conference, BCS HCI 2016, Bournemouth University, Poole, UK, 11-15 July 2016*, Workshops in Computing. BCS, 2016.

[162] J. K. Fawcett. Supporting human interaction with the Sentient vehicle. In *Proceedings. The IEEE 5th International Conference on Intelligent Transportation Systems*, pages 307–312, September 2002.

[163] Jesse Feiler. Chapter 24 - Managing Menus. In Jesse Feiler, editor, *Mac OSX Developer Guide*, pages 517 – 523. Morgan Kaufmann, San Francisco, 2002.

[164] Leah Findlater and Krzysztof Z. Gajos. Design space and evaluation challenges of adaptive graphical user interfaces. *AI Magazine*, 30(4):68–73, 2009.

[165] Leah Findlater and Joanna McGrenere. A comparison of static, adaptive, and adaptable menus. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '04, pages 89–96, New York, NY, USA, 2004. ACM.

[166] Leah Findlater and Joanna McGrenere. A comparison of static, adaptive, and adaptable menus. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '04, page 89–96, New York, NY, USA, 2004. Association for Computing Machinery.

[167] Leah Findlater and Joanna McGrenere. Impact of screen size on performance, awareness, and user satisfaction with adaptive graphical user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, pages 1247–1256, New York, NY, USA, 2008. ACM.

[168] Leah Findlater, Karyn Moffatt, Joanna McGrenere, and Jessica Dawson. Ephemeral adaptation: The use of gradual onset to improve menu selection performance. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, pages 1655–1664. ACM, 2009.

[169] Sebastian Fischer and Stephan Schwan. Adaptively shortened pull down menus: location knowledge and selection efficiency. *Behaviour & Information Technology*, 27(5):439–444, 2008.

[170] Andrew W Fitzgibbon. Simultaneous linear estimation of multiple view geometry and lens distortion. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, volume 1, pages I–I. IEEE, 2001.

[171] George Fitzmaurice, Azam Khan, Robert Pieké, Bill Buxton, and Gordon Kurtenbach. Tracking Menus. In *Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology*, UIST '03, pages 71–79, New York, NY, USA, 2003. ACM. event-place: Vancouver, Canada.

[172] P. W. Foltz, S. E. Davies, P. G. Polson, and D. E. Kieras. Transfer between Menu Systems. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '88, pages 107–112, New York, NY, USA, 1988. Association for Computing Machinery. event-place: Washington, D.C., USA.

[173] Peter W. Foltz, Susan E. Davies, Peter G. Polson, and David Kieras. TRANSFER BETWEEN SIMILAR MENU SYSTEMS. *SIGCHI Bull.*, 19(3):63–65, January 1988.

[174] Greg Foster and Terence Foxcroft. Barrel Menu: A New Mobile Phone Menu for Feature Rich Devices. In *Proceedings of the South African Institute of Computer Scientists and Information Technologists Conference on Knowledge, Innovation and Leadership in a Diverse, Multidisciplinary Environment*, SAICSIT '11, pages 97–105, New York, NY, USA, 2011. ACM. event-place: Cape Town, South Africa.

[175] Fractal Foundation. Fractal foundation online course. https://fractalfoundation.org/OFC/OFC-index.htm, 2018. [Online: accessed 05-Dec-2019].

[176] Jeremie Francone, Gilles Bailly, Laurence Nigay, and Eric Lecolinet. Wavelet Menus: A Stacking Metaphor for Adapting Marking Menus to Mobile Devices. In *Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services*, MobileHCI '09, New York, NY, USA, 2009. Association for Computing Machinery. event-place: Bonn, Germany.

[177] Jérémie Francone, Gilles Bailly, Eric Lecolinet, Nadine Mandran, and Laurence Nigay. Wavelet Menus on Handheld Devices: Stacking Metaphor for Novice Mode and Eyes-free Selection for Expert Mode. In *Proceedings of the International Conference on Advanced Visual Interfaces*, AVI '10, pages 173–180, New York, NY, USA, 2010. ACM. event-place: Roma, Italy.

[178] Jérémie Francone, Gilles Bailly, Laurence Nigay, and Eric Lecolinet. Wavelet Menu: And Adaptation Of Marking Menus For Mobile Devices. In *Proceedings of the 21st International Conference on Association Francophone d'Interaction Homme-Machine*, IHM '09, pages 367–370, New York, NY, USA, 2009. Association for Computing Machinery. event-place: Grenoble, France.

[179] Y. Fujita and S. Lam. Menu driven user interface for home system. *IEEE Transactions on Consumer Electronics*, 40(3):587–597, August 1994.

[180] Yusuke Fukazawa, Mirai Hara, Masashi Onogi, and Hidetoshi Ueno. Automatic Mobile Menu Customization Based on User Operation History. In *Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services*, MobileHCI '09, New York, NY, USA, 2009. Association for Computing Machinery. event-place: Bonn, Germany.

[181] Masaharu Fukumoto, Hiroyuki Segawa, Osamu Oda, and Miwako Yoritate. *Customized menu system for hierarchical menu and television system with the same.* Google Patents, May 2002.

[182] G. W. Furnas. Generalized fisheye views. *SIGCHI Bull.*, 17(4):16–23, April 1986.

[183] Krzysztof Z. Gajos and Krysta Chauncey. The influence of personality traits and cognitive load on the use of adaptive user interfaces. In *Proceedings of the 22nd International Conference on Intelligent User Interfaces*, IUI '17, pages 301–306, New York, NY, USA, 2017. ACM.

[184] Krzysztof Z. Gajos, Mary Czerwinski, Desney S. Tan, and Daniel S. Weld. Exploring the design space for adaptive graphical user interfaces. In *Proceedings of the Working Conference on Advanced Visual Interfaces*, AVI '06, pages 201–208, New York, NY, USA, 2006. ACM.

[185] Krzysztof Z. Gajos, Katherine Everitt, Desney S. Tan, Mary Czerwinski, and Daniel S. Weld. Predictability and accuracy in adaptive user interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, pages 1271–1274, New York, NY, USA, 2008. ACM.

[186] Shengkui Gao and Viktor Gruev. Bilinear and bicubic interpolation methods for division of focal plane polarimeters. *Optics express*, 19(27):26161–26173, 2011.

[187] Harry B Garland. *Command entry highlight editing for a menu selection system and method*. Google Patents, June 2001.

[188] P. Gaucher, F. Argelaguet, J. Royan, and A. Lécuyer. A novel 3D carousel based on pseudo-haptic feedback and gestural interaction for virtual showcasing. In *2013 IEEE Symposium on 3D User Interfaces (3DUI)*, pages 55–58, March 2013.

[189] Varun Gaur, Md. Sami Uddin, and Carl Gutwin. Multiplexing Spatial Memory: Increasing the Capacity of FastTap Menus with Multiple Tabs. In *Proceedings of the 20th International Conference on Human-Computer Interaction with Mobile Devices and Services*, MobileHCI '18, pages 22:1–22:13, New York, NY, USA, 2018. ACM. event-place: Barcelona, Spain.

[190] S. Gebhardt, S. Pick, F. Leithold, B. Hentschel, and T. Kuhlen. Extended Pie Menus for Immersive Virtual Environments. *IEEE Transactions on Visualization and Computer Graphics*, 19(4):644–651, April 2013.

[191] S. Gebhardt, S. Pick, T. Oster, B. Hentschel, and T. Kuhlen. An evaluation of a smart-phone-based menu system for immersive virtual environments. In *2014 IEEE Symposium on 3D User Interfaces (3DUI)*, pages 31–34, March 2014.

[192] Sascha Gebhardt, Till Petersen-Krau, Sebastian Pick, Dominik Rausch, Christian Nowke, Thomas Knott, Patric Schmitz, Daniel Zielasko, Bernd Hentschel, and Torsten W. Kuhlen. Vista Widgets: A Framework for Designing 3D User Interfaces from Reusable Interaction Building Blocks. In *Proceedings of the 22Nd ACM Conference on Virtual Reality Software and Technology*, VRST '16, pages 251–260, New York, NY, USA, 2016. ACM. event-place: Munich, Germany.

[193] D. Gerber and D. Bechmann. The spin menu: a menu system for virtual environments. In *IEEE Proceedings. VR 2005. Virtual Reality, 2005.*, pages 271–272, March 2005.

[194] Kentaro Go and Hiroki Kasuga. Multi-tapping Shortcut: A Technique for Augmenting Linear Menus on Multi-touch Surface. In *Proceedings of the 10th Asia Pacific Conference on Computer Human Interaction*, APCHI '12, pages 209–218, New York, NY, USA, 2012. ACM. event-place: Matsue-city, Shimane, Japan.

[195] Camille Gobert, Kashyap Todi, Gilles Bailly, and Antti Oulasvirta. SAM: a modular framework for self-adapting web menus. In Wai-Tat Fu, Shimei Pan, Oliver Brdiczka, Polo Chau, and Gaelle Calvary, editors, *Proceedings of the 24th International Conference on Intelligent User Interfaces, IUI 2019, Marina del Ray, CA, USA, March 17-20, 2019*, pages 481–484. ACM, 2019.

[196] Yamato Gomi and Katsuhiko Onishi. Study of Tile Menu Selection Technique Using the Relative Position of Joints for Gesture Operation. In *HCI (27)*, volume 528 of *Communications in Computer and Information Science*, pages 481–484. Springer, 2015.

[197] Mikhail V. Goubko and Alexander I. Danilenko. An Automated Routine for Menu Structure Optimization. In *Proceedings of the 2nd ACM SIGCHI Symposium on Engineering Interactive Computing Systems*, EICS '10, pages 67–76, New York, NY, USA, 2010. Association for Computing Machinery. event-place: Berlin, Germany.

[198] John Gray. The Role of Menu Titles as a Navigational Aid in Hierarchical Menus. *SIGCHI Bull.*, 17(3):33–40, January 1986.

[199] John S. Gray. Menu Stacking—Help or Hindrance? *SIGCHI Bull.*, 25(3):52–57, July 1993.

[200] Paul E. Green and V. Srinivasan. Conjoint analysis in marketing: New developments with implications for research and practice. *Journal of Marketing*, 54(4):3–19, 1990.

[201] Saul Greenberg and Ian H. Witten. Adaptive personalized interfaces - a question of viability. *Behaviour & Information Technology*, 4(1):31–45, 1985.

[202] Asnat Greenstein-Messica, Lior Rokach, and Michael Friedman. Session-Based Recommendations Using Item Embedding. In *Proceedings of the 22nd International Conference on Intelligent User Interfaces*, IUI '17, pages 629–633, New York, NY, USA, 2017. Association for Computing Machinery. event-place: Limassol, Cyprus.

[203] Keith D. Gregory. Menus. In Keith D. Gregory, editor, *Programming with Motif^{TM}*, pages 261–303. Springer, New York, NY, 1992.

[204] Kim T Gribbon and Donald G Bailey. A novel approach to real-time bilinear interpolation. In *Proceedings. DELTA 2004. Second IEEE International Workshop on Electronic Design, Test and Applications*, pages 126–131. IEEE, 2004.

[205] KT Gribbon, CT Johnston, and Donald G Bailey. A real-time fpga implementation of a barrel distortion correction algorithm with bilinear interpolation. In *Image and Vision Computing New Zealand*, pages 408–413, 2003.

[206] R. Grosse, I. Thouvenin, D. Lenne, and S. Aubry. Circular and Linear Menus: User Studies on Gaze-Based Interaction Methods. In *Proceedings of the 29th Conference on l'Interaction Homme-Machine*, IHM '17, pages 291–297, New York, NY, USA, 2017. Association for Computing Machinery. event-place: Poitiers, France.

[207] Romain Grosse, Dominique Lenne, Indira Thouvenin, and Stéphane Aubry. Analyzing Eye-gaze Interaction Modalities in Menu Navigation. In *VISIGRAPP (2: HUCAPP)*, pages 17–25. SciTePress, 2018.

[208] Tovi Grossman, Ken Hinckley, Patrick Baudisch, Maneesh Agrawala, and Ravin Balakrishnan. Hover Widgets: Using the Tracking State to Extend the Capabilities of Pen-operated Devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '06, pages 861–870, New York, NY, USA, 2006. ACM. event-place: Montréal, Québec, Canada.

[209] François Guimbretière and Chau Nguyen. Bimanual Marking Menu for near Surface Interactions. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 825–828, New York, NY, USA, 2012. Association for Computing Machinery. event-place: Austin, Texas, USA.

[210] Jan Gulliksen, Morten Borup Harning, Philippe A. Palanque, Gerrit C. van der Veer, and Janet Wesson, editors. *Engineering Interactive Systems - EIS 2007 Joint Working Conferences, EHCI 2007, DSV-IS 2007, HCSE 2007, Salamanca, Spain, March 22-24, 2007. Selected Papers*, volume 4940 of *Lecture Notes in Computer Science*. Springer, 2008.

[211] Shrey Gupta and Michael J. McGuffin. Multitouch Radial Menu Integrating Command Selection and Control of Arguments with Up to 4 Degrees of Freedom. In *Proceedings of the International Working Conference on Advanced Visual Interfaces*, AVI '16, pages 256–263, New York, NY, USA, 2016. ACM. event-place: Bari, Italy.

[212] Martin Hachet, Fabrice Decle, Sebastian Knödel, and Pascal Guitton. Navidget for 3D interaction: Camera positioning and further uses. *International Journal of Human-Computer Studies*, 67(3):225 – 236, 2009.

[213] Lynne E. Hall and Xavi Bescos. Menu — what menu? *Interacting with Computers*, 7(4):383 – 394, 1995.

[214] Dianyuan Han. Comparison of commonly used image interpolation methods. In *Proceedings of the 2nd international conference on computer science and electronics engineering*. Atlantis Press, 2013.

[215] Miran Han and Peom Park. A Study of Interface Design for Widgets in Web Services Through Usability Evaluation. In *Proceedings of the 2Nd International Conference on Interaction Sciences: Information Technology, Culture and Human*, ICIS '09, pages 1013–1018, New York, NY, USA, 2009. ACM. event-place: Seoul, Korea.

[216] Sung H Han and Jiyoung Kwahk. Design of a menu for small displays presenting a single item at a time. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, volume 38, pages 360–364. SAGE Publications Sage CA: Los Angeles, CA, 1994.

[217] Maike Harberts, Nils Porrmann, and Steffen Skopp. *Graphical User Interface Having An Orbital Menu System*. Google Patents, September 2012.

[218] Robert Hardy and Enrico Rukzio. Touch & Interact: Touch-Based Interaction of Mobile Phones with Displays. In *Proceedings of the 10th International Conference on Human Computer Interaction with Mobile Devices and Services*, MobileHCI '08, pages 245–254, New York, NY, USA, 2008. Association for Computing Machinery. event-place: Amsterdam, The Netherlands.

[219] Robert Hardy and Enrico Rukzio. Touch & Interact: Touch-Based Interaction with a Tourist Application. In *Proceedings of the 10th International Conference on Human Computer Interaction with Mobile Devices and Services*, MobileHCI '08, pages 531–534, New York, NY, USA, 2008. Association for Computing Machinery. event-place: Amsterdam, The Netherlands.

[220] Sara Harris and Alan W. Biermann. Mouse Selection Versus Voice Selection of Menu Items. *I. J. Speech Technology*, 5(4):389–402, 2002.

[221] Claudia Hauff, Matthias Hagen, Anna Beyer, and Benno Stein. Towards Realistic Known-Item Topics for the ClueWeb. In *Proceedings of the 4th Information Interaction in Context Symposium*, IIIX '12, pages 274–277, New York, NY, USA, 2012. Association for Computing Machinery. event-place: Nijmegen, The Netherlands.

[222] Michael J. Heffler. Description of a Menu Creation and Interpretation System. *Softw., Pract. Exper.*, 12(3):269–281, 1982.

[223] Chris Heilmann. Adding Special Effects. In Christian Heilmann and Mark Norman Francis, editors, *Web Development Solutions: Ajax, APIs, Libraries, and Hosted Services Made Easy*, pages 199–223. Apress, Berkeley, CA, 2007.

[224] Florian Heimerl, Steffen Lohmann, Simon Lange, and Thomas Ertl. Word cloud explorer: Text analytics based on word clouds. In *47th Hawaii International Conference on System Sciences, HICSS 2014, Waikoloa, HI, USA, January 6-9, 2014* [7], pages 1833–1842.

[225] Seongkook Heo, Jingun Jung, and Geehyuk Lee. MelodicTap: Fingering Hotkey for Touch Tablets. In *Proceedings of the 28th Australian Conference on Computer-Human Interaction*, OzCHI '16, pages 396–400, New York, NY, USA, 2016. ACM. event-place: Launceston, Tasmania, Australia.

[226] Tobias Hesselmann, Stefan Flöring, and Marwin Schmitt. Stacked Half-Pie Menus: Navigating Nested Menus on Interactive Tabletops. In *Proceedings of the ACM International Conference on Interactive Tabletops and Surfaces*, ITS '09, pages 173–180, New York, NY, USA, 2009. ACM. event-place: Banff, Alberta, Canada.

[227] Harry Hochheiser and Ben Shneiderman. Performance Benefits of Simultaneous Over Sequential Menus as Task Complexity Increases. *Int. J. Hum. Comput. Interaction*, 12(2):173–192, 2000.

[228] Christopher W. V. Hogue and Hugh B. Fackrell. The menu workbench: An automatic menu generator for bio-medical programs. *International Journal of Bio-Medical Computing*, 21(3):253 – 264, 1987.

[229] Vera Hollink, Maarten van Someren, and Bob J. Wielinga. A semi-automatic usage-based method for improving hyperlink descriptions in menus. *International Journal of Human-Computer Studies*, 67(4):366 – 381, 2009.

[230] Y. Horiguchi, S. An, T. Sawaragi, and H. Nakanishi. Analysis of menu selection behavior using information scent model. In *The 6th International Conference on Soft Computing and Intelligent Systems, and The 13th International Symposium on Advanced Intelligence Systems*, pages 1514–1519, November 2012.

[231] Y. Horiguchi, W. Gejo, T. Sawaragi, and H. Nakanishi. Menu system design based on conceptual dependency structures between functional elements. In *2011 IEEE/SICE International Symposium on System Integration (SII)*, pages 262–266, December 2011.

[232] Yukio Horiguchi, Shinsu An, Tetsuo Sawaragi, and Hiroaki Nakanishi. Menu Hierarchy Generation Based on Syntactic Dependency Structures in Item Descriptions. In Sakae Yamamoto, editor, *Human Interface and the Management of Information. Information and Knowledge Design and Evaluation*, Lecture Notes in Computer Science, pages 157–166, Cham, 2014. Springer International Publishing.

[233] Kasper Hornbundefinedk and Morten Hertzum. Untangling the Usability of Fisheye Menus. *ACM Trans. Comput.-Hum. Interact.*, 14(2):6–es, August 2007.

[234] Eric Horvitz. Principles of mixed-initiative user interfaces. In Williams and Altom [509], pages 159–166.

[235] Ahammad Hossain, Md Nurujjaman, and Dr Ahmed. A review of fractals properties: Mathematical approach. *Journal of Applied Mathematics*, 5:98–105, 06 2017.

[236] Yi-Ta Hsieh. Tactile Interaction Gestures: A Wearable Hand Tactile System for Interacting with Ubiquitous Information. In *Adjunct Proceedings of the 2015 ACM International Joint Conference on Pervasive and Ubiquitous Computing and Proceedings of the 2015 ACM International Symposium on Wearable Computers*, UbiComp/ISWC'15 Adjunct, pages 539–544, New York, NY, USA, 2015. Association for Computing Machinery. event-place: Osaka, Japan.

[237] Sheng-Cheng Huang, I-Fan Chou, and Randolph G Bias. Empirical evaluation of a popular cellular phone's menu system: Theory meets practice. *Journal of Usability Studies*, 1(2):91–108, 2006.

[238] Hyunjeong Lee and Joonah Park. Touch play pool: Touch gesture interaction for mobile multifunction devices. In *2012 IEEE International Conference on Consumer Electronics (ICCE)*, pages 291–292, January 2012.

[239] Adrian Iftene and Jean Vanderdonckt, editors. *13th International Conference on Human Computer Interaction, RoCHI 2016, Iasi, Romania, September 8-9, 2016*. Matrix Rom, 2016.

[240] Y. Ikei, H. Yamazaki, K. Hirota, and M. Hirose. vCocktail: Multiplexed-voice Menu Presentation Method for Wearable Computers. In *IEEE Virtual Reality Conference (VR 2006)*, pages 183–190, March 2006.

[241] Kaori Ikematsu, Mana Sasagawa, and Itiro Siio. 2.5 Dimensional Panoramic Viewing Technique Utilizing a Cylindrical Mirror Widget. In *Proceedings of the 29th Annual Symposium on User Interface Software and Technology*, UIST '16 Adjunct, pages 145–146, New York, NY, USA, 2016. Association for Computing Machinery. event-place: Tokyo, Japan.

[242] M. Imran and A. M. Hussaan. Adaptive amp; dynamic interfaces for automated teller machines using clusters. In *2018 International Conference on Computing, Mathematics and Engineering Technologies (iCoMET)*, pages 1–6, March 2018.

[243] Pourang Irani, Peer Shajahan, and Christel Kemke. VoiceMarks: restructuring hierarchical voice menus for improving navigation. *International Journal of Speech Technology*, 9(3):75–94, December 2006.

[244] M. Iwata, H. Miyamot, T. Hara, K. Shimatani, T. Mashita, K. Kiyokawa, S. Nishio, and H. Takemura. A Menu-Based Content Search System Based on Relationships between Mobile User Context and Information Needs. In *2014 28th International Conference on Advanced Information Networking and Applications Workshops*, pages 197–202, May 2014.

[245] Julie A Jacko and Gavriel Salvendy. Hierarchical menu design: Breadth, depth, and task complexity. *Perceptual and Motor skills*, 82(3_suppl):1187–1201, 1996.

[246] Julie A. Jacko, Gavriel Salvendy, and Richard J. Koubek. Modelling of menu design in computerized work. *Interacting with Computers*, 7(3):304 – 330, 1995.

[247] Wallace Jackson. Android UI Local Menus: The ContextMenu Class and PopupMenu Class. In Wallace Jackson, editor, *Pro Android UI*, pages 131–152. Apress, Berkeley, CA, 2014.

[248] Wallace Jackson. Android UI Options Menus: OptionsMenu Class and the Action Bar. In Wallace Jackson, editor, *Pro Android UI*, pages 97–129. Apress, Berkeley, CA, 2014.

[249] Bertin Jacques et al. Sémiologie graphique. *Paris, Mouton/Gauthier-Villars*, 1967.

[250] Mohit Jain and Ravin Balakrishnan. User Learning and Performance with Bezel Menus. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 2221–2230, New York, NY, USA, 2012. ACM. event-place: Austin, Texas, USA.

[251] Anthony Jameson, Silvia Gabrielli, and Antti Oulasvirta. Users' preferences regarding intelligent user interfaces: Differences among users and changes over time. In *Proceedings of the 14th International Conference on Intelligent User Interfaces*, IUI '09, pages 497–498, New York, NY, USA, 2009. ACM.

[252] Wesley Jamison and C. Michael Lewis. UNIXTUTOR: A MENU-BASED TRANSITIONAL INTERFACE. *SIGCHI Bull.*, 21(3):41–45, January 1990.

[253] Myounghoon Jeon, Siddharth Gupta, Benjamin K. Davison, and Bruce N. Walker. Auditory Menus Are Not Just Spoken Visual Menus: A Case Study of "Unavailable" Menu Items. In *CHI '10 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '10, pages 3319–3324, New York, NY, USA, 2010. ACM. event-place: Atlanta, Georgia, USA.

[254] Myounghoon Jeon and Bruce N. Walker. Spindex (Speech Index) Improves Auditory Menu Acceptance and Navigation Performance. *ACM Trans. Access. Comput.*, 3(3), April 2011.

[255] Kristiina Jokinen, Antti Kerminen, Mauri Kaipainen, Tommi Jauhiainen, Graham Wilcock, Markku Turunen, Jaakko Hakulinen, Jukka Kuusisto, and Krista Lagus. Adaptive Dialogue Systems - Interaction with Interact. In *Proceedings of the 3rd SIGdial Workshop on Discourse and Dialogue - Volume 2*, SIGDIAL '02, pages 64–73, USA, 2002. Association for Computational Linguistics. event-place: Philadelphia, Pennsylvania.

[256] Raine A. Kajastila and Tapio Lokki. Eyes-free interaction with free-hand gestures and auditory menus. *Int. J. Hum.-Comput. Stud.*, 71(5):627–640, 2012.

[257] Yvonne Kammerer, Katharina Scheiter, and Wolfgang Beinhauer. Looking My Way Through the Menu: The Impact of Menu Design and Multimodal Input on Gaze-based Menu Selection. In *Proceedings of the 2008 Symposium on Eye Tracking Research & Applications*, ETRA '08, pages 213–220, New York, NY, USA, 2008. ACM. event-place: Savannah, Georgia.

[258] Victor Kaptelinin. Item Recognition in Menu Selection: The Effect of Practice. In *INTERACT '93 and CHI '93 Conference Companion on Human Factors in Computing Systems*, CHI '93, pages 183–184, New York, NY, USA, 1993. Association for Computing Machinery. event-place: Amsterdam, The Netherlands.

[259] A. G. Kara and K. Cagiltay. A Comparison of Cascading Horizontal and Vertical Menus with Overlapping and Traditional Designs in Terms of Effectiveness, Error Rate and user Satisfaction. In *2007 IEEE 18th International Symposium on Personal, Indoor and Mobile Radio Communications*, pages 1–4, September 2007.

[260] Hani Karam and Jiro Tanaka. Two-Handed Interactive Menu: An Application of Asymmetric Bimanual Gestures and Depth Based Selection Techniques. In Sakae Yamamoto, editor, *Human Interface and the Management of Information. Information and Knowledge Design and Evaluation*, Lecture Notes in Computer Science, pages 187–198, Cham, 2014. Springer International Publishing.

[261] John Karat, James E. McDonald, and Matthew P. Anderson. A Comparison of Menu Selection Techniques: Touch Panel, Mouse and Keyboard. *International Journal of Man-Machine Studies*, 25(1):73–88, 1986.

[262] J. Karimov, M. Ozbayoglu, B. Tavli, and E. Dogdu. Generic menu optimization for multi-profile customer systems. In *2015 IEEE International Symposium on Systems Engineering (ISSE)*, pages 163–169, September 2015.

[263] Jofish Kaye and Paul Dourish. Special issue on science fiction and ubiquitous computing. *Personal and Ubiquitous Computing*, 18(4):765–766, April 2014.

[264] M. Kemp, S. Mills, and J. Noyes. Usability of menu-versus icon-based software in automatic test equipment (ATE) control. In *IEE Colloquium on Man-Machine Interfaces for Instrumentation*, pages 5/1–5/5, October 1995.

[265] Frederic Kerber, Tobias Kiefer, Markus Löchtefeld, and Antonio Krüger. Investigating Current Techniques for Opposite-Hand Smartwatch Interaction. In *Proceedings of the 19th International Conference on Human-Computer Interaction with Mobile Devices and Services*, MobileHCI '17, New York, NY, USA, 2017. Association for Computing Machinery. event-place: Vienna, Austria.

[266] Iyad Khaddam, Sara Bouzit, Gaëlle Calvary, and Denis Chêne. Menuergo: Computer-aided design of menus by automated guideline review. In *Actes de la 28ième ConféRence Francophone sur l'Interaction Homme-Machine*, IHM '16, pages 36–47, New York, NY, USA, 2016. ACM.

[267] Iyad Khaddam and Jean Vanderdonckt. Flippable user interfaces for internationalization. In *Proceedings of the 3rd ACM SIGCHI Symposium on Engineering Interactive Computing Systems*, EICS '11, pages 223–228, New York, NY, USA, 2011. ACM.

[268] John I Kiger. The depth/breadth trade-off in the design of menu-driven user interfaces. *International journal of man-machine studies*, 20(2):201–213, 1984.

[269] Hyun K. Kim, Jaehyun Park, Kyeong Park, and Mungyeong Choe. Analyzing thumb interaction on mobile touchpad devices. *International Journal of Industrial Ergonomics*, 67:201 – 209, 2018.

[270] Hyungon Kim, Gun Lee, and Mark Billinghurst. A Non-Linear Mapping Technique for Bare-Hand Interaction in Large Virtual Environments. In *Proceedings of the Annual Meeting of the Australian Special Interest Group for Computer Human Interaction*, OzCHI '15, pages 53–61, New York, NY, USA, 2015. Association for Computing Machinery. event-place: Parkville, VIC, Australia.

[271] Kyungdoh Kim, Robert W. Proctor, and Gavriel Salvendy. Menu Design in Cell Phones: Use of 3D Menus. In Julie A. Jacko, editor, *Human-Computer Interaction. Ambient, Ubiquitous and Intelligent Interaction*, Lecture Notes in Computer Science, pages 48–57, Berlin, Heidelberg, 2009. Springer.

[272] Kyungdoh Kim, Robert W. Proctor, and Gavriel Salvendy. Comparison of 3D and 2D menus for cell phones. *Computers in Human Behavior*, 27(5):2056 – 2066, 2011.

[273] Scott R. Klemmer, Michael Thomsen, Ethan Phelps-Goodman, Robert Lee, and James A. Landay. Where do web sites come from?: capturing and interacting with design history. In *Proceedings of the SIGCHI conference on Human factors in computing systems Changing our world, changing ourselves - CHI '02*, page 1, Minneapolis, Minnesota, USA, 2002. ACM Press.

[274] R. Komerska and C. Ware. A study of haptic linear and pie menus in a 3D fish tank VR environment. In *12th International Symposium on Haptic Interfaces for Virtual Environment and Teleoperator Systems, 2004. HAPTICS '04. Proceedings.*, pages 224–231, March 2004.

[275] S. Koo, K. Park, H. Kim, and D. Kwon. A dual-layer user model based cognitive system for user-adaptive service robots. In *2011 RO-MAN*, pages 59–64, July 2011.

[276] Lim Chee Koon. A Case Study of Icon-Scenario Based Animated Menu's Concept Development. In *Proceedings of the 8th Conference on Human-Computer Interaction with Mobile Devices and Services*, MobileHCI '06, pages 177–180, New York, NY, USA, 2006. Association for Computing Machinery. event-place: Helsinki, Finland.

[277] Larry Koved and Ben Shneiderman. Embedded Menus: Selecting Items in Context. *Commun. ACM*, 29(4):312–318, April 1986.

[278] M. Kranz, P. Holleis, and A. Schmidt. DistScroll - a new one-handed interaction device. In *25th IEEE International Conference on Distributed Computing Systems Workshops*, pages 499–505, June 2005.

[279] Nikhil Krishnaswamy, Pradyumna Narayana, Isaac Wang, Kyeongmin Rim, Rahul Bangar, Dhruva Patil, Gururaj Mulay, Ross Beveridge, Jaime Ruiz, Bruce Draper, and James Pustejovsky. Communicating and Acting: Understanding Gesture in Simulation Semantics. *IWCS 2017 — 12th International Conference on Computational Semantics — Short papers*, 2017.

[280] G. Kubryk and M. Kubryk. Three ways to construct adaptative WEB menus using ants. In *2008 Second International Conference on Research Challenges in Information Science*, pages 349–360, June 2008.

[281] Thomas Kühme, Uwe Malinowski, and James D. Foley. Facilitating interactive tool selection by adaptive prompting. In Ashlund et al. [41], pages 149–150.

[282] Arun Kulshreshth and Joseph J. LaViola. Exploring the Usefulness of Finger-Based 3D Gesture Menu Selection. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, pages 1093–1102, New York, NY, USA, 2014. Association for Computing Machinery. event-place: Toronto, Ontario, Canada.

[283] André Kunert, Alexander Kulik, Christopher Lux, and Bernd Fröhlich. Facilitating System Control in Ray-Based Interaction Tasks. In *Proceedings of the 16th ACM Symposium on Virtual Reality Software and Technology*, VRST '09, pages 183–186, New York, NY, USA, 2009. Association for Computing Machinery. event-place: Kyoto, Japan.

[284] Ekaterina Kurdyukova, Mohammad Obaid, and Elisabeth André. Direct, Bodily or Mobile Interaction? Comparing Interaction Techniques for Personalized Public Displays. In *Proceedings of the 11th International Conference on Mobile and Ubiquitous Multimedia*, MUM '12, New York, NY, USA, 2012. Association for Computing Machinery. event-place: Ulm, Germany.

[285] Gordon Kurtenbach and William Buxton. The Limits of Expert Performance Using Hierarchic Marking Menus. In *Proceedings of the INTERACT '93 and CHI '93 Conference on Human Factors in Computing Systems*, CHI '93, pages 482–487, New York, NY, USA, 1993. ACM. event-place: Amsterdam, The Netherlands.

[286] Gordon Kurtenbach, George W. Fitzmaurice, Russell N. Owen, and Thomas Baudel. The Hotbox: Efficient Access to a Large Number of Menu-items. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '99, pages 231–237, New York, NY, USA, 1999. ACM. event-place: Pittsburgh, Pennsylvania, USA.

[287] Gordon Kurtenbach, Abigail Sellen, and William Buxton. An Empirical Evaluation of Some Articulatory and Cognitive Aspects of Marking Menus. *Human-Computer Interaction*, 8(1):1–23, 1993.

[288] Thomas Költringer, Martin Tomitsch, Karin Kappel, Daniel Kalbeck, and Thomas Grechenig. Implications for Designing the User Experience of DVD Menus. In *CHI '05 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '05, pages 1565–1568, New York, NY, USA, 2005. ACM. event-place: Portland, OR, USA.

[289] John Lamping, Ramana Rao, and Peter Pirolli. A focus+ context technique based on hyperbolic geometry for visualizing large hierarchies. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 401–408, 1995.

[290] John Lamping, Ramana Rao, and Peter Pirolli. A focus+context technique based on hyperbolic geometry for visualizing large hierarchies. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '95, pages 401–408, New York, NY, USA, 1995. ACM Press/Addison-Wesley Publishing Co.

[291] T. K. Landauer and D. W. Nachbar. Selection from Alphabetic and Numeric Menu Trees Using a Touch Screen: Breadth, Depth, and Width. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '85, pages 73–78, New York, NY, USA, 1985. Association for Computing Machinery. event-place: San Francisco, California, USA.

[292] David M. Lane, H. Albert Napier, Richard R. Batsell, and John L. Naman. Predicting the Skilled Use of Hierarchical Menus With the Keystroke-Level Model. *Human-Computer Interaction*, 8(2):185–192, 1993.

[293] Ronald S Lane and Miriam Weiss Lane. *Hierarchical menu bar system with dynamic graphics and text windows*. Google Patents, December 1997.

[294] Talia Lavie and Joachim Meyer. Benefits and costs of adaptive user interfaces. *International Journal of Human-Computer Studies*, 68(8):508 – 524, 2010.

[295] Dong-Seok Lee and Wan Chul Yoon. Quantitative results assessing design issues of selection-supportive menus. *International Journal of Industrial Ergonomics*, 33(1):41 – 52, 2004.

[296] H. Lee, Y. Choi, and Y. Kim. An adaptive user interface based on spatiotemporal structure learning. In *2011 IEEE Consumer Communications and Networking Conference (CCNC)*, pages 923–927, Jan 2011.

[297] H. Lee, D. Kim, and W. Woo. Graphical Menus Using a Mobile Phone for Wearable AR Systems. In *2011 International Symposium on Ubiquitous Virtual Reality*, pages 55–58, July 2011.

[298] Hyeongmook Lee and Woontack Woo. Tangible Spin Cube for 3D Ring Menu in Real Space. In *CHI '10 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '10, pages 4147–4152, New York, NY, USA, 2010. Association for Computing Machinery. event-place: Atlanta, Georgia, USA.

[299] Seung Eun Lee and Geehyuk Lee. K-menu: A Keyword-based Dynamic Menu Interface for Small Computers. In *CHI '07 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '07, pages 2543–2548, New York, NY, USA, 2007. ACM. event-place: San Jose, CA, USA.

[300] Ioannis Leftheriotis and Konstantinos Chorianopoulos. Multi-user Chorded Toolkit for Multi-touch Screens. In *Proceedings of the 3rd ACM SIGCHI Symposium on Engineering Interactive Computing Systems*, EICS '11, pages 161–164, New York, NY, USA, 2011. ACM. event-place: Pisa, Italy.

[301] Daniel Leithinger and Michael Haller. Improving menu interaction for cluttered tabletop setups with user-drawn path menus. In *Second Annual IEEE International Workshop on Horizontal Interactive Human-Computer Systems (TABLETOP'07)*, pages 121–128. IEEE, 2007.

[302] James Lee Lentz. *Display interface to a computer controlled display system with variable comprehensiveness levels of menu items dependent upon size of variable display screen available for menu item display.* Google Patents, August 2007.

[303] G. Julian Lepinski, Tovi Grossman, and George Fitzmaurice. The Design and Evaluation of Multitouch Marking Menus. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '10, pages 2233–2242, New York, NY, USA, 2010. ACM. event-place: Atlanta, Georgia, USA.

[304] Howard Levene. Robust tests for equality of variances. In Ingram Olkin and Harold Hotelling et al., editors, *Contributions to Probability and Statistics: Essays in Honor of Harold Hotelling*, pages 278–292. Stanford University Press, Palo Alto, CA, USA, 1960.

[305] Huizhi Liang, Yue Xu, Yuefeng Li, and Richi Nayak. Personalized Recommender System Based on Item Taxonomy and Folksonomy. In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, CIKM '10, pages 1641–1644, New York, NY, USA, 2010. Association for Computing Machinery. event-place: Toronto, ON, Canada.

[306] Alessandro Liberati, Douglas G. Altman, Jennifer Tetzlaff, Cynthia Mulrow, Peter C. Gøtzsche, John P. A. Ioannidis, Mike Clarke, P. J. Devereaux, Jos Kleijnen, and David Moher. The PRISMA Statement for Reporting Systematic Reviews and Meta-Analyses of Studies That Evaluate Health Care Interventions: Explanation and Elaboration. *PLoS Medicine*, 6(7):e1000100, July 2009.

[307] Rensis Likert. A technique for the measurement of attitudes. *Archives of Psychology*, 22(140):55–, 1932.

[308] Youn-kyung Lim, Erik Stolterman, Heekyoung Jung, and Justin Donaldson. Interaction Gestalt and the Design of Aesthetic Interactions. In *Proceedings of the 2007 Conference on Designing Pleasurable Products and Interfaces*, DPPI '07, pages 239–254, New York, NY, USA, 2007. Association for Computing Machinery. event-place: Helsinki, Finland.

[309] Ming-Chyuan Lin, Yi-Hsien Lin, Chun-Chun Lin, and Jenn-Yang Lin. A Study on the Interface Design of a Functional Menu and Icons for In-Vehicle Navigation Systems. In *HCI (13)*, volume 8522 of *Lecture Notes in Computer Science*, pages 261–272. Springer, 2014.

[310] Yusan Lin, Peifeng Yin, and Wang-Chien Lee. Modeling Menu Bundle Designs of Crowdfunding Projects. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, CIKM '17, pages 1079–1088, New York, NY, USA, 2017. Association for Computing Machinery. event-place: Singapore, Singapore.

[311] N. G. Lipari and C. W. Borst. Handymenu: Integrating menu selection into a multifunction smartphone-based VR controller. In *2015 IEEE Symposium on 3D User Interfaces (3DUI)*, pages 129–132, March 2015.

[312] Yidan Liu, Min Xie, and Laks V.S. Lakshmanan. Recommending User Generated Item Lists. In *Proceedings of the 8th ACM Conference on Recommender Systems*, RecSys '14, pages 185–192, New York, NY, USA, 2014. Association for Computing Machinery. event-place: Foster City, Silicon Valley, California, USA.

[313] Steffen Lohmann, Jürgen Ziegler, and Lena Tetzlaff. Comparison of tag cloud layouts: Task-related performance and visual exploration. In Tom Gross, Jan Gulliksen, Paula Kotzé, Lars Oestreicher, Philippe Palanque, Raquel Oliveira Prates, and Marco Winckler, editors, *Human-Computer Interaction – INTERACT 2009*, pages 392–404, Berlin, Heidelberg, 2009. Springer Berlin Heidelberg.

[314] Víctor López-Jaquero, Jean Vanderdonckt, Francisco Montero Simarro, and Pascual González. Towards an extended model of user interface adaptation: The isatine framework. In Gulliksen et al. [210], pages 374–392.

[315] Gaëtan Lorho, Jarmo Hiipakka, and Juha Marila. Structured Menu Presentation Using Spatial Sound Separation. In Fabio Paternò, editor, *Human Computer Interaction with Mobile Devices*, Lecture Notes in Computer Science, pages 419–424, Berlin, Heidelberg, 2002. Springer.

[316] Dimitri Lozeve. Generating and representing l-systems. https://www.lozeve.com/posts/lsystems, 2018. [Online: accessed 17-March-2020].

[317] Eric Jui-Lin Lu and Rai-Fu Chen. Design and implementation of a fine-grained menu control processor for web-based information systems. *Future Generation Comp. Syst.*, 19(7):1105–1119, 2003.

[318] John F. Lucas, Ji-Sun Kim, and Doug A. Bowman. Resizing Beyond Widgets: Object Resizing Techniques for Immersive Virtual Environments. In *CHI '05 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '05, pages 1601–1604, New York, NY, USA, 2005. ACM. event-place: Portland, OR, USA.

[319] P. J. Lyons, M. Pitchforth, D. Page, T. Given, and M. D. Apperley. The oval menu-evolution and evaluation of a widget. In *Proceedings Sixth Australian Conference on Computer-Human Interaction*, pages 252–259, November 1996.

[320] Fei Lyu, Rui Xi, Yuxin Han, and Yujie Liu. MagicMark: a marking menu using 2D direction and 3D depth information. *Science China Information Sciences*, 61(6):064101, May 2018.

[321] Jock D Mackinlay, George G Robertson, and Stuart K Card. The perspective wall: Detail and context smoothly integrated. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 173–176, 1991.

[322] Karen R. Mahach, Deborah A. Boehm-Davis, and Robert W. Holt. The effects of mice and pull-down menus versus command-driven interfaces on writing. *Int. J. Hum. Comput. Interaction*, 7(3):213–234, 1995.

[323] Arun Mahanta, Hemanta Sarmah, and Gautam Choudhury. Iterated function systems as a generator of fractal objects. 03 2018.

[324] Khalid Majrashi. Post-transitioning user performance on cross-device menu interfaces. *International Journal of Human-Computer Studies*, 130:130 – 151, 2019.

[325] Mark A Malamud, John E Elsbree, Laura J Butler, and David A Barnes Jr. *Context sensitive menu system/menu behavior*. Google Patents, September 1997.

[326] Benoit Mandelbrot. How long is the coast of britain? statistical self-similarity and fractional dimension. *Science*, 156(3775):636–638, 1967.

[327] Xiaoyang Mao, Yuji Hatanaka, Atsumi Imamiya, Yuki Kato, and Kentaro Go. Visualizing computational wear with physical wear. In Pier-Luigi Emiliani and Constantine Stephanidis, editors, *Proceedings of the 6th ERCIM Workshop "User Interfaces for All"*, UI4All '00, Pisa, Italy, 2000. CNR-IROE.

[328] Monica A. Marics and George Engelbeck. Chapter 45 - Designing Voice Menu Applications for Telephones. In Marting G. Helander, Thomas K. Landauer, and Prasad V. Prabhu, editors, *Handbook of Human-Computer Interaction (Second Edition)*, pages 1085 – 1102. North-Holland, Amsterdam, second edition edition, 1997.

[329] C. Marshall, C. Nelson, and M.M. Gardiner. Robust tests for equality of variances. In Design guidelines, editor, *Applying Cognitive Psychology to User- Interface Design*. Wiley & Sons Ltd, Chichester, 1987.

[330] Benoundefinedt Martin and Poika Isokoski. Trackmouse Trackball in Pie Menu Use: Data on Accuracy. In *Proceedings of the 19th Conference on l'Interaction Homme-Machine*, IHM '07, pages 127–130, New York, NY, USA, 2007. Association for Computing Machinery. event-place: Paris, France.

[331] Justin Matejka, Tovi Grossman, and George Fitzmaurice. Patina: Dynamic heatmaps for visualizing application usage. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '13, pages 3227–3236, New York, NY, USA, 2013. ACM.

[332] S. Matsui and S. Yamada. Performance evaluation of a genetic algorithm for optimizing hierarchical menus. In *2009 IEEE Congress on Evolutionary Computation*, pages 947–954, May 2009.

[333] Shouichi Matsui and Seiji Yamada. Genetic Algorithm Can Optimize Hierarchical Menus. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, pages 1385–1388, New York, NY, USA, 2008. Association for Computing Machinery. event-place: Florence, Italy.

[334] Shouichi Matsui and Seiji Yamada. Optimizing Hierarchical Menus by Genetic Algorithm and Simulated Annealing. In *Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation*, GECCO '08, pages 1587–1594, New York, NY, USA, 2008. ACM. event-place: Atlanta, GA, USA.

[335] Fabrice Matulic, Daniel Vogel, and Raimund Dachselt. Hand Contact Shape Recognition for Posture-Based Tabletop Widgets and Interaction. In *Proceedings of the 2017 ACM International Conference on Interactive Surfaces and Spaces*, ISS '17, pages 3–11, New York, NY, USA, 2017. Association for Computing Machinery. event-place: Brighton, United Kingdom.

[336] Dan Mauney, Jonathan Howarth, Andrew Wirtanen, and Miranda Capra. Cultural Similarities and Differences in User-defined Gestures for Touchscreen User Interfaces. CHI EA '10, pages 4015–4020, New York, NY, USA, 2010. ACM.

[337] Sébastien Maury, Sylvie Athènes, and Stéphane Chatty. Rhythmic menus: toward interaction based on rhythm. In *CHI Extended Abstracts*, pages 254–255. ACM, 1999.

[338] James E. McDonald, Tom Dayton, and Deborah R. McDonald. Adapting menu layout to tasks. *International Journal of Man-Machine Studies*, 28(4):417 – 435, 1988.

[339] Y. K. Meena, H. Cecotti, K. Wong-Lin, A. Dutta, and G. Prasad. Toward Optimization of Gaze-Controlled Human–Computer Interaction: Application to Hindi Virtual Keyboard for Stroke Patients. *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, 26(4):911–922, April 2018.

[340] Daniel Mendes, Daniel Medeiros, Maurício Sousa, Eduardo Cordeiro, Alfredo Ferreira, and Joaquim A. Jorge. Design and evaluation of a novel out-of-reach selection technique for VR using iterative refinement. *Computers & Graphics*, 67:95 – 102, 2017.

[341] Amilcar Meneses Viveros, Erika Hernández Rubio, and Dario Emmanuel Vázquez Ceballos. Equivalence of Navigation Widgets for Mobile Platforms. In Aaron Marcus, editor, *Design, User Experience, and Usability. User Experience Design for Diverse Interaction Platforms and Environments*, Lecture Notes in Computer Science, pages 269–278, Cham, 2014. Springer International Publishing.

[342] Oussama Metatla, Fiore Martin, Tony Stockman, and Nick Bryan-Kinns. Non-Visual Menu Navigation: The Effect of an Audio-Tactile Display. In *Proceedings of the 28th International BCS Human Computer Interaction Conference on HCI 2014 - Sand, Sea and Sky - Holiday HCI*, BCS-HCI '14, pages 213–217, Swindon, GBR, 2014. BCS. event-place: Southport, UK.

[343] G. A. Miller. The magical number seven plus or minus two: some limits on our capacity for processing information. *Psychological review*, 63 2:81–97, 1956.

[344] Robert Howard Miller, Sheila Renee Crosby, and Robert Joseph Logan. *Hierarchical menu graphical user interface*. Google Patents, December 2003.

[345] Mark Mine, Arun Yoganandan, and Dane Coffey. Principles, interactions and devices for real-world immersive modeling. *Computers & Graphics*, 48:84 – 98, 2015.

[346] J. Mitchell and B. Shneiderman. Dynamic versus static menus: An exploratory comparison. *SIGCHI Bull.*, 20(4):33–37, April 1989.

[347] Peter Mitchell and Brett Wilkinson. Periphery triggered menus for head mounted menu interface interactions. In *OZCHI*, pages 30–33. ACM, 2016.

[348] Mitsuhiro Matsumoto, Ryozo Kiyohara, Hidenori Fukui, Masayuki Numao, and Satoshi Kurihara. Proposition of the context-aware interface for cellular phone operations. In *2008 5th International Conference on Networked Sensing Systems*, pages 233–233, June 2008.

[349] Hiroki Miyamoto, Daijiro Komaki, Takahiro Hara, Kentaro Shimatani, Tomohiro Mashita, Kiyoshi Kiyokawa, Toshiaki Uemukai, Shojiro Nishio, and Haruo Takemura. A Menu-based Content Search Support System Considering Mobile User's Situations. *Procedia Computer Science*, 5:434 – 441, 2011.

[350] Yusuke Miyazawa, Fuminori Homma, Tomoya Narita, and Tatsushi Nashida. *Facilitating display of a menu and selection of a menu item via a touch screen interface*. Google Patents, August 2014.

[351] Armin Moehrle. *Active path menu navigation system*. Google Patents, May 2007.

[352] Karyn Moffatt and Joanna McGrenere. Exploring Methods to Improve Pen-Based Menu Selection for Younger and Older Adults. *ACM Trans. Access. Comput.*, 2(1), May 2009.

[353] P. Monteiro, H. Coelho, G. Gonçalves, M. Melo, and M. Bessa. Comparison of Radial and Panel Menus in Virtual Reality. *IEEE Access*, 7:116370–116379, 2019.

[354] Tomer Moscovich. Contact Area Interaction with Sliding Widgets. In *Proceedings of the 22nd Annual ACM Symposium on User Interface Software and Technology*, UIST '09, pages 13–22, New York, NY, USA, 2009. Association for Computing Machinery. event-place: Victoria, BC, Canada.

[355] Sue Mosher. 21 - Menus, Toolbars, and the Outlook Bar. In Sue Mosher, editor, *Microsoft Outlook Programming*, pages 453 – 479. Digital Press, Burlington, 2003.

[356] Sue Mosher. 23 - Menus, Toolbars, and the Navigation Pane. In Sue Mosher, editor, *Microsoft Outlook 2007 Programming*, pages 737 – 770. Digital Press, Burlington, 2007.

[357] Ob-orm Muangmoon, Mongkhol Rattanakhum, and Pradorn Sureephong. Game Menu Navigation for the Elderly Using Non-Tactile Gesture Interaction: A Pilot Study. In *Proceedings of the International Convention on Rehabilitation Engineering & Assistive Technology*, i-CREATe 2016, Midview City, SGP, 2016. Singapore Therapeutic, Assistive & Rehabilitative Technologies (START) Centre.

[358] Martin Mundt and Tintu Mathew. Exploring Pie Menus for System Control Tasks in Virtual Reality. In *Mensch & Computer*, pages 509–513. GI / ACM, 2019.

[359] Brad A. Myers. Creating Dynamic Interaction Techniques by Demonstration. In *Proceedings of the SIGCHI/GI Conference on Human Factors in Computing Systems and Graphics Interface*, CHI '87, pages 271–278, New York, NY, USA, 1986. Association for Computing Machinery. event-place: Toronto, Ontario, Canada.

[360] Takeshi Nagami, Yoshikazu Seki, Hidenori Sakai, and Hiroaki Ikeda. Usability Evaluation of 4-Direction Keys for Ladder Menu Operation. In Masaaki Kurosu, editor, *Human-Computer Interaction. Interaction Platforms and Techniques*, Lecture Notes in Computer Science, pages 330–340, Cham, 2016. Springer International Publishing.

[361] Namgyu Kim, G. J. Kim, Chan-Mo Park, Inseok Lee, and S. H. Lim. Multimodal menu presentation and selection in immersive virtual environments. In *Proceedings IEEE Virtual Reality 2000 (Cat. No.00CB37048)*, pages 281–, March 2000.

[362] Mickaël Naud, Paul Richard, and Jean-Louis Ferrier. Augmented Reality Interaction Techniques - Design and Evaluation of the Flip-Flop Menu. In *GRAPP*, pages 345–352. INSTICC Press, 2009.

[363] Naveen Kumar S G, S. Parthasarathy, A. Bahuguna, and J. Bose. Dynamic website specific browser menus. In *2015 IEEE International Advance Computing Conference (IACC)*, pages 1122–1127, June 2015.

[364] Tao Ni, Doug A. Bowman, Chris North, and Ryan P. McMahan. Design and evaluation of freehand menu selection interfaces using tilt and pinch gestures. *International Journal of Human-Computer Studies*, 69(9):551 – 562, 2011.

[365] Jakob Nielsen and Jonathan Levy. Measuring usability: Preference vs. performance. *Commun. ACM*, 37(4):66–75, April 1994.

[366] Nintendo R&D1 and Intelligent Systems. *Super Metroid*. Game [SNES], April 1994. Nintendo, Kyoto, Japan. Played August 2011.

[367] Kent L. Norman. Better Design of Menu Selection Systems Through Cognitive Psychology and Human Factors. *Human Factors*, 50(3):556–559, 2008.

[368] Lauren Norrie and Roderick Murray-Smith. Investigating ui displacements in an adaptive mobile homescreen. *Int. J. Mob. Hum. Comput. Interact.*, 8(3):1–17, July 2016.

[369] Forbes Holten Norris III. *Flexible, dynamic menu-based web-page architecture*. Google Patents, June 2011.

[370] Ian Oakley, Stephen Brewster, and Philip Gray. Solving Multi-Target Haptic Problems in Menu Interaction. In *CHI '01 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '01, pages 357–358, New York, NY, USA, 2001. Association for Computing Machinery. event-place: Seattle, Washington.

[371] Ian Oakley and Junseok Park. A motion-based marking menu system. In *CHI'07 Extended Abstracts on Human Factors in Computing Systems*, pages 2597–2602. ACM, 2007.

[372] Ian Oakley and Junseok Park. Motion marking menus: An eyes-free approach to motion input for handheld devices. *International Journal of Human-Computer Studies*, 67(6):515 – 532, 2009.

[373] Tobias Gregor Oberstein. *Configurable pie menu*. Google Patents, March 2014.

[374] Daniel L. Odell, Richard C. Davis, Andrew Smith, and Paul K. Wright. Toolglasses, Marking Menus, and Hotkeys: A Comparison of One and Two-Handed Command Selection Techniques. In *Graphics Interface*, volume 62 of *ACM International Conference Proceeding Series*, pages 17–24. Canadian Human-Computer Communications Society, 2004.

[375] C. Olaverri-Monreal, C. Draxler, and K. Bengler. Variable menus for the local adaptation of Graphical User Interfaces. In *6th Iberian Conference on Information Systems and Technologies (CISTI 2011)*, pages 1–6, June 2011.

[376] Dan R. Olsen. MIKE: The Menu Interaction Kontrol Environment. *ACM Trans. Graph.*, 5(4):318–344, 1986.

[377] Dan R. Olsen and Travis Nielsen. Laser Pointer Interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '01, pages 17–22, New York, NY, USA, 2001. Association for Computing Machinery. event-place: Seattle, Washington, USA.

[378] Katsuhiko Onishi and Yamato Gomi. User-Centered Tile Menu Selection Technique in Large Scale Display. In *HCI (26)*, volume 617 of *Communications in Computer and Information Science*, pages 371–375. Springer, 2016.

[379] Kenneth R. Paap and Nancy J. Cooke. Chapter 24 - Design of Menus. In Marting G. Helander, Thomas K. Landauer, and Prasad V. Prabhu, editors, *Handbook of Human-Computer Interaction (Second Edition)*, pages 533 – 572. North-Holland, Amsterdam, second edition edition, 1997.

[380] Kenneth R. Paap and Renate J. Roske-Hofstrand. Chapter 10 - Design of Menus. In MARTIN HELANDER, editor, *Handbook of Human-Computer Interaction*, pages 205 – 235. North-Holland, Amsterdam, 1988.

[381] Isaac Paquette, Christopher Kwan, and Margrit Betke. Menu Controller: Making Existing Software More Accessible for People with Motor Impairments. In *Proceedings of the 4th International Conference on PErvasive Technologies Related to Assistive Environments*, PETRA '11, New York, NY, USA, 2011. Association for Computing Machinery. event-place: Heraklion, Crete, Greece.

[382] Raja Parasuraman and Victor Riley. Humans and automation: Use, misuse, disuse, abuse. *Human Factors*, 39(2):230–253, 1997.

[383] J. Park and N. Baek. A Text-Based User Interface Style Toolkit for Low-Tier Embedded Systems. In *2013 International Conference on IT Convergence and Security (ICITCS)*, pages 1–4, December 2013.

[384] J. Park, N. Baek, and H. Lee. Design of a small footprint embedded graphics system. In *The 1st IEEE Global Conference on Consumer Electronics 2012*, pages 187–188, October 2012.

[385] Jungchul Park and Sung H. Han. Complementary menus: Combining adaptable and adaptive approaches for menu interface. *International Journal of Industrial Ergonomics*, 41(3):305 – 316, 2011.

[386] Jungchul Park, Sung H. Han, Yong S. Park, and Youngseok Cho. Adaptable versus adaptive menus on the desktop: Performance and user satisfaction. *International Journal of Industrial Ergonomics*, 37(8):675 – 684, 2007.

[387] Jungchul Park, Sung H. Han, Yong S. Park, and Youngseok Cho. Usability of adaptable and adaptive menus. In Nuray Aykin, editor, *Usability and Internationalization. HCI and Culture*, pages 405–411, Berlin, Heidelberg, 2007. Springer.

[388] Jungchul Park, Sung H. Han, Yong S. Park, and Youngseok Cho. Usability of Adaptable and Adaptive Menus. In David Hutchison, Takeo Kanade, Josef Kittler, Jon M. Kleinberg, Friedemann Mattern, John C. Mitchell, Moni Naor, Oscar Nierstrasz, C. Pandu Rangan, Bernhard Steffen, Madhu Sudan, Demetri Terzopoulos, Doug Tygar, Moshe Y. Vardi, Gerhard Weikum, and Nuray Aykin, editors, *Usability and Internationalization. HCI and Culture*, volume 4559, pages 405–411. Springer Berlin Heidelberg, Berlin, Heidelberg, 2007. Series Title: Lecture Notes in Computer Science.

[389] Stanley R Parkinson, Martin D Hill, Norwood Sisson, and Cynthia Viera. Effects of breadth, depth and number of responses on computer menu search performance. *International Journal of Man-Machine Studies*, 28(6):683–692, 1988.

[390] P. Pecev, B. Markoski, L. Ratgeber, D. Lacmanović, and Z. Ivanković. Making mu-Java more accessible. In *2012 IEEE 13th International Symposium on Computational Intelligence and Informatics (CINTI)*, pages 147–150, November 2012.

[391] Gary Perlman and Leo C. Sherwin. Designing Menu Display Format to Match Input Device Format. *SIGCHI Bull.*, 20(2):78–82, October 1988.

[392] Saverio Perugini, Taylor J. Anderson, and William F. Moroney. A study of out-of-turn interaction in menu-based, IVR, voicemail systems. In *CHI*, pages 961–970. ACM, 2007.

[393] Matthew Peterson. 20 - Menus. In Matthew Peterson, editor, *Interactive QuickTime*, QuickTime Developer Series, pages 249 – 254. Morgan Kaufmann, San Francisco, 2004.

[394] Sebastian Pick, Andrew S. Puika, and Torsten W. Kuhlen. Comparison of a speech-based and a pie-menu-based interaction metaphor for application control. In *VR*, pages 381–382. IEEE Computer Society, 2017.

[395] Martin Pielot, Anastasia Kazakova, Tobias Hesselmann, Wilko Heuten, and Susanne Boll. Pocketmenu: Non-visual menus for touch screen devices. In *Proceedings of the 14th International Conference on Human-computer Interaction with Mobile Devices and Services*, MobileHCI '12, pages 327–330, New York, NY, USA, 2012. ACM.

[396] Antoine Ponsard, Kamyar Ardekani, Kailun Zhang, Frederic Ren, Matei Negulescu, and Joanna McGrenere. Twist and pulse: Ephemeral adaptation to improve icon selection on smartphones. In *Proceedings of the 41st Graphics Interface Conference*, GI '15, pages 219–222, Toronto, Ont., Canada, Canada, 2015. Canadian Information Processing Society.

[397] D. A. Prosvirin and V. P. Kharchenko. Developing of WIMP interfaces on critical platforms for automatic flight control system of UAV. In *2017 IEEE 4th International Conference Actual Problems of Unmanned Aerial Vehicles Developments (APUAVD)*, pages 112–115, October 2017.

[398] Lumpapun Punchoojit and Nutttanont Hongwarittorrn. Age Differences in Menu Item Selection for Smartphone: The Effects of Icon Background Colors and Icon Symbols. In *Proceedings of the 5th International ACM In-Cooperation HCI and UX Conference*, CHIuXiD'19, pages 55–64, New York, NY, USA, 2019. Association for Computing Machinery. event-place: Jakarta, Surabaya, Bali, Indonesia.

[399] Dmitry Pyryeskin, Mark Hancock, and Jesse Hoey. Comparing Elicited Gestures to Designer-created Gestures for Selection Above a Multitouch Surface. In *Proceedings of the 2012 ACM International Conference on Interactive Tabletops and Surfaces*, ITS '12, pages 1–10, New York, NY, USA, 2012. ACM.

[400] Whitney Quesenbery. What does usability mean: Looking beyond 'ease of use'. 2001.

[401] Philip Quinn and Andy Cockburn. The Effects of Menu Parallelism on Visual Search and Selection. In *Proceedings of the Ninth Conference on Australasian User Interface - Volume 76*, AUIC '08, pages 79–84, AUS, 2008. Australian Computer Society, Inc. event-place: Wollongong, Australia.

[402] Budiu Raluca. Expandable Menus: Pull-Down, Square, or Pie? (https://www.nngroup.com/articles/expandable-menus/). may 2016.

[403] Gonzalo Ramos and Ravin Balakrishnan. Fluid Interaction Techniques for the Control and Annotation of Digital Video. In *Proceedings of the 16th Annual ACM Symposium on User Interface Software and Technology*, UIST '03, pages 105–114, New York, NY, USA, 2003. Association for Computing Machinery. event-place: Vancouver, Canada.

[404] Gonzalo Ramos, Matthew Boulos, and Ravin Balakrishnan. Pressure Widgets. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '04, pages 487–494, New York, NY, USA, 2004. Association for Computing Machinery. event-place: Vienna, Austria.

[405] Gonzalo Ramos, Andy Cockburn, Ravin Balakrishnan, and Michel Beaudouin-Lafon. Pointing lenses: facilitating stylus input through visual-and motor-space magnification. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 757–766, 2007.

[406] Dimitrios Raptis, Nikolaos Tselios, Jesper Kjeldskov, and Mikael B. Skov. Does size matter?: Investigating the impact of mobile phone screen size on users' perceived usability, effectiveness and efficiency. In *Proceedings of the 15th International Conference on Human-computer Interaction with Mobile Devices and Services*, MobileHCI '13, pages 127–136, New York, NY, USA, 2013. ACM.

[407] Andrew Ray and Doug A. Bowman. Towards a System for Reusable 3D Interaction Techniques. In *Proceedings of the 2007 ACM Symposium on Virtual Reality Software and Technology*, VRST '07, pages 187–190, New York, NY, USA, 2007. Association for Computing Machinery. event-place: Newport Beach, California.

[408] A. Read, A. Tarrell, and A. Fruhling. Exploring User Preference for the Dashboard Menu Design. In *2009 42nd Hawaii International Conference on System Sciences*, pages 1–10, January 2009.

[409] S. M. Redwan. Instant-access start menu for an imaginary WIMP based future operating system. In *2007 10th international conference on computer and information technology*, pages 1–5, December 2007.

[410] Dennis Reiter and Andreas Butz. Design and Implementation of a Widget Set for Steerable Projector-Camera Units. In *Proceedings of the 4th IEEE/ACM International Symposium on Mixed and Augmented Reality*, ISMAR '05, pages 216–217, USA, 2005. IEEE Computer Society.

[411] G. Ren and E. O'Neill. 3D Marking menu selection with freehand gestures. In *2012 IEEE Symposium on 3D User Interfaces (3DUI)*, pages 61–68, March 2012.

[412] X. Ren, J. Yin, T. Oya, and Y. Liu. Enhancing Pie-Menu Selection with Pen Pressure. In *2008 3rd International Conference on Innovative Computing Information and Control*, pages 364–364, June 2008.

[413] C. Ressel, J. Ziegler, and E. Naroska. An approach towards personalized user interfaces for ambient intelligent home environments. In *2006 2nd IET International Conference on Intelligent Environments - IE 06*, volume 1, pages 247–255, July 2006.

[414] George G Robertson and Jock D Mackinlay. The document lens. In *Proceedings of the 6th annual ACM symposium on User interface software and technology*, pages 101–108, 1993.

[415] Teresita de Jesús Álvarez Robles, Francisco Javier Álvarez Rodríguez, Edgard Benítez-Guerrero, and Cristian Rusu. Adapting card sorting for blind people: Evaluation of the interaction design in TalkBack. *Computer Standards & Interfaces*, 66:103356, 2019.

[416] Davide Rocchesso, Francesco Saverio Cannizzaro, Giovanni Capizzi, and Francesco Landolina. Accessing and selecting menu items by in-air touch. In *CHItaly*, pages 20:1–20:9. ACM, 2019.

[417] Michael Rohs, Albrecht Schmidt, Daniel Ashbrook, and Enrico Rukzio, editors. *15th International Conference on Human-Computer Interaction with Mobile Devices and Services, MobileHCI '13, Munich, Germany, August 27 - 30, 2013*. ACM, 2013.

[418] Casey Rothschild. Screening with convex menus and optimal flow taxation. *Journal of Public Economics*, 178:104052, 2019.

[419] Anne Roudaut, Gilles Bailly, Eric Lecolinet, and Laurence Nigay. Leaf menus: Linear menus with stroke shortcuts for small handheld devices. In Tom Gross, Jan Gulliksen, Paula Kotzé, Lars Oestreicher, Philippe Palanque, Raquel Oliveira Prates, and Marco Winckler, editors, *Human-Computer Interaction – INTERACT 2009*, pages 616–619, Berlin, Heidelberg, 2009. Springer.

[420] T. N. Ryan, P. McMahan, and D. A. Bowman. Tech-note: rapMenu: Remote Menu Selection Using Freehand Gestural Input. In *2008 IEEE Symposium on 3D User Interfaces*, pages 55–58, March 2008.

[421] M. A. Abdel Salam, A. E. Keshk, N. A. Ismail, and H. M. Nassar. Automated testing of java menu-based GUIs using XML visual editor. In *2007 International Conference on Computer Engineering Systems*, pages 313–318, November 2007.

[422] S. K. Salman, D. D. Aklil, and R. Stirling. Investigation into the enhancement of substation automation systems using visual programming technique. In *2002 Fifth International Conference on Power System Management and Control Conf. Publ. No. 488)*, pages 439–444, April 2002.

[423] Krystian Samp and Stefan Decker. Supporting Menu Design with Radial Layouts. In *Proceedings of the International Conference on Advanced Visual Interfaces*, AVI '10, pages 155–162, New York, NY, USA, 2010. Association for Computing Machinery. event-place: Roma, Italy.

[424] Krystian Samp and Stefan Decker. Visual search in radial menus. In Pedro Campos, Nicholas Graham, Joaquim Jorge, Nuno Nunes, Philippe Palanque, and Marco Winckler, editors, *Human-Computer Interaction – INTERACT 2011*, pages 248–255, Berlin, Heidelberg, 2011. Springer.

[425] Manojit Sarkar and Marc H Brown. Graphical fisheye views of graphs. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 83–91, 1992.

[426] Manojit Sarkar and Marc H Brown. Graphical fisheye views. *Communications of the ACM*, 37(12):73–83, 1994.

[427] Dietmar Saupe. Fractals. In *Encyclopedia of Computer Science*, pages 725–732. John Wiley and Sons Ltd., GBR, 2003.

[428] Joey Scarr, Andy Cockburn, Carl Gutwin, and Andrea Bunt. Improving Command Selection with CommandMaps. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 257–266, New York, NY, USA, 2012. ACM. event-place: Austin, Texas, USA.

[429] Sabine Schröder and Martina Ziefle. Making a Completely Icon-Based Menu in Mobile Devices to Become True: A User-Centered Design Approach for Its Development. In *Proceedings of the 10th International Conference on Human Computer Interaction with Mobile Devices and Services*, MobileHCI '08, pages 137–146, New York, NY, USA, 2008. Association for Computing Machinery. event-place: Amsterdam, The Netherlands.

[430] E Eugene Schultz Jr and Patrick S Curran. Menu structure and ordering of menu selection: independent or interactive effects? *ACM SIGCHI Bulletin*, 18(2):69–71, 1986.

[431] M. Schwartz and Association of American University Presses Task Force on Bias-Free Language. *Guidelines for bias-free writing*. Indiana University Press, 1995.

[432] Andrew Sears and Ben Shneiderman. Split Menus: Effectively Using Selection Frequency to Organize Menus. *ACM Trans. Comput.-Hum. Interact.*, 1(1):27–51, March 1994.

[433] Edwin J Selker. *Pie menu graphical user interface.* Google Patents, April 2003.

[434] Meg McGinity Shannon. This Menu Has Changed. *Commun. ACM*, 51(5):19–21, May 2008.

[435] C. Sheng, L. Jiang, B. Tang, and X. Tang. A novel menu interaction method using head-mounted display for smartphone-based virtual reality. In *2017 Progress In Electromagnetics Research Symposium - Spring (PIERS)*, pages 2384–2388, May 2017.

[436] Shijie Cai, Xiaoyan Liu, and Xining Li. The design and implementation of DUIDS based on multi-tree structured user-interface model. In *Proceedings of 1996 Canadian Conference on Electrical and Computer Engineering*, volume 1, pages 306–309 vol.1, May 1996.

[437] Choonsung Shin, Jin-Hyuk Hong, and Anind K. Dey. Understanding and prediction of mobile application usage for smart phones. In *Proceedings of the 2012 ACM Conference on Ubiquitous Computing*, UbiComp '12, pages 173–182, New York, NY, USA, 2012. ACM.

[438] Alireza Sahami Shirazi, Christian Winkler, and Albrecht Schmidt. Flashlight Interaction: A Study on Mobile Phone Interaction Techniques with Large Displays. In *Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services*, MobileHCI '09, New York, NY, USA, 2009. Association for Computing Machinery. event-place: Bonn, Germany.

[439] Ben Shneiderman. Designing menu selection systems. *Journal of the American Society for information science*, 37(2):57–70, 1986.

[440] Shouichi Matsui and Seiji Yamada. A genetic algorithm for optimizing hierarchical menus. In *2008 IEEE Congress on Evolutionary Computation (IEEE World Congress on Computational Intelligence)*, pages 2851–2858, June 2008.

[441] Peretz Shoval. Functional design of a menu-tree interface within structured system development. *International Journal of Man-Machine Studies*, 33(5):537 – 556, 1990.

[442] Abhishek Shrivastava and Anirudha Joshi. Effects of Visuals, Menu Depths, and Menu Positions on IVR Usage by Non-Tech Savvy Users. In *Proceedings of the India HCI 2014 Conference on Human Computer Interaction*, IndiaHCI '14, pages 35–44, New York, NY, USA, 2014. Association for Computing Machinery. event-place: New Delhi, India.

[443] Ludwig Sidenmark and Anders Lundström. Gaze Behaviour on Interacted Objects during Hand Interaction in Virtual Reality for Eye Tracking Calibration. In *Proceedings of the 11th ACM Symposium on Eye Tracking Research & Applications*, ETRA '19, New York, NY, USA, 2019. Association for Computing Machinery. event-place: Denver, Colorado.

[444] John Sigle and John E. Howland. Structured Development of Menu-Driven Application Systems. In *Proceedings of the International Conference on APL: Part 1*, APL '79, pages 188–195, New York, NY, USA, 1979. Association for Computing Machinery. event-place: New York, New York, USA.

[445] Anita Sinner, Rita L Irwin, and Jeff Adams. *Provoking the field: International perspectives on visual arts PhDs in education*. Intellect Books, 2019.

[446] Norwood Sisson, Stanley R. Parkinson, and Kathleen Snowberry. Considerations of menu structure and communication rate for the design of computer menu displays. *International Journal of Man-Machine Studies*, 25(5):479 – 489, 1986.

[447] Ross T. Smith, Bruce H. Thomas, and Wayne Piekarski. Digital Foam Interaction Techniques for 3D Modeling. In *Proceedings of the 2008 ACM Symposium on Virtual Reality Software and Technology*, VRST '08, pages 61–68, New York, NY, USA, 2008. Association for Computing Machinery. event-place: Bordeaux, France.

[448] Kathleen Snowberry, Stanley Parkinson, and Norwood Sisson. Effects of help fields on navigating through hierarchical menu structures. *International Journal of Man-Machine Studies*, 22(4):479–491, 1985.

[449] Jaka Sodnik, Grega Jakus, and Sašo Tomažič. Multiple spatial sounds in hierarchical menu navigation for visually impaired computer users. *International Journal of Human-Computer Studies*, 69(1):100 – 112, 2011.

[450] Jun-il Sohn, Dong-Yoon Kim, Won-chul Bang, Eun-Seok Choi, and Sung-jung Cho. *Apparatus and method for controlling speed of moving between menu list items*. Google Patents, July 2010.

[451] Benjamin L. Somberg. A comparison of rule-based and positionally constant arrangements of computer menu items. In *Proceedings of the SIGCHI/GI Conference on Human Factors in Computing Systems and Graphics Interface*, CHI '87, pages 255–260, New York, NY, USA, 1987. ACM.

[452] Park Jong Soon and Rohae Myung. Evaluation of menu structure based on signal detection theory. In *SIGDOC*, pages 81–86. ACM, 2008.

[453] Marco Speicher, Jan Ehrlich, Vito Gentile, Donald Degraen, Salvatore Sorce, and Antonio Krüger. Pseudo-haptic Controls for Mid-air Finger-based Menu Interaction. In *CHI Extended Abstracts*. ACM, 2019.

[454] Edward Srenger, Daniel Rokusek, and Kevin Weirich. *Adaptive menu for a user interface*. Google Patents, September 2006.

[455] Robert St Amant, Thomas E Horton, and Frank E Ritter. Model-based evaluation of cell phone menu interaction. In *Proceedings of the SIGCHI conference on Human factors in computing systems*, pages 343–350. ACM, 2004.

[456] Thomas Jan Stovicek, Yoojin Hong, and Donald James Lindsay. *Graphical context short menu*. Google Patents, August 2011.

[457] R. Sugai and M. Kurisu. Position determination of a popup menu on operation screens of a teleoperation system using a low cost head tracker. In *2015 15th International Conference on Control, Automation and Systems (ICCAS)*, pages 875–880, October 2015.

[458] Bernhard Suhm, Barbara Freeman, and David Getty. Curing the Menu Blues in Touch-Tone Voice Interfaces. In *CHI '01 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '01, pages 131–132, New York, NY, USA, 2001. Association for Computing Machinery. event-place: Seattle, Washington.

[459] M. Sun and X. Ren. An Evaluation of Thumb Interface for Menu Selection on Mobile Equipment. In *2009 Fourth International Conference on Innovative Computing, Information and Control (ICICIC)*, pages 805–809, December 2009.

[460] Y. Sun, Y. Zhang, and G. Xue. SmartWheelTag: Flexible and Battery-less User Interface for Drivers. In *2018 IEEE Vehicular Networking Conference (VNC)*, pages 1–2, December 2018.

[461] Veikko Surakka, Marko Illi, and Poika Isokoski. Gazing and Frowning as a New Human–Computer Interaction Technique. *ACM Trans. Appl. Percept.*, 1(1):40–56, July 2004.

[462] A. G. Sutcliffe. Interface Components and Interaction Styles. In A. G. Sutcliffe, editor, *Human-Computer Interface Design*, Macmillan Computer Science Series, pages 98–132. Macmillan Education UK, London, 1995.

[463] Michael Helligso Svinth. *Menu searching of a hierarchical menu structure*. Google Patents, March 2016.

[464] Erum Tanvir, Andrea Bunt, Andy Cockburn, and Pourang Irani. Improving cascading menu selections with adaptive activation areas. *International Journal of Human-Computer Studies*, 69(11):769 – 785, 2011.

[465] Erum Tanvir, Jonathan Cullen, Pourang Irani, and Andy Cockburn. AAMU: Adaptive Activation Area Menus for Improving Selection in Cascading Pull-down Menus. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, pages 1381–1384, New York, NY, USA, 2008. ACM. event-place: Florence, Italy.

[466] H. Tate, S. Goto, and S. Takeuchi. A proposal on content information management for multimedia service navigation system. In *Proceedings of NOMS '96 - IEEE Network Operations and Management Symposium*, volume 2, pages 466–475 vol.2, April 1996.

[467] Felix Thalmann, Ulrich von Zadow, Marcel Heckel, and Raimund Dachselt. X-O Arch Menu: Combining Precise Positioning with Efficient Menu Selection on Touch Devices. In *Proceedings of the Ninth ACM International Conference on Interactive Tabletops and Surfaces*, ITS '14, pages 317–322, New York, NY, USA, 2014. Association for Computing Machinery. event-place: Dresden, Germany.

[468] Feng Tian, Lishuang Xu, Hongan Wang, Xiaolong Zhang, Yuanyuan Liu, Vidya Setlur, and Guozhong Dai. Tilt Menu: Using the 3D Orientation Information of Pen Devices to Extend the Selection Capability of Pen-based User Interfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '08, pages 1371–1380, New York, NY, USA, 2008. ACM. event-place: Florence, Italy.

[469] Geoffrey Tien and M. Stella Atkins. Improving Hands-Free Menu Selection Using Eyegaze Glances and Fixations. In *Proceedings of the 2008 Symposium on Eye Tracking Research & Applications*, ETRA '08, pages 47–50, New York, NY, USA, 2008. Association for Computing Machinery. event-place: Savannah, Georgia.

[470] Tong Lin (tlin2@cs.umbc.edu). Turtle interpretation. https://www.csee.umbc.edu/~ebert/693/TLin/node5.html, 1996. [Online: accessed 17-March-2020].

[471] Christian Tominski, Stefan Gladisch, Ulrike Kister, Raimund Dachselt, and Heidrun Schumann. Interactive lenses for visualization: An extended survey. In *Computer Graphics Forum*, volume 36, pages 173–200. Wiley Online Library, 2017.

[472] Ayumi Tomita, Keisuke Kambara, and Itiro Siio. Slant Menu: Novel GUI Widget with Ergonomic Design. In *CHI '12 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '12, pages 2051–2056, New York, NY, USA, 2012. ACM. event-place: Austin, Texas, USA.

[473] Scott V Totman, Jeffrey David Kimball, Steve Clark, James Andrew Canfield, Elisa Nader, Thomas Brent Canfield, and Keith R Deaven. *Visually distinguishing menu items*. Google Patents, November 2008.

[474] Luigi Troiano, Cosimo Birtolo, and Roberto Armenise. Searching optimal menu layouts by linear genetic programming. *Journal of Ambient Intelligence and Humanized Computing*, 7(2):239–256, April 2016.

[475] Luigi Troiano, Cosimo Birtolo, Roberto Armenise, and Gennaro Cirillo. Optimization of Menu Layouts by Means of Genetic Algorithms. In Jano van Hemert and Carlos Cotta, editors, *Evolutionary Computation in Combinatorial Optimization*, Lecture Notes in Computer Science, pages 242–253, Berlin, Heidelberg, 2008. Springer.

[476] Theophanis Tsandilas and m. c. schraefel. An empirical assessment of adaptation techniques. In *CHI '05 extended abstracts on Human factors in computing systems - CHI '05*, page 2009, Portland, OR, USA, 2005. ACM Press.

[477] Theophanis Tsandilas and m. c. schraefel. Bubbling Menus: A Selective Mechanism for Accessing Hierarchical Drop-down Menus. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '07, pages 1195–1204, New York, NY, USA, 2007. Association for Computing Machinery. event-place: San Jose, California, USA.

[478] Shin-Ping Tucker. E-commerce standard user interface: an E-menu system. *Industrial Management & Data Systems*, 108(8):1009–1028, 2008.

[479] Thomas S Tullis. Designing a menu-based interface to an operating system. In *ACM SIGCHI Bulletin*, volume 16, pages 79–84. ACM, 1985.

[480] Kimmo Tuomainen, Sanna Suomalainen, and Katja Konkka. *Context dependent auxiliary menu elements*. Google Patents, December 2004.

[481] Md. Sami Uddin, Carl Gutwin, and Benjamin Lafreniere. HandMark Menus: Rapid Command Selection and Large Command Sets on Multi-Touch Displays. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems*, CHI

'16, pages 5836–5848, New York, NY, USA, 2016. ACM. event-place: San Jose, California, USA.

[482] Jacob Ukelson and Johann Makowsky. Formal interactive menu design. *Interacting with Computers*, 4(1):83 – 101, 1992.

[483] Brygg Ullmer, Rajesh Sankaran, Srikanth Jandhyala, Blake Tregre, Cornelius Toole, Karun Kallakuri, Christopher Laan, Matthew Hess, Farid Harhad, Urban Wiggins, and Shining Sun. Tangible menus and interaction trays: core tangibles for common physical/digital activities. In *Tangible and Embedded Interaction*, pages 209–212. ACM, 2008.

[484] Tuomas Vaittinen, Timo-Pekka Viljamaa, and Petri Piippo. Design Issues Related to Pie Menus for 5-way Joysticks. In *Proceedings of the 4th International Conference on Mobile Technology, Applications, and Systems and the 1st International Symposium on Computer Human Interaction in Mobile Technology*, Mobility '07, pages 564–571, New York, NY, USA, 2007. ACM. event-place: Singapore.

[485] Dimitar Valkov, Alexander Giesler, and Klaus H. Hinrichs. Imperceptible Depth Shifts for Touch Interaction with Stereoscopic Objects. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '14, pages 227–236, New York, NY, USA, 2014. Association for Computing Machinery. event-place: Toronto, Ontario, Canada.

[486] Jean Vanderdonckt. Computer-Aided Design of Menu Bar and Pull-Down Menus for Business Oriented Applications. In David Duke and Angel Puerta, editors, *Design, Specification and Verification of Interactive Systems '99*, Eurographics, pages 84–99, Vienna, 1999. Springer.

[487] Jean Vanderdonckt. Computer-aided design of menu bar and pull-down menus for business oriented applications. In Duke and Puerta [156], pages 84–99.

[488] JEAN VANDERDONCKT and DAVID AHLSTRÖM. A systematic literature review of menu selection for any interactive system. *ACM Computing Surveys*, 52(1), 2019.

[489] Jean Vanderdonckt, Sara Bouzit, Gaëlle Calvary, and Denis Chêne. Cloud menus: a circular adaptive menu for small screens. In Shlomo Berkovsky, Yoshinori Hijikata, Jun Rekimoto, Margaret M. Burnett, Mark Billinghurst, and Aaron Quigley, editors, *Proceedings of the 23rd International Conference on Intelligent User Interfaces, IUI 2018, Tokyo, Japan, March 07-11, 2018*, pages 317–328. ACM, 2018.

[490] Jean Vanderdonckt, Sara Bouzit, Gaëlle Calvary, and Denis Chêne. Exploring a design space of graphical adaptive menus: Normal vs. small screens. *ACM Trans. Interact. Intell. Syst.*, 10(1):2:1–2:40, 2020.

[491] Jean Vanderdonckt, Sara Bouzit, Gaelle Calvary, and Denis Chêne. Exploring a design space of graphical adaptive menus: Normal vs. small screens. *ACM Transactions on Interactive Intelligent Systems*, 10:1–40, 07 2019.

[492] Jean Vanderdonckt, Sara Bouzit, Gaëlle Calvary, and Denis Chundefinedne. Cloud Menus: A Circular Adaptive Menu for Small Screens. In *23rd International Conference on Intelligent User Interfaces*, IUI '18, pages 317–328, New York, NY, USA, 2018. Association for Computing Machinery. event-place: Tokyo, Japan.

[493] Jean Vanderdonckt, Sara Bouzit, Gaëlle Calvary, and Denis Chundefinedne. Exploring a Design Space of Graphical Adaptive Menus: Normal vs. Small Screens. *ACM Trans. Interact. Intell. Syst.*, 10(1), July 2019.

[494] Jean Vanderdonckt and Éric Petit. G-menu: A keyword-by-gesture based dynamic menu interface for smartphones. In Masaaki Kurosu, editor, *Human-Computer Interaction. Recognition and Interaction Technologies - Thematic Area, HCI 2019, Held as Part of the 21st HCI International Conference, HCII 2019, Orlando, FL, USA, July 26-31, 2019, Proceedings, Part II*, volume 11567 of *Lecture Notes in Computer Science*, pages 99–114. Springer, 2019.

[495] Jean Vanderdonckt and Éric Petit. G-Menu: A Keyword-by-Gesture Based Dynamic Menu Interface for Smartphones. In Masaaki Kurosu, editor, *Human-Computer Interaction. Recognition and Interaction Technologies*, Lecture Notes in Computer Science, pages 99–114, Cham, 2019. Springer International Publishing.

[496] Jean Vanderdonckt, Mathieu Zen, and Radu-Daniel Vatavu. Ab4web: An on-line a/b tester for comparing user interface design alternatives. *Proc. ACM Hum.-Comput. Interact.*, 3(EICS), June 2019.

[497] Jean Vanderdonckt, Mathieu Zen, and Radu-Daniel Vatavu. Ab4web: An on-line A/B tester for comparing user interface design alternatives. *PACMHCI*, 3(EICS):18:1–18:28, 2019.

[498] A. N. Varnavsky. Research of preferences dependence in hierarchical text menus of user interface from performance cognitive processes. In *2016 Cognitive Sciences, Genomics and Bioinformatics (CSGB)*, pages 1–5, August 2016.

[499] Santiago Villarreal-Narvaez, Jean Vanderdonckt, Radu-Daniel Vatavu, and Jacob O. Wobbrock. A systematic review of gesture elicitation studies: What can we learn from 216 studies? In *Proceedings of the 2020 ACM Designing Interactive Systems Conference*, DIS '20, page 855–872, New York, NY, USA, 2020. Association for Computing Machinery.

[500] John S Wang. *Menu bar editor*. Google Patents, June 1996.

[501] Xiangyu Wang. Design and Evaluation of Intelligent Menu Interface through Cognitive Walkthrough Procedure and Automated Logging for Management Information System. In Weiming Shen, Jianming Yong, Yun Yang, Jean-Paul A. Barthès, and Junzhou Luo, editors, *Computer Supported Cooperative Work in Design IV*, Lecture Notes in Computer Science, pages 408–418, Berlin, Heidelberg, 2008. Springer.

[502] Willian Massami Watanabe and Renata Pontin de Mattos Fortes. Automatic Identification of Drop-down Menu Widgets Using Mutation Observers and Visibility Changes. In *Proceedings of the 31st Annual ACM Symposium on Applied Computing*, SAC '16, pages 766–771, New York, NY, USA, 2016. ACM. event-place: Pisa, Italy.

[503] Keith Weiskamp. 13 - Developing Pop-Up Menus. In Keith Weiskamp, editor, *Advanced Programming with Microsoft Quickc*, pages 477 – 542. Academic Press, 1989.

[504] Shelly Welch and Si-Jung Kim. Determining the effect of menu element size on usability of mobile applications. In Aaron Marcus, editor, *Design, User Experience, and Usability. Web, Mobile, and Product Design*, pages 740–749, Berlin, Heidelberg, 2013. Springer Berlin Heidelberg.

[505] Janet L. Wesson, Akash Singh, and Bradley van Tonder. Can adaptive interfaces improve the usability of mobile applications? In Peter Forbrig, Fabio Paternó, and Annelise Mark Pejtersen, editors, *Human-Computer Interaction*, pages 187–198, Berlin, 2010. Springer.

[506] Aaron Joseph Wheeler, Alejandro Kauffmann, Liang-yu Tom Chi, Hayes Solos Raffle, and Luis Ricardo Prada Gomez. *Graphical menu and interaction therewith through a viewing window*. Google Patents, February 2014.

[507] S. White, D. Feng, and S. Feiner. Poster: Shake menus: Towards activation and placement techniques for prop-based 3D graphical menus. In *2009 IEEE Symposium on 3D User Interfaces*, pages 129–130, March 2009.

[508] Michelle Wiebe, Denise Y. Geiskkovitch, and Andrea Bunt. Exploring user attitudes towards different approaches to command recommendation in feature-rich software. In *Proceedings of the 21st International Conference on Intelligent User Interfaces*, IUI '16, pages 43–47, New York, NY, USA, 2016. ACM.

[509] Marian G. Williams and Mark W. Altom, editors. *Proceeding of the CHI '99 Conference on Human Factors in Computing Systems: The CHI is the Limit, Pittsburgh, PA, USA, May 15-20, 1999*. ACM, 1999.

[510] Graham Wilson, Craig Stewart, and Stephen A. Brewster. Pressure-Based Menu Selection for Mobile Devices. In *Proceedings of the 12th International Conference on Human Computer Interaction with Mobile Devices and Services*, MobileHCI '10, pages 181–190, New York, NY, USA, 2010. Association for Computing Machinery. event-place: Lisbon, Portugal.

[511] Langdon Winner. *The Social Shaping of Technology*, chapter Do artifacts have politics?, pages 28–40. Open University Press, UK, 2nd edition, 1999.

[512] H. Witt and E. M. Kluge. Studying Input Device Performance: An End-User Driven Experiment in wearIT@work. In *First International Conference on Advances in Computer-Human Interaction*, pages 9–15, February 2008.

[513] Ian H Witten, John G Cleary, and Saul Greenberg. On frequency-based menu-splitting algorithms. *International Journal of Man-Machine Studies*, 21(2):135–148, 1984.

[514] Xue Wu, Changxu Wu, Dong Wei, and Yan Xiao. Alternative computer mouse trigger designs in computerized physician order entry (CPOE) system to reduce clinicians' drop-down menu selection errors. *International Journal of Industrial Ergonomics*, 71:14 – 19, 2019.

[515] Xiangyu Wang. Using Cognitive Walkthrough procedure to prototype and evaluate dynamic menu interfaces: A design improvement. In *2008 12th International Conference on Computer Supported Cooperative Work in Design*, pages 76–80, April 2008.

[516] Wenchang Xu, Chun Yu, Jie Liu, and Yuanchun Shi. RegionalSliding: Facilitating small target selection with marking menu for one-handed thumb use on touchscreen-based mobile devices. *Pervasive and Mobile Computing*, 17:63 – 78, 2015.

[517] Pavani Yalla and Bruce N. Walker. Advanced Auditory Menus: Design and Evaluation of Auditory Scroll Bars. In *Proceedings of the 10th International ACM SIGACCESS Conference on Computers and Accessibility*, Assets '08, pages 105–112, New York, NY, USA, 2008. ACM. event-place: Halifax, Nova Scotia, Canada.

[518] Takehiko Yamaguchi, Damien Chamaret, and Paul Richard. The Effect of Haptic Cues on Working Memory in 3D Menu Selection. In Julie A. Jacko, editor, *Human-Computer Interaction. Interaction Techniques and Environments*, Lecture Notes in Computer Science, pages 158–166, Berlin, Heidelberg, 2011. Springer.

[519] Takuto Yanagida, Hidetoshi Nonaka, and Masahito Kurihara. Personalizing Graphical User Interfaces on Flexible Widget Layout. In *Proceedings of the 1st ACM SIGCHI Symposium on Engineering Interactive Computing Systems*, EICS '09, pages 255–264, New York, NY, USA, 2009. Association for Computing Machinery. event-place: Pittsburgh, PA, USA.

[520] Min Yin and Shumin Zhai. The Benefits of Augmenting Telephone Voice Menu Navigation with Visual Browsing and Search. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '06, pages 319–328, New York, NY, USA, 2006. Association for Computing Machinery. event-place: Montréal, Québec, Canada.

[521] Takuto Yoshikawa, Buntarou Shizuki, and Jiro Tanaka. HandyWidgets: Local Widgets Pulled-out from Hands. In *Proceedings of the 2012 ACM International Conference on Interactive Tabletops and Surfaces*, ITS '12, pages 197–200, New York, NY, USA, 2012. ACM. event-place: Cambridge, Massachusetts, USA.

[522] YoungJin Ro, Ho-Jin Choi, Dan Hyung Lee, and In-Young Ko. Dynamic pull-down menu organization in run-time by analyzing menu selection behaviors. In *2008 8th IEEE International Conference on Computer and Information Technology*, pages 730–735, July 2008.

[523] Yaohua Yu and Zhengjie Liu. Improving the Performance and Usability for Visual Menu Interface on Mobile Computers. In *Proceedings of the International Conference on Advanced Visual Interfaces*, AVI '10, pages 369–372, New York, NY, USA, 2010. ACM. event-place: Roma, Italy.

[524] Yaohua Yu and Zhengjie Liu. A User Study of Visual Versus Sonically-enhanced Interfaces for Use While Walking. In *Proceedings of the 18th ACM International Conference on Multimedia*, MM '10, pages 687–690, New York, NY, USA, 2010. ACM. event-place: Firenze, Italy.

[525] Panayiotis Zaphiris, Sri Hastuti Kurniawan, and R. Darin Ellis. Age Related Differences and the Depth vs. Breadth Tradeoff in Hierarchical Online Information Systems. In Noëlle Carbonell and Constantine Stephanidis, editors, *Universal Access Theoretical Perspectives, Practice, and Experience*, Lecture Notes in Computer Science, pages 23–42, Berlin, Heidelberg, 2003. Springer.

[526] Xinda Zeng, Feng Tian, Yingying Jiang, Xiaolong (Luke) Zhang, Guozhong Dai, and Hongan Wang. Thumb Widgets: Apply Thumb-tracking to Enhance Capabilities of Multi-touch on Mobile Devices. In *CHI '13 Extended Abstracts on Human Factors in Computing Systems*, CHI EA '13, pages 1281–1286, New York, NY, USA, 2013. ACM. event-place: Paris, France.

[527] Yuguang Zeng and Jingyuan Zhang. Multiple User Context Menus for Large Displays. In *Proceedings of the 2014 ACM Southeast Regional Conference*, ACM SE '14, pages 44:1–44:4, New York, NY, USA, 2014. ACM. event-place: Kennesaw, Georgia.

[528] X. Zhang, Y. Cao, and X. Mu. The Dynamic Retrieval Tree Menu Based on Dojo. In *2011 International Conference of Information Technology, Computer Engineering and Management Sciences*, volume 1, pages 202–205, September 2011.

[529] X. ZHANG, W. SHAN, Q. XU, B. YANG, and Y. ZHANG. An Ergonomics Study of Menu-Operation on Mobile Phone Interface. In *Workshop on Intelligent Information Technology Application (IITA 2007)*, pages 247–251, December 2007.

[530] Yin Zhang, Haokai Lu, Wei Niu, and James Caverlee. Quality-Aware Neural Complementary Item Recommendation. In *Proceedings of the 12th ACM Conference on Recommender Systems*, RecSys '18, pages 77–85, New York, NY, USA, 2018. Association for Computing Machinery. event-place: Vancouver, British Columbia, Canada.

[531] Shengdong Zhao, Maneesh Agrawala, and Ken Hinckley. Zone and Polygon Menus: Using Relative Position to Increase the Breadth of Multi-Stroke Marking Menus. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '06, pages 1077–1086, New York, NY, USA, 2006. Association for Computing Machinery. event-place: Montréal, Québec, Canada.

[532] Shengdong Zhao and Ravin Balakrishnan. Simple vs. Compound Mark Hierarchical Marking Menus. In *Proceedings of the 17th Annual ACM Symposium on User Interface Software and Technology*, UIST '04, pages 33–42, New York, NY, USA, 2004. ACM. event-place: Santa Fe, NM, USA.

[533] Shengdong Zhao, Pierre Dragicevic, Mark Chignell, Ravin Balakrishnan, and Patrick Baudisch. Earpod: Eyes-free Menu Selection Using Touch Input and Reactive Audio Feedback. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '07, pages 1395–1404, New York, NY, USA, 2007. ACM. event-place: San Jose, California, USA.

[534] Jingjie Zheng, Xiaojun Bi, Kun Li, Yang Li, and Shumin Zhai. M3 Gesture Menu: Design and Experimental Analyses of Marking Menus for Touchscreen Mobile Interaction. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, New York, NY, USA, 2018. Association for Computing Machinery. event-place: Montreal QC, Canada.

[535] Martina Ziefle. Information presentation in small screen devices: The trade-off between visual density and menu foresight. *Applied Ergonomics*, 41(6):719 – 730, 2010.

[536] Martina Ziefle and Susanne Bay. Mental models of a cellular phone menu. Comparing older and younger novice users. In *International Conference on Mobile Human-Computer Interaction*, pages 25–37. Springer, 2004.

[537] J. E. Zucco, B. H. Thomas, K. Grimmer-Somers, and A. Cockburn. A Comparison of Menu Configurations and Pointing Devices for Use with Wearable Computers while Mobile and Stationary. In *2009 International Symposium on Wearable Computers*, pages 63–70, September 2009.

[538] Oleg Špakov and Darius Miniotas. Gaze-Based Selection of Standard-Size Menu Items. In *Proceedings of the 7th International Conference on Multimodal Interfaces*, ICMI '05, pages 124–128, New York, NY, USA, 2005. Association for Computing Machinery. event-place: Torento, Italy.

Appendices

Appendix A

Systematic Literature Review appendix

A.1 Paper Machines Generated Graphics

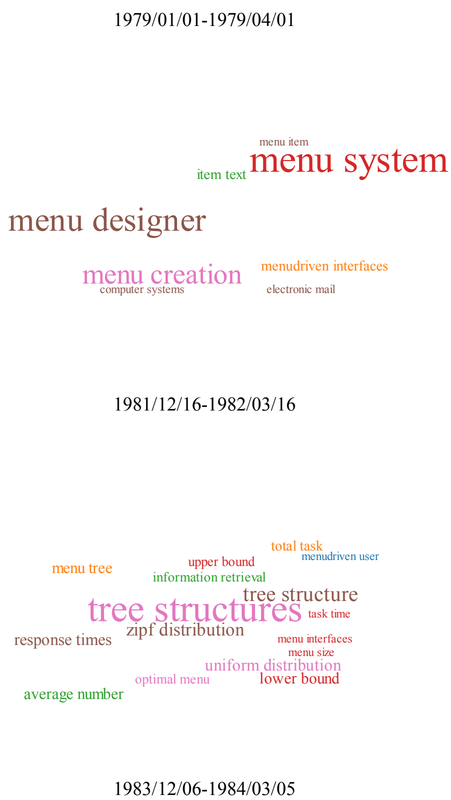


Figure A.1: Chronological Word Cloud Dunning

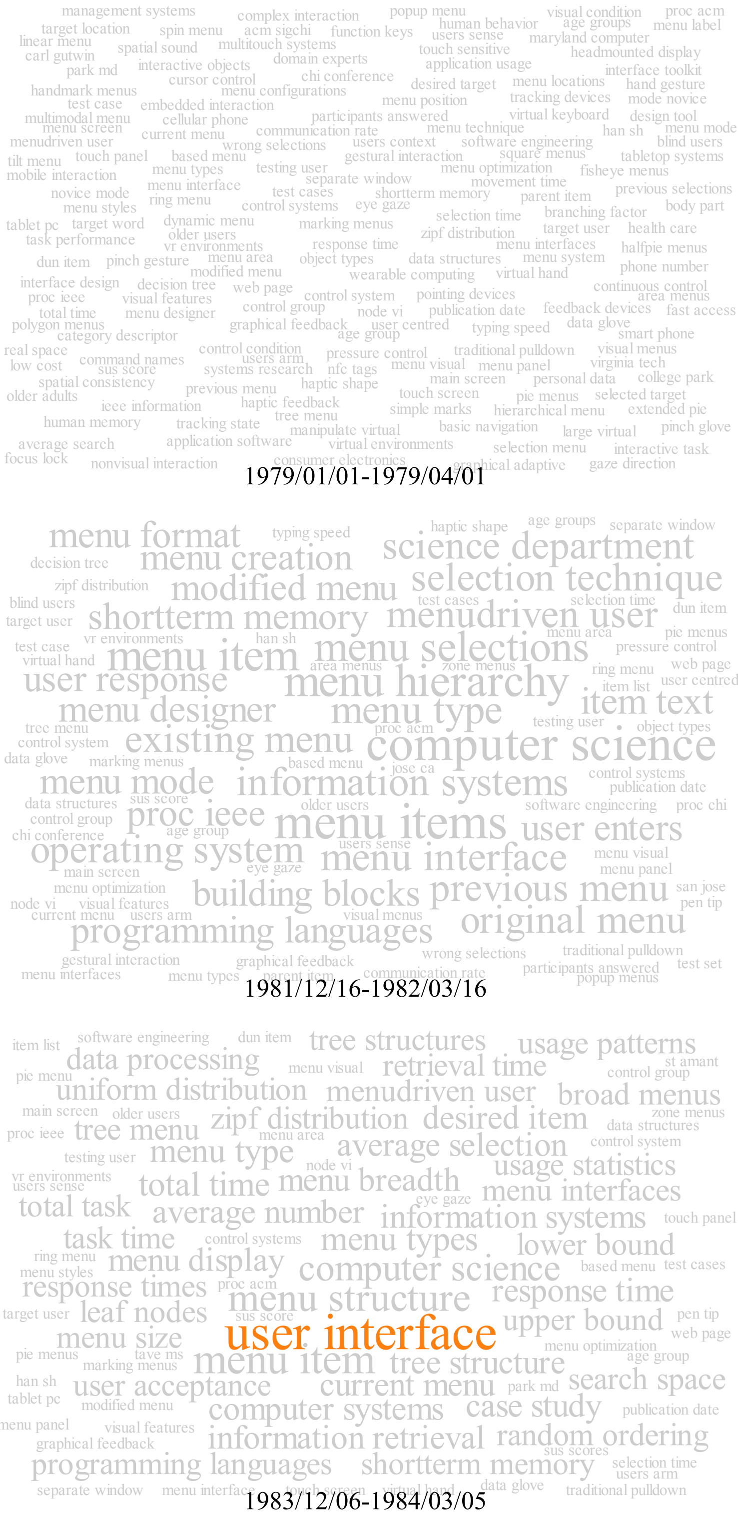
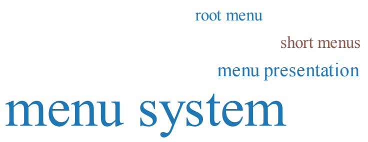
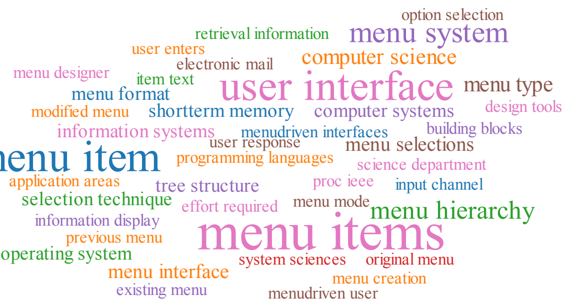


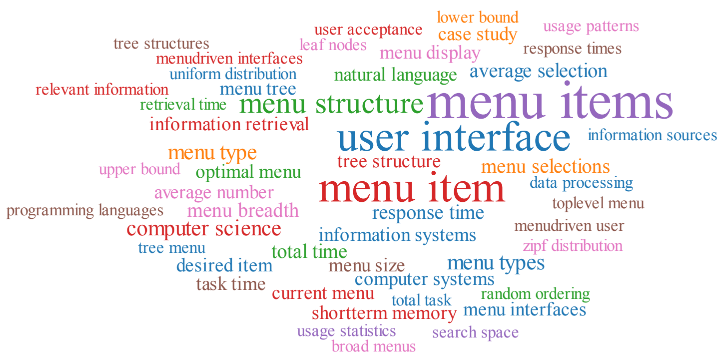
Figure A.2: Chronological Word Cloud Mann Whitney



1979/01/01-1979/04/01

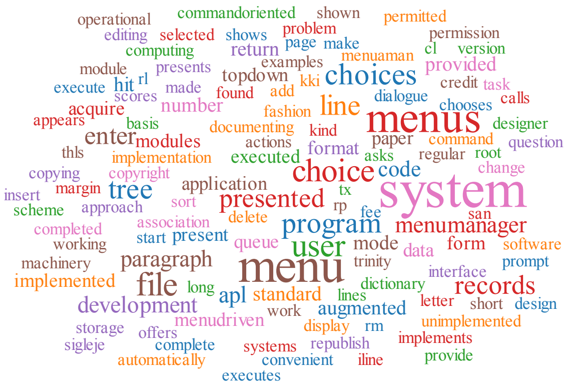


1981/12/16-1982/03/16



1983/12/06-1984/03/05

Figure A.3: Chronological Word Cloud TFIDF



1979/01/01-1979/04/01

1981/12/16-1982/03/16

1983/12/06-1984/03/05

Figure A.4: Chronological Word Cloud

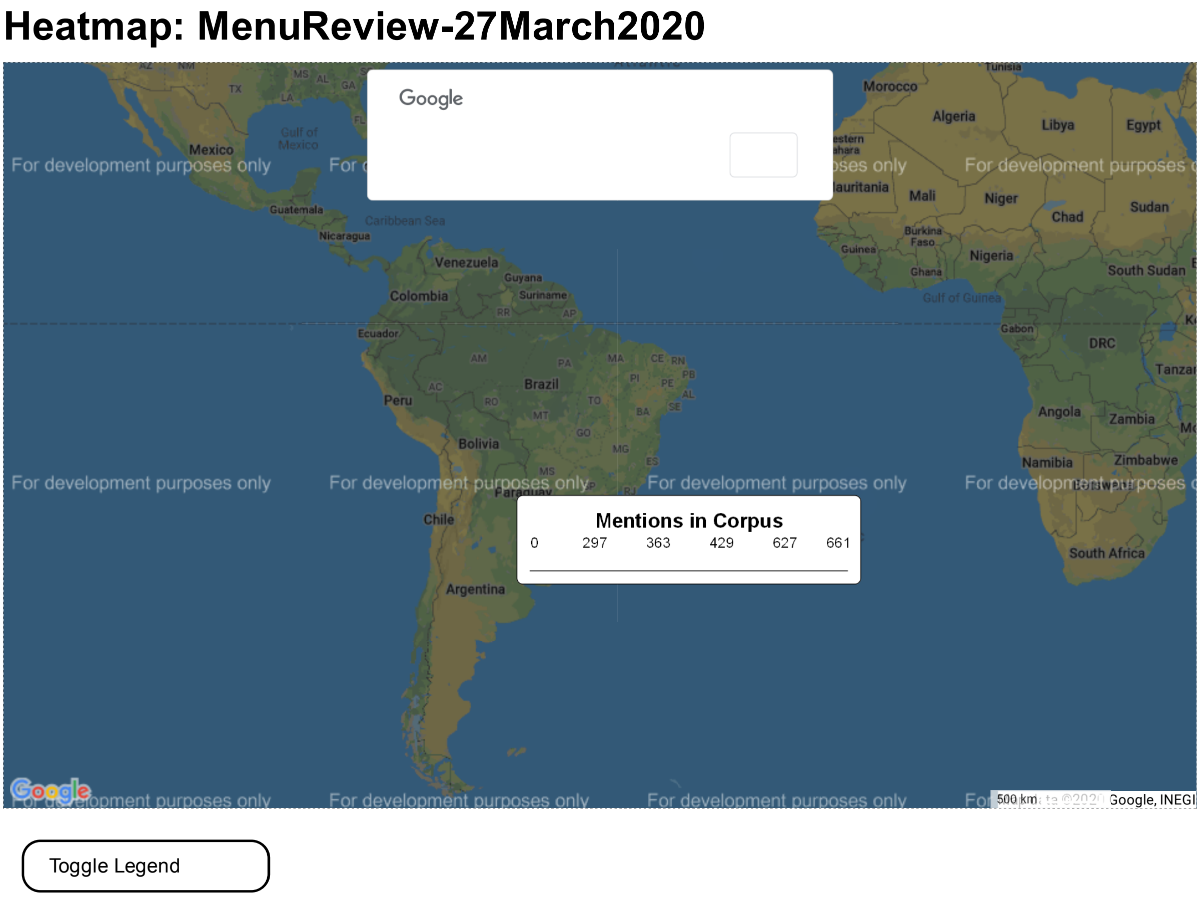


Figure A.5: HeatMap

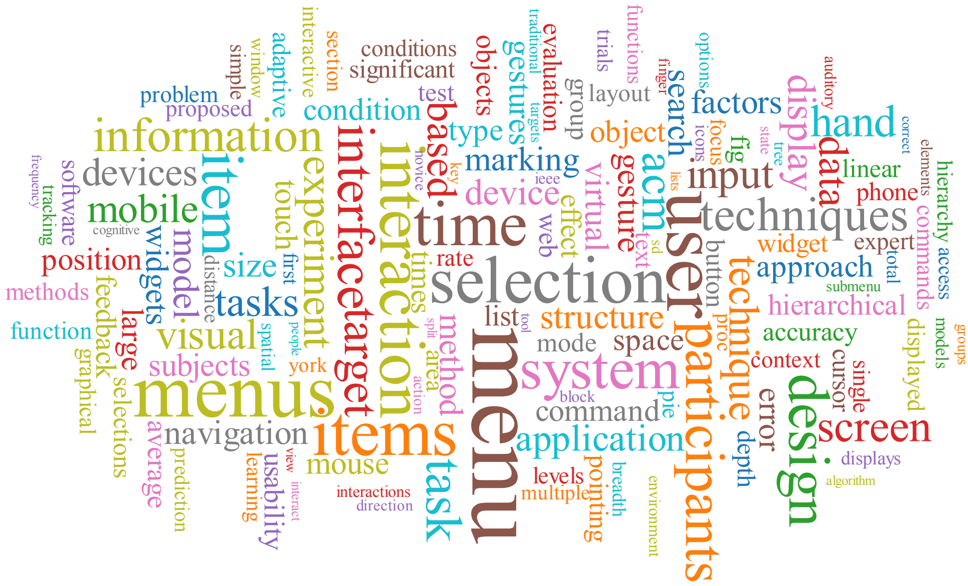


Figure A.6: LargeWordCloud

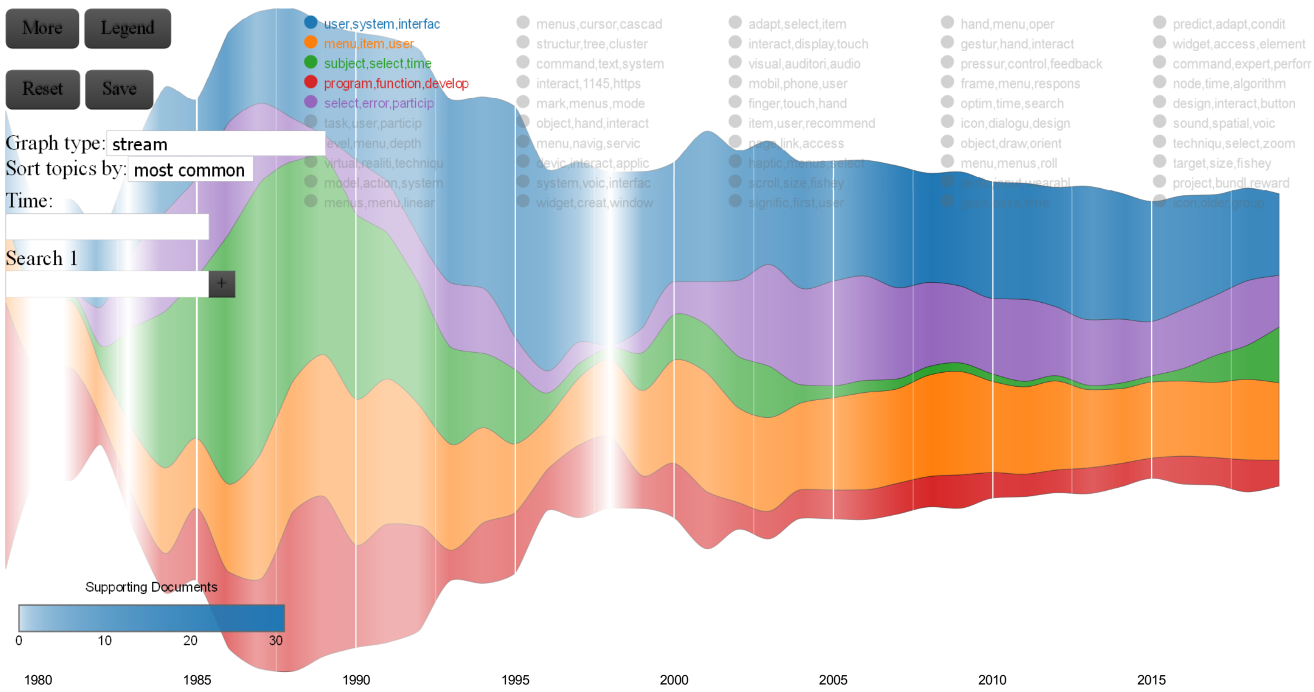


Figure A.7: TopicModelling

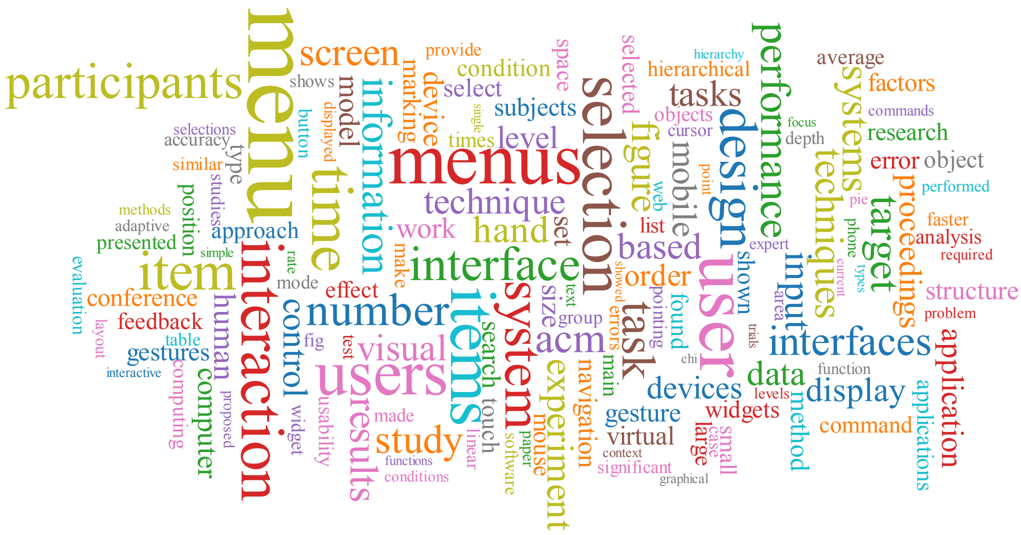


Figure A.8: WordCloud-Global

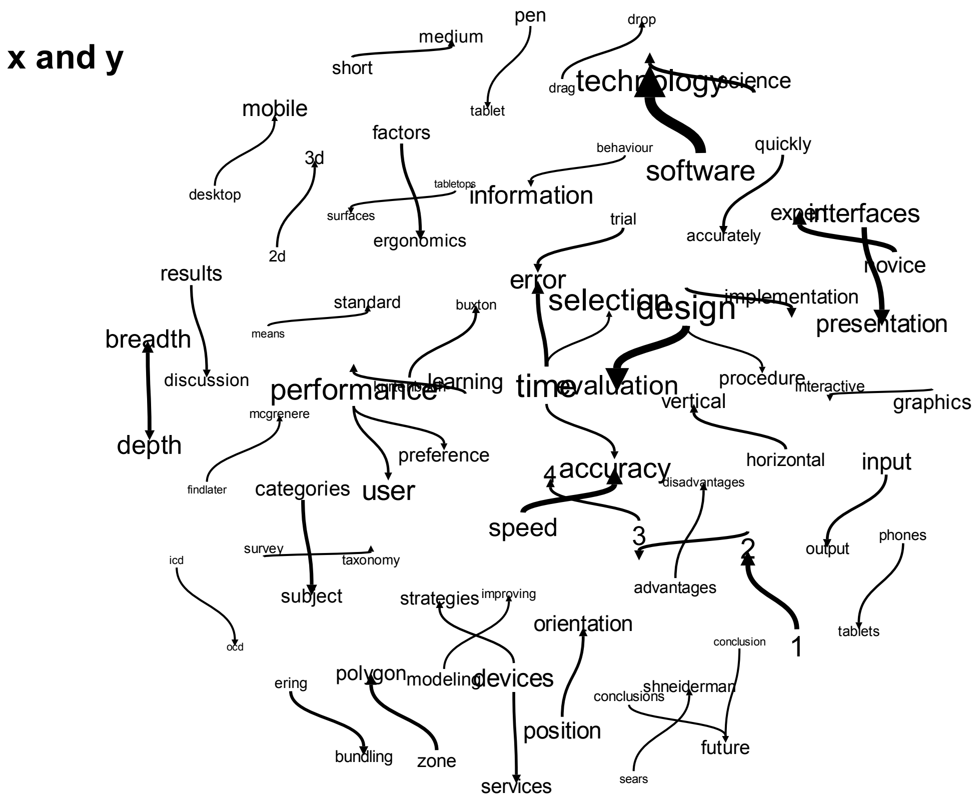


Figure A.9: XandY

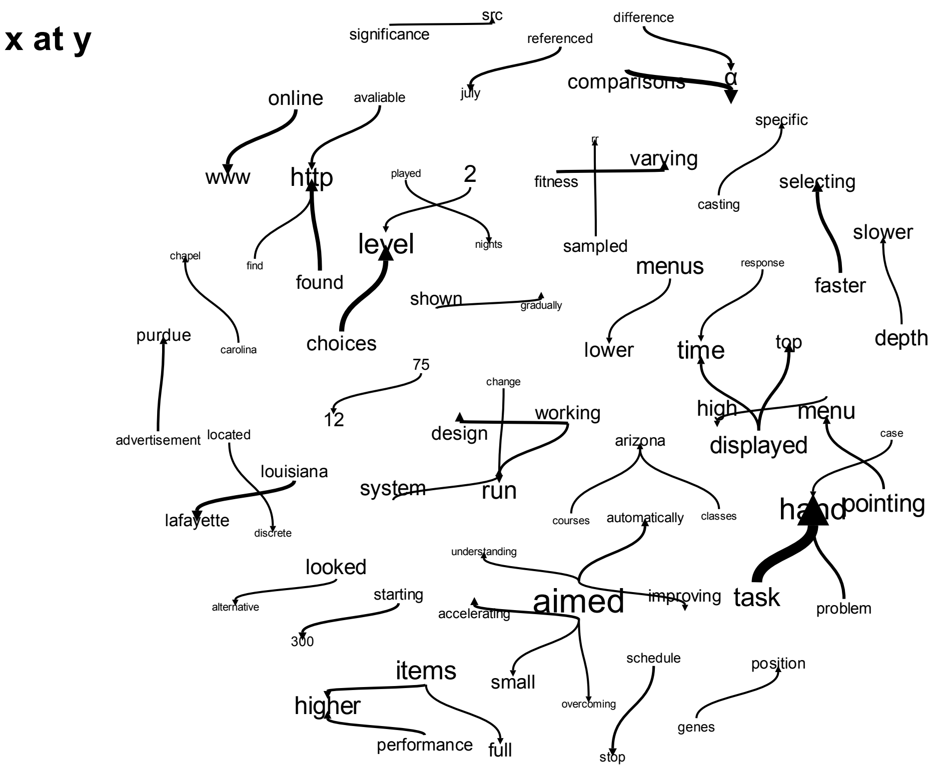


Figure A.10: XatYm

x a y

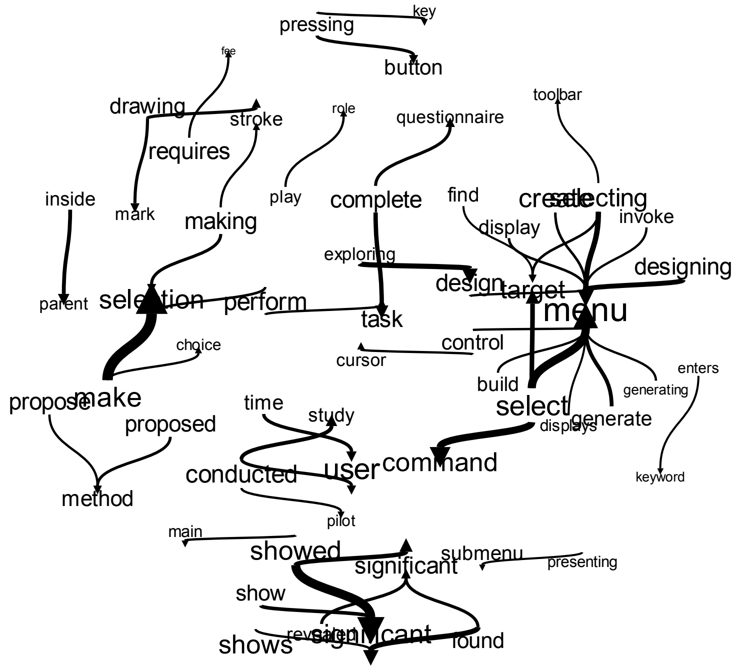


Figure A.11: XaY

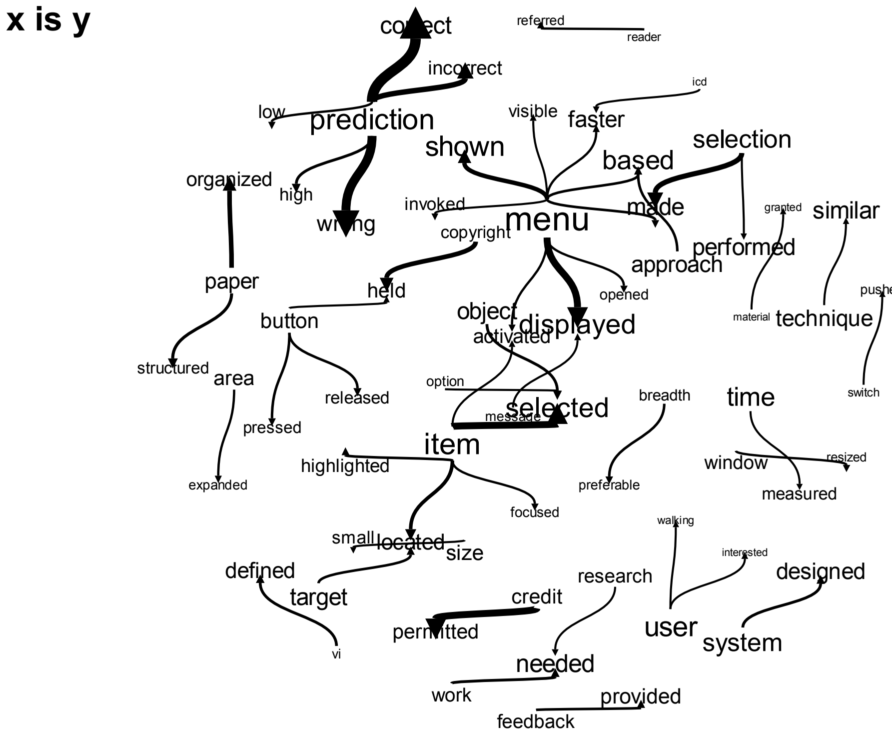


Figure A.12: XisY

A.1.1 XofTheY

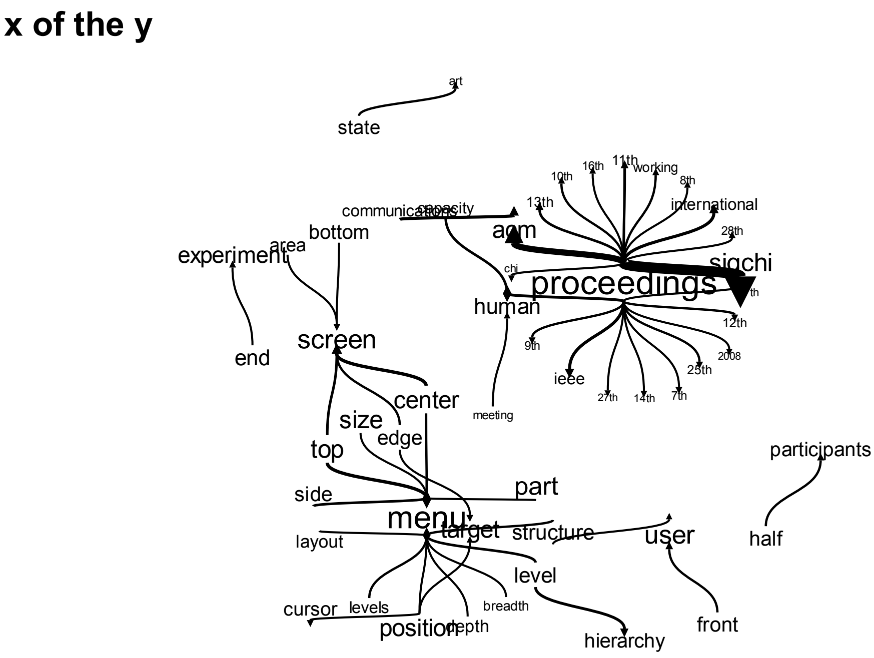


Figure A.13: XofTheY

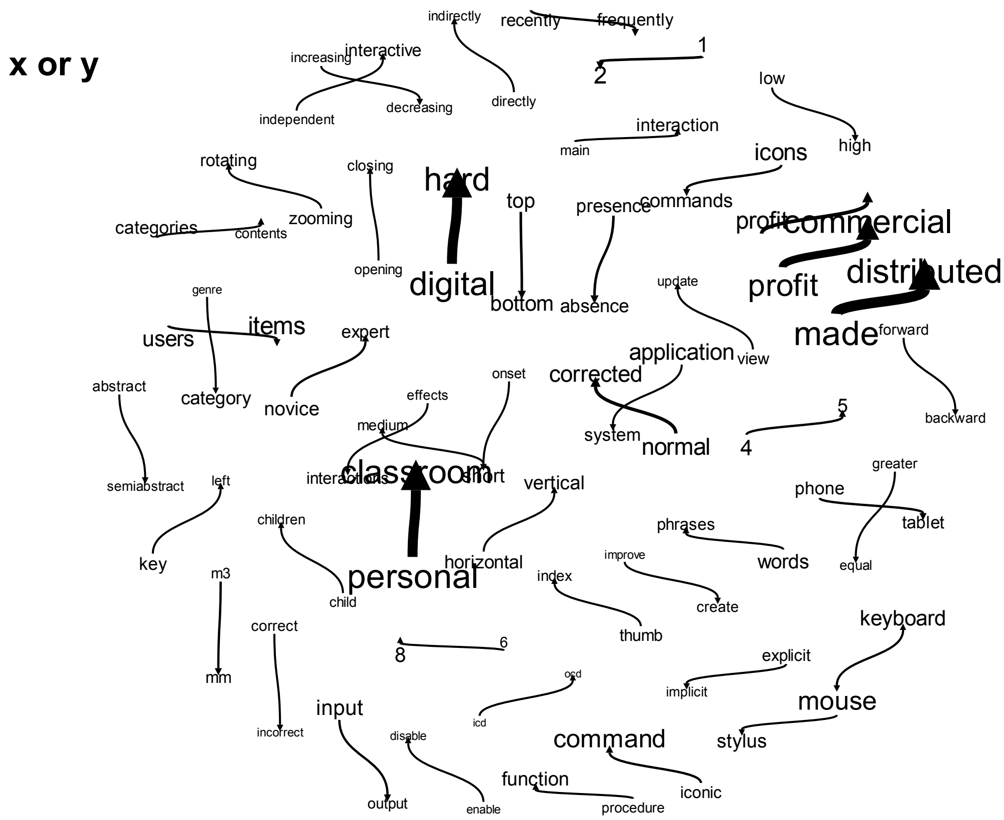


Figure A.14: XorY

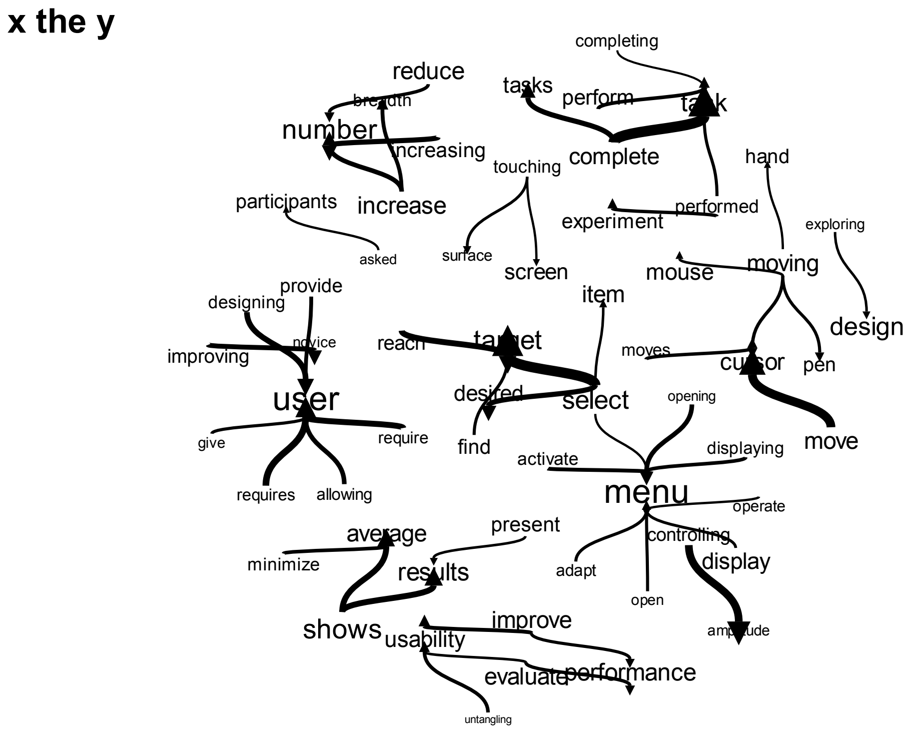


Figure A.15: XTheY

Appendix B

Fractal Menu appendix

B.1 Fractal Menu Class Diagrams

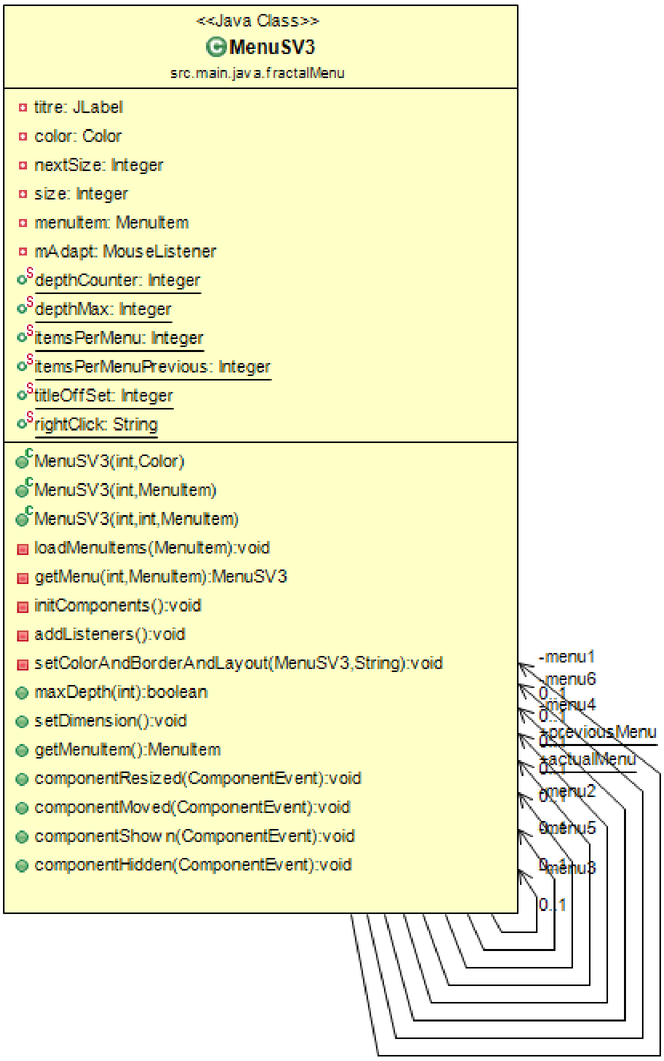


Figure B.1: Rectangular Fractal Menu Class Diagram

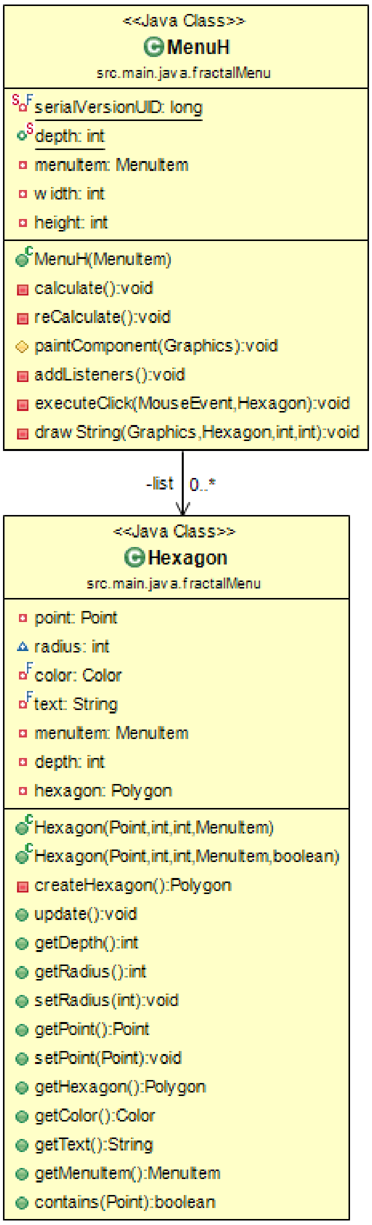


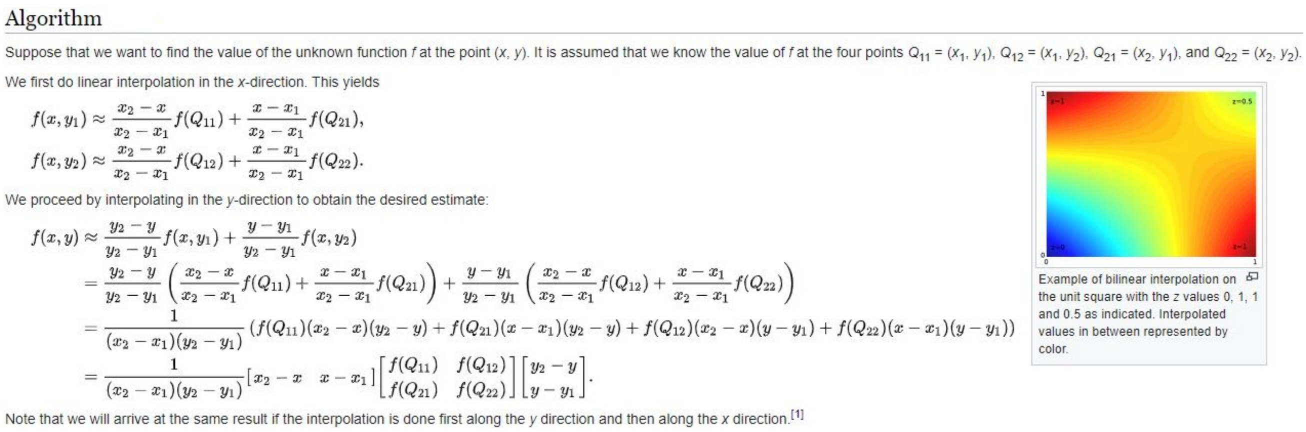
Figure B.2: Hexagonal Fractal Menu Class Diagram

Figure B.3: Global Class Diagram Fractal Menu

Appendix C

Bilinear Menu appendix

C.1 Appendix: Bilinear interpolation algorithm for Chapter 4



Algorithm

Suppose that we want to find the value of the unknown function f at the point (x, y). It is assumed that we know the value of f at the four points $Q_{11} = (x_1, y_1)$, $Q_{12} = (x_1, y_2)$, $Q_{21} = (x_2, y_1)$, and $Q_{22} = (x_2, y_2)$.

We first do linear interpolation in the x-direction. This yields

$$f(x, y_1) \approx \frac{x_2 - x}{x_2 - x_1} f(Q_{11}) + \frac{x - x_1}{x_2 - x_1} f(Q_{21}),$$

$$f(x, y_2) \approx \frac{x_2 - x}{x_2 - x_1} f(Q_{12}) + \frac{x - x_1}{x_2 - x_1} f(Q_{22}).$$

We proceed by interpolating in the y-direction to obtain the desired estimate:

$$\begin{aligned} f(x, y) &\approx \frac{y_2 - y}{y_2 - y_1} f(x, y_1) + \frac{y - y_1}{y_2 - y_1} f(x, y_2) \\ &= \frac{y_2 - y}{y_2 - y_1} \left(\frac{x_2 - x}{x_2 - x_1} f(Q_{11}) + \frac{x - x_1}{x_2 - x_1} f(Q_{21}) \right) + \frac{y - y_1}{y_2 - y_1} \left(\frac{x_2 - x}{x_2 - x_1} f(Q_{12}) + \frac{x - x_1}{x_2 - x_1} f(Q_{22}) \right) \\ &= \frac{1}{(x_2 - x_1)(y_2 - y_1)} \left(f(Q_{11})(x_2 - x)(y_2 - y) + f(Q_{21})(x - x_1)(y_2 - y) + f(Q_{12})(x_2 - x)(y - y_1) + f(Q_{22})(x - x_1)(y - y_1) \right) \\ &= \frac{1}{(x_2 - x_1)(y_2 - y_1)} \begin{bmatrix} x_2 - x & x - x_1 \end{bmatrix} \begin{bmatrix} f(Q_{11}) & f(Q_{12}) \\ f(Q_{21}) & f(Q_{22}) \end{bmatrix} \begin{bmatrix} y_2 - y \\ y - y_1 \end{bmatrix}. \end{aligned}$$

Note that we will arrive at the same result if the interpolation is done first along the y direction and then along the x direction.[1]

Example of bilinear interpolation on the unit square with the z values 0, 1, 1 and 0.5 as indicated. Interpolated values in between represented by color.

Figure C.1: Bilinear interpolation algorithm for Chapter 4

UNIVERSITÉ CATHOLIQUE DE LOUVAIN
École polytechnique de Louvain
Rue Archimède, 1 bte L6.11.01, 1348 Louvain-la-Neuve, Belgique | **www.uclouvain.be/epl**