DEPARTMENT OF INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Bachelor's Thesis in Informatics: Games Engineering

Closed Loop Generative Design Recommender System for Clothing

Mert Uelker

DEPARTMENT OF INFORMATICS

TECHNISCHE UNIVERSITÄT MÜNCHEN

Bachelor's Thesis in Informatics: Games Engineering

Closed Loop Generative Design Recommender System for Clothing

Ein Closed-Loop Generatives Recommender System für Modedesign

Author:	Mert Uelker
Supervisor:	Prof. Dr. Nils Thuerey
Advisor:	Theodoros Papadopoulos
Submission Date:	16.12.2019

I confirm that this bachelor's thesis is my own work and I have documented all sources and material used.

Munich, 16.12.2019 Mert Uelker

Acknowledgments

I would first like to express my gratitude to my supervisor, Professor Nils Thuerey, for giving me the opportunity to carry out this research. I would also like to acknowledge Dirk Hartmann, who was instrumental in defining the path of my research. This thesis would not have been possible without his initial engagement and insightful comments. My sincere thanks also goes to my advisor, Theodoros Papadopoulos, for his persistent support. His guidance helped me during my research and pointed me in the right direction whenever I needed.

Finally, I take this opportunity to thank my family and friends who unfailingly supported me through the process of researching and writing this thesis. I am grateful for the continuous encouragement and moral support.

Abstract

Mass customization is predicted as a key trend disrupting design and engineering as well as production and service. It leverages flexible computer aided design, engineering and manufacturing systems to realize true custom lot size one products, not only customizing single attributes but the complete product itself. This in particular requires the buyer/user to get involved in the design process. When the information production is poorly managed during this process, this leads to the so-called information overload phenomenon and thus a reduction in decision quality, especially considering that the buyers/users are no design experts. Recommender systems have been widely used to tackle this problem by managing, building and representing information customized for individual users. They remove redundant information and address the overload by providing relevant and personalized information to an individual user. However, most recommender systems depend on explicit user feedback in order to refine user models and make more accurate predictions, which can be an effortful process for the user.

In this thesis we propose a closed loop content-based recommender system specialized on clothing with an emphasis on both implicit and low effort explicit user feedback. The proposed design utilizes unobtrusive preference-/critique-based feedback in order to generate relevant recommendations with minimal user effort. Furthermore, the system makes use of basic physical and visual simulations with the intent of clearly projecting the recommended product to the user and capturing different types of user feedback in return. The user feedback is then evaluated in the generative design/optimization loop in the form of key performance indicators to refine the user model and generate better recommendations, increasing both accuracy and usability.

Contents

1 Introduction

High product customization and individualization is one of the latest trends in many industries such as automotive and fashion. Individuals have varying design preferences and aesthetic standards, which can not always be fulfilled by mass-produced products. While customized products can address this issue, they also require the customers' involvement. The customization is mainly driven by customer engagement during the design process and is related with an abundance of design options. The configuration of a product with such a plethora of offerings quite often turns out to be exhausting and not an added value to the customer. Since most customers are not design experts, exposure to too much information can prevent them to make effective decisions and the design process can become overwhelming.

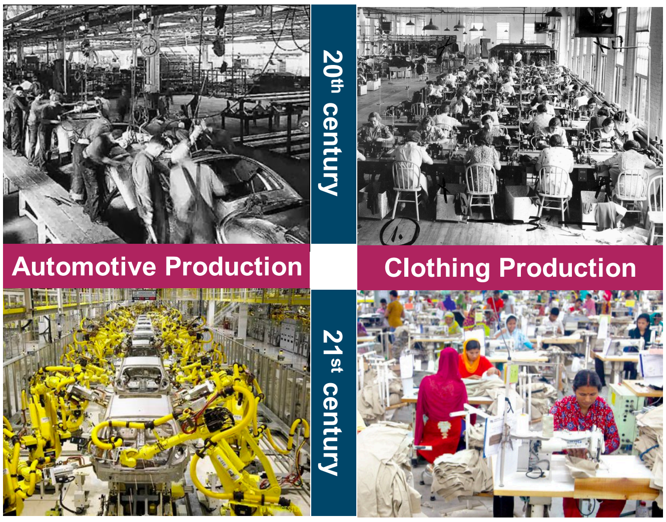

Figure 1.1: A comparison of advancements in automotive and fashion industries, [Pri05][Ped13][Riv19][Udd18]

This study aims to deal with the problem of the design process and produce user-relevant design recommendations. A closed loop generative design recommender system is proposed to overcome this obstacle and an example from the fashion industry is selected as case study to demonstrate the concept. The selected fashion industry example for the proposed system is the creation of a t-shirt design. This industry is chosen because even in the era of digitalization, it still mainly applies the same methods and techniques similarly to the last century. Solutions such as our concept can be the first step for disrupting this industry's manufacturing processes. The proposed recommender system employs eye tracking techniques for implicit user feedback along with low-effort explicit user feedback. The target user's design preference is inferred through the captured feedback and is exploited to produce relevant individualized design recommendations. With the intent of clearly representing the intermediary and final designs for the target user, the proposed system additionally utilizes physics simulation and visualization techniques. The recommender system generates a baseline design solely based on implicit feedback and builds a closed loop for design refinement, capturing user feedback and simulating the refined design by turns. Through the adoption of these techniques, our proposed recommender system is capable of generating accurate individualized clothing designs while still keeping the required user effort minimal.

In the following, we first introduce the core concepts and review the related work in Chapter 2. We then continue by introducing our methods and employed tools in Chapter 3, which is split into two sections to introduce the individual components and the overall pipeline separately. The former section covers an in depth analysis of the employed individual components and their contributions to the proposed system, while the latter section focuses on their cooperation and the overall workflow realized by their coupling. We present evaluations of the proposed system and the core concepts, analyze their results in Chapter 4. Finally, we draw conclusions by revisiting the results and present future work in Chapter 5.

2 Related Work

In this chapter, we introduce the core concepts relevant for our proposal. We highlight the benefits derived from the employment of these techniques, systems and methods, while also introducing the related work on the subject. Moreover, we provide examples and/or suggestions for the application of the given concepts.

2.1 Content-Based Recommender Systems

Recommender systems are prevalent tools which mainly aim to produce personalized recommendations for individual users or sometimes, groups of users. Many companies, commercial websites and web services employ recommender systems to represent individually customized information, which in turn produces and/or delivers more relevant items for the user and enhances the overall user experience. Recommender systems in general can be categorized in six types:

- Collaborative Filtering
- Content-based
- Demographic
- Utility-based
- Knowledge-based
- Hybrid

The first two types of systems are the most commonly used and both have their own distinguished strengths and weaknesses. Some services make use of hybrid recommender systems which combine various strategies to benefit from the complement advantages and avoid certain disadvantages of different types of systems [ÇM17]. Collaborative filtering systems require an information domain consisting of users and their expressed preference on various items [ERK11]. Using this approach, they do not depend on item metadata, but only require a collection of users and their respective item ratings as input. Content-based recommender systems on the other hand, do not require any user ratings and exploit item descriptions instead. This section focuses on content-based recommender systems, their techniques, advantages and application domains. Subsequently, we present a review on a subtype of content-based recommender systems, preference-/critique-based recommender systems, and examine the utilization of implicit feedback in content-based recommender systems.

The basic idea behind content-based recommender systems is trying to recommend items to a given user, which are similar to the previously liked items by the same user [LGS11]. These systems aim to match the user profile with the item set in hand without the need of additional user ratings. Unlike collaborative filtering, content-based systems produce recommendations solely based on the user profile built up by analyzing the content of items that the target user has liked in the past [Lü+12]. Various methods can be applied to represent items and match the user model with the item sets, hence the content-based recommender systems can utilize distinct approaches and particularly differ from each other. In general, both the user model and the item model can be represented by vectors, the former consisting of ratings/preferences while the latter is defined by features. The user model and the item model are both exploited to produce the recommendations. The actual recommendation process is built upon matching attributes from those representations, where the result of this matching is a relevance judgment representing the target user's level of interest in the matched item [LGS11]. Thus, the representation of both user and item data have a critical effect on the relevance of the recommendation. The problem of recommendation is practically a search of items whose content is most similar to the content of the previously preferred items [Lü+12]. Given the effect of user/item representations on the recommendation relevance, the problem further evolves into building well-defined models and applying suitable methods for the actual matching.

As expressed by Zenebe and Norcio (2009), the performance of content-based recommender systems highly depends on the user behavior and item feature data along with how this data is represented and inferred. Given insufficient definitions, the data on items' features and users' behavior can be subjective and imprecise, which in turn induces uncertainty on the two sets of data and their relationship [ZN09]. In order to avoid this uncertainty and increase the recommendation relevance, the user and item data should be carefully and fittingly represented. Item representation in content-based recommender systems can be divided into three categories: structured, semi-structured and unstructured item representation. The first category of item representation, being the most well-organized type fittingly, consists of attribute-value sets describing the item data. Under this category, the items for potential recommendation are often represented in database tables built from records containing values for each attribute/property [PB07]. Since each item is described by the same set of attributes and known set of possible values, the items are represented by means of structured data and as a result, many machine learning algorithms may also be used to learn a user profile [LGS11]. These algorithms can help to decide whether a new item is likely to be of interest for the target user. On the other hand, some content-based recommender systems make use of unstructured item representations rather than extracting features and building structured representations. Instead of defining attributes and sets of values, unstructured knowledge sources can be used in more direct ways such as natural language item description texts. These sources include a variety of user-generated content like reviews and tags, which can lead to rich item representations [Lop+19] but can also increase ambiguity as a downside. Other content-based recommender systems exploit semi-structured item representations, since many domains are best represented by some attributes with sets of restricted values and some free text fields [PB07]. Semi-structured representations do not solely depend on free text in contrast to unstructured representations; therefore, the complications deriving from natural language ambiguity are reduced.

The user representation in content-based recommender systems consists of the information describing the user interests. This information can be embodied by rating vectors similarly to collaborative filtering recommender systems or, as Pazzani and Billsus (2007) concentrate on, user preference models and/or user interaction/transaction history. Possible representations of the user preference model include descriptions of the types of items that interest the user and functions predicting the likelihood of user interest for any given item. User interaction/transaction history on the other hand, can store viewed/purchased items, typed queries or gaze/dwelling data by the user [PB07].

Along with the item and user representations themselves, representation matching methods also play a critical role in producing relevant recommendations for the target user. Content-based recommender systems depend on similarity metrics among items to be able to generate these recommendations. Lü et al. (2012) name attributes, contents and tags as exploitable information to define or improve the similarity metric. The similarity between objects can be obtained by calculating the correlation of their corresponding attribute vectors, based on the assumption that two users/items are similar when they have many common features. Exploiting the tags, which provide a rich source of information, algorithms can be easily designed to calculate user/item similarity by considering tag vectors in user/item space [Lü+12]. Moreover, additional clustering and nearest neighbor methods are commonly used among content-based recommender systems to distribute items into respective clusters/neighborhoods, and thus defining similarity among items. Other content-based recommender systems may utilize more sophisticated methods such as linear classifiers and probabilistic methods.

Content-based recommender systems have many advantages making them suitable contenders for recommendation generation. Among those is the content-based recommender systems' independence from other users' data. In contrast to collaborative filtering, content-based recommender systems solely exploit the ratings of the target user. As another advantage, these ratings do not have to be explicit but can rather be inferred from the interaction/transaction history or the user preference model. Furthermore, since content-based recommender systems generate recommendations exploiting the item similarity, they can also recommend items which are new to the item set. Collaborative filtering recommender systems, on the other hand, need the new item to be rated by a number of users before being able to recommend it to a target user. Lops et al. (2011) draw attention to yet another advantage of the content-based recommender systems, which is the transparency coming from content features. Unlike collaborative filtering systems, which can only explain an item recommendation by "unknown users with similar tastes liking the recommended item", content-based systems can provide more transparent and informative explanations. These type of recommender systems can provide explanations on how the system works by explicitly listing content features or descriptions that caused an item to be recommended [LGS11]. While having many advantages, content-based recommender systems have their shortcomings as well. Disadvantages include the item model's limitation to the analyzed features and the required effort to build the item model. Generally, only a shallow analysis of certain kinds of content can be supplied and the items in some domains may not be amenable to any useful feature extraction methods [BS97]. Another limitation of the content-based recommender systems is the overspecialization or the portfolio effect of these systems. Since the recommendations are based on item similarity, the target user would not receive any unexpected item recommendations. Due to this effect, content-based systems perform usually poorly regarding their serendipity and diversity. While almost unsusceptible to the new item problem, content-based recommender systems may still suffer from the new user problem. In order to produce accurate recommendations for a new user, the system first needs to collect data and build the target user's profile. This means that the recommender system can not produce a reliable recommendation before understanding the user preferences, which is a problem more commonly encountered among collaborative filtering recommender systems.

Apart from the obvious benefits such as personalized content and item recommendations for the users, recommender systems also have noticeable benefits for businesses. Given their ability to produce relevant and personalized recommendations, an increase in the user engagement can be realized on web services or businesses adopting recommender systems. Since the users have to put less effort to find liked or interesting products, there is also an increase in the usability of these services and this leads to users being more satisfied with their overall experience using these systems. These benefits can have direct effects on the revenue of a business, which is strongly related to the primary purposes of a business. Recommender systems have therefore been utilized for a variety of different domains. Previous literature reviews and surveys indicate that majority of recommender systems were designed and developed for recommending web pages, personalized news, music, movies, documents and information [RT08] [MC17]. These domains are suitable for recommender systems for many reasons, one of them being the generally encountered information overload. As expressed by Burke et al. (2011), personalized recommendations have been an important part of many online e-commerce applications such as Amazon, Netflix and Pandora; leading researchers to find inspiration for extending the reach of these systems into new and challenging areas. The research of recommender systems, which encompasses many information access environments, has seen an expansion of interest in the past decade; catalyzed partly by the Netflix Prize and evidenced by the rapid growth of the ACM Recommender Systems Conference [BFG11]. Furthermore, Lops et al. (2019) report that even though the current research landscape of recommender systems is dominated by pure collaborative methods, considering information about the content is already of high importance in today's practical applications and will be increasingly relevant in the future. This is due to the fact that content-based recommender systems have proven to be more effective than the pure collaborative filtering based methods in real-world applications of some domains. Another reason is that the only way of effectively implementing some aspects of the recommender systems is by considering knowledge about item features. One further reason for the potential increase of content-based systems' relevancy is the increasing number of knowledge sources which contain richer item information continuously [Lop+19].

2.1.1 Preference-/Critique-Based Recommender Systems

Most recommender systems adopt an approach where the target user simply receives a personalized recommendation regarding the user/item model and can express no further input on the recommendations. Conversational recommender systems on the other hand, provide the users with the option to give further feedback on the recommended item and thus, the recommendation can be further refined/improved. These recommender systems can utilize different methods to realize the user feedback elicitation and produce further personalized and well-suited recommendations for the user. Commonly used methods for this purpose are retrieving preference-based user feedback and critique-based user feedback. The former type of feedback can be described as the user expressing preference for a recommended item over a set of recommendations; while the latter consists of the user critiquing certain features of the recommended product, such as color, price or size. Both types of user feedback result in a refined item recommendation, which in turn may lead to an iterative user feedback elicitation. Recommender systems usually exploit both types of feedback with similar models. Chen and Pu (2012) describe a typical critique-based recommender system with a four-step interaction model between the user and the system as depicted in Figure 2.1. The first step consists of the elicitation of user's initial preferences, which can be derived from a specification of a reference product or certain value preferences over item features, and is followed by the second step, where the system returns a recommendation according to the initial preferences. The final two steps build a cycle together, where the user can critique the recommended items and get refined recommendations as a next step or can terminate the interaction by selecting an item as final [CP12]. Preference-based recommender systems usually adopt a similar model; but instead of critiquing certain item features, the user expresses preference for a recommended item over others. The recommendation refinement can then be realized by exploiting the preference expression and item similarity metrics.

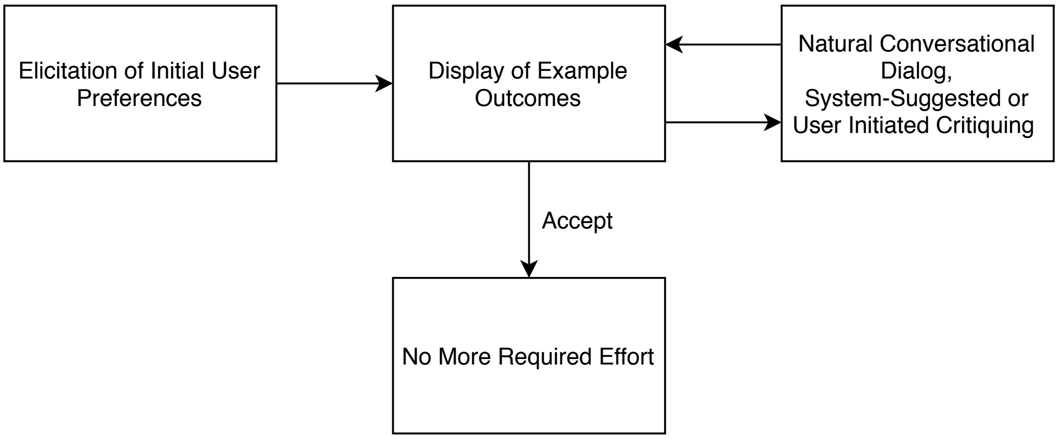

Figure 2.1: The typical interaction model between users and a critiquing-based recommender system. Adapted from "Critiquing-based recommenders: survey and emerging trends", [CP12]

Preference-/critique-based recommender systems are adopted by many e-commerce sites and web services today because of their advantages over other types of recommender systems. One of the most notable advantages of these systems is their higher reliability in comparison with their counterparts. Since the acquired preferences are explicitly given by the user, these recommender systems are able to generate more reliable and accurate recommendations for the target user and can still function with minimal user effort [RN05]. Although it is undeniable that these recommender systems require more user effort than their implicit feedback-based alternatives, it is possible for them to keep the required user effort low enough while still benefiting from higher reliability/relevancy of the item recommendations. Apart from being more reliable, explicit feedback is also easier to interpret and evaluate in comparison to implicit feedback, which also makes the adoption of preference-/critique-based systems more appealing. Among those web services that adopt this type of recommender systems are Amazon and Movielens, which exploit critiquing methods to refine item recommendations [CP12]. Given their advantages, it is quite certain that these recommender systems will continue to be adopted by many businesses of different domains.

2.2 Implicit Feedback in Recommender Systems

User feedback plays a crucial role in elicitation of user preferences and refinement of the user profile in recommender systems. Recommender systems have no way of developing insight regarding users' impression of recommended items without the user feedback. Therefore, the user feedback is essential for producing relevant item recommendations and improving these recommendations. User feedback in recommender systems is divided into two categories: explicit user feedback and implicit user feedback. Explicit feedback, as the name implies, consists of any kind of feedback explicitly given by the user; such as ratings, critiques and constraints. Implicit feedback, on the other hand, is rather inferred from the user behavior. Most collaborative filtering recommender systems rely on explicit feedback collected directly from the user, which can be in form of ratings and reviews; but it is either impossible or very difficult to obtain this type of feedback in some environments [LPP08]. Due to the many advantages offered by implicit feedback and the occasional insufficiency of explicit feedback, the former type of feedback is widely adopted by recommender systems in a variety of application fields.

Implicit feedback has it's edges and drawbacks when compared to explicit feedback. Among many advantages of implicit feedback in recommender systems, one of the most prominent is the lower or no required user effort. Since the user feedback is inferred by the system rather than being explicitly given by the user, the required user effort to generate recommendations and to refine them is kept to minimum. This induces an abundance of user feedback for the system and higher usability for the user, while reducing the accuracy in turn. Being easier to extract, implicit feedback is generally less accurate than it's explicit counterpart because it still has to be inferred by the system instead of being explicitly provided with tools such as rating scales and constraint/critique inputs. Jawaheer et al. (2010) summarize the characteristics of implicit/explicit feedback with Table 2.1, which serves as a clear overview of advantages/disadvantages of the two types of feedback over another. While having differences in terms of accuracy and abundance, both implicit and explicit feedback are sensitive to the user's context. Another difference between the two types of feedback is the degrees of expressivity of the user's preferences. Recommender systems utilizing explicit feedback typically employ rating scales, which the users can use to rate the items with varying scores. In this way, both positive and negative user preferences can be captured, whereas implicit feedback can only be positive. One further difference between implicit and explicit feedback is that explicit feedback tends to concentrate on the extrema of the rating scale because of users' inclination of expressing preference on strongly liked/disliked items more commonly. Implicit feedback, on the other hand, represents a degree of preference and is therefore relative, whereas explicit feedback is absolute [JSK10].

	Implicit Feedback	**Explicit Feedback**
Accuracy	Low	High
Abundance	High	Low
Context-Sensitive	Yes	Yes
Expressivity of User Preference	Positive	Positive and Negative
Measurement Reference	Relative	Absolute

Table 2.1: Characteristics of explicit and implicit feedback. Adapted from "Comparison of implicit and explicit feedback from an online music recommendation service", [JSK10]

Given the abundance of implicit feedback sources, recommender systems can potentially utilize a wide variety of implicit feedback types. These can vary from displayed/purchased items and time spent on item pages by the user to more complex alternatives, such as user gaze count/duration and recognized user emotion. Oard and Kim (1998) categorize observable behavior for implicit feedback initially in three fields: *Examination*, *Retention*, *Reference*; later on adding a fourth category: *Annotation* (2001). *Examination* consists of observable behaviors such as selection of individual objects, which can provide a first cue about a user's interest, and item purchase/subscription. Both total and regional examination duration also belong in this category. Furthermore, the authors include the repetition of these user behaviors in the same category since examination may extend beyond a single interaction between the user and system. The second category introduced by them, *Retention*, aims to cluster the behaviors that suggest some degree of intention to make future use of an item. They group "printing" with this category because of the permanence of the printed page and the association of printing with a desire for retention. They also include "deleting" in the same category, which might support the inference of a degree of preference between the deleted item and the retained ones [OK98]. Their other two categories, *Reference* and *Annotation*, primarily covers actions with the effect of establishing some form of link between items and actions that intentionally add to the value of an information object, respectively [OK01]. The categorization, originally consisting of four categories, is further improved by Kelly and Teevan (2003) with the additional category *Create*. They describe this category as a grouping of behaviors the user engages in when creating original information. The authors also suggest that their classification of behaviors should be viewed as a sample of the possible behaviors rather than an exhaustive classification [KT03]. Another further improvement to the categorization is made by Jannach et al. (2018) with the introduction of *Social & Public Action* and *Physical Action*, where they depicted the extended categorization of observable behavior with Table 2.2. In contrast to the entries in the previous categorizations, their additions are not focusing on document-centric applications but rather on a wider application field based on the review of more up to date scenarios. Their first additional category, *Social & Public Action*, covers typical social interactions such as posting and commenting publicly, following and connecting with people. The information publishing on social media and the user's embedding within a social network can be exploited as another type of implicit preference signal. The content of the posts can be analyzed to build user profiles reflecting user's interests and the social neighborhood of the user can be analyzed to recommend additional friends/followees and channel subscriptions along with their topics. The content of these posts can be further exploited to compute more relevant query results. The final category presented by them, *Physical Action*, covers observable behaviors such as the past and current movement profile of the user, which can be a valuable indicator of the user's interests. The same category also includes additional recognizable activities which can be derived from a variety of devices connected to the Internet of Things. These can be automatic identification of users or the use of gaze tracking in combination with explicit ratings to derive content-based interest profiles [JLZ18]. While the categorization is not fully comprehensive and there may exist some overlapping between certain categories, it still represents a sufficient overview for the different types of observable behaviors.

Concrete examples for the applications of implicit feedback include movie/video streaming services such as Youtube, Hulu and Netflix. Along with their use of explicit feedback by incorporating like/dislike functionalities, these systems also benefit from implicit user feedback. They analyze the content previously watched by the user and infer other media of potential user interest based on the content level similarity of the items. Some of these systems also employ informative cues for the user by displaying a relevancy score accompanying the recommended item. Music streaming services such as Spotify and SoundCloud utilize implicit user feedback in a similar manner. These

Category	Examples of Observable Behavior
Examination	Duration of viewing time, repeated consumption, selection of text parts, dwell time at specific locations in a document, purchase or subscription
Retention	Preparation for future use by bookmarking or saving a named or annotated reference, printing, deleting
Reference	Establishing a link between objects. Forwarding a document and replying, creating hyperlinks between documents, referencing documents
Annotation	Mark up, rate or publish an object (includes explicit feedback)
Create	Write or edit a document
Social & Public Action	Public posting, commenting and communicating, activity posts, following and connecting with people, joining groups, expressing trust
Physical Action	Observed user actions that can be interpreted as feedback towards objects of the physical world. Being at a location, roaming profiles and dwelling time, other recognizable activities in the physical world

Table 2.2: Extension of observable behavior types. Adapted from "Recommending Based on Implicit Feedback", [JLZ18]

services provide recommendation playlists based on their similarity with the previously listened songs or artists. The recommendations include songs from the same/related artists and of the same/related genres. Such systems also make recommendations based on user similarities. User models can be built and refined with the implicit feedback, in turn finding similar users and recommending the content they consume. Another type of service utilizing implicit user feedback is e-commerce. Services such as Amazon and eBay make product recommendations to the users based on their previously examined or purchased products. Google Maps also incorporates a recommender system utilizing both explicit and implicit user feedback. Along with it's exploitation of explicit user ratings, the web mapping service recommends restaurants or cafes to the target user by exploiting the location of the user. Other applications of implicit feedback include social media services such as Facebook and Instagram. These social media services make page, friend or content suggestions based on previously read articles, other visited pages and the social network of the users.

2.2.1 Eye Tracking in Recommender Systems

Eye tracking is the technique and process of measuring a user's eye motion and position. Utilizing eye tracking technologies, recommender systems can extract user data such as gaze location, duration and pupil diameter. Although it is not as commercially widespread as other implicit feedback alternatives, eye tracking in recommender systems is still a commonly studied type of implicit user feedback. Eye tracking technologies' lack of popularity among commercial recommender systems is mainly related to hardware issues, along with possible privacy concerns. A dedicated eye tracking component should be incorporated for a recommender system to utilize eye tracking feedback, which can introduce physical constraints to the system and the user. The need to constrain the physical relationship between the eye tracking system and the user remains as one of the most significant barriers to incorporation of eye tracking in more settings [JK03]. Nevertheless, eye movement recordings can provide a dynamic trace of where a user's attention is being directed in relation to a visual display and the extracted data such as fixations can reveal the amount of processing being applied to objects at the point of regard [PB06]. Using this information, recommender systems can infer user preferences in order to refine the user model and produce relevant recommendations.

As with any type of implicit feedback, eye tracking feedback has a clear advantage over explicit feedback types regarding the required user effort. Since the eye movements happen nearly unconsciously, they share the advantage of being easy for the user to generate [JK03]; making the use of eye tracking in recommender systems unobtrusive. Eye tracking additionally provides a distinct advantage in terms of the eye-mind assumption relating overt (visual) and covert (cognitive) attention [FJ11]. The eye-mind assumption holds that the gaze target of the eyes strongly relates to what the mind is engaged with [JC80]. Even though the advantages mostly make up for them, eye tracking feedback still has it's disadvantages. These disadvantages include the eye tracking devices' need for calibration before use, although this only takes few seconds for most devices, and the limited head movement of the user [Pal+08]. One further disadvantage of eye tracking can be it's high financial impact on research [Sch+03] or commercial use. Besides potentially being preferred over explicit feedback types, eye tracking feedback can also be preferred over it's implicit counterparts due to a couple of reasons. These reasons include eye tracking feedback's high versatility, as they can potentially be adapted to any system supporting visualization, and it's continuity. One example to eye tracking being preferred over an implicit counterpart is the study presented by Franco-Watkins and Johnson (2011). The authors discuss that the eye tracking methodologies introduced by them have an advantage over mouse tracing in that they produce a greater number of fixations, of shorter duration, and are not susceptible to significant variability over the course of an experiment [FJ11]. Eye tracking, by all means, may also be employed alongside other types of implicit feedback in order to have improved results. In their study, Schiessl et al. (2003) acquire results which indicate that click analysis alone is insufficient for their use case. They express that the additional eye tracking method in their studies gave insight into otherwise not reportable behavior right before or in absence of the mouse click. Moreover, they conclude that the combination of conventional methods and eye tracking analysis allowed differentiating the origins of the problem [Sch+03].

Eye tracking feedback and it's correlation with user interest/attention has been researched and utilized in a multitude of different fields. A wide variety of (interdisciplinary) eye tracking applications has been and still being developed in fields including but not limited to neuroscience and psychology, industrial engineering and human factors, marketing/advertising and computer science [Duc07] . A large number of research focusing on (implicit feedback) recommender systems have also utilized eye tracking on multiple occasions. Xu et al. (2008) propose a recommendation algorithm for personalized online documents, images and videos using commodity eye tracking. Their proposed recommender system recommends an optimal list of online materials most interesting to the target user under the assumption that an item which attracts more attention from the user, in comparison to another item of same type, also indicates a higher interest for the user. They define the attention with the time the user spends on reading, browsing or watching the object and their reported statistics suggest that their algorithm can satisfactorily produce a personalized online content recommendation which is in better agreement with the user's expectation and preference [XJL08]. Chen and Pu (2010) investigate the effect of the recommender interface designs on users' decision making strategies through the observation of their eye movements and product selection behavior. With this study, they aim to understand the visual searching pattern of users in a ranked list and to reveal more effective layout designs that can prompt users to consider more recommendations, in turn leading to more accurate decision outcomes. Reviewing their eye tracking results, they conclude that the users practically adapted their searching behavior to different recommendation displays and suggest layouts which can lead to a higher decision quality [CP10]. Relying on a critique-based recommender system, Castagnos et al. (2010) examine the impact of recommender systems on users' product search and purchase decisions. By employing an eye tracking system and collecting users' interaction behaviors, they outline how users progressively use the recommender system and how users need to explore recommended alternatives as they get closer to their desired item [CJP10]. In a more practical application, Kliegr and Kuchař (2014) present a recommender system for online videos, employing eye tracking to determine user interest in the content. Their proposed recommender system exploits eye tracking and motion sensing alongside explicit user actions and constructs user profile from preference rules discovered with an association rule learner [KK14]. Motivated by the eye gaze patterns' capacity to hold rich signals concerning user preference and by the unavailability of eye tracking data to most recommender systems, Zhao et al. (2016) demonstrate the possibility to model and predict user gaze without requiring the deployment of eye tracking technology. They propose predicting gaze by combining the user browsing data with the eye tracking data collected from a small number of users and inferring patterns for other users using this combination. Their results show that the incorporation of eye tracking data into the model training significantly boosts accuracy in comparison to only using normally logged user browsing data [Zha+16]. On yet another study, Chen et al. (2017) engage in analyzing users' eye movement behavior during the evaluation of recommendations from a critique-based recommender system, aiming to identify the correlation between the users' eye movement and critiquing feedback. In their study, they utilize the metrics fixation count, total fixation duration, average fixation duration and they define *improving/compromising* for the actual critique done by the user on a product's attribute. Following an in depth investigation of eye movements over recommended products through an eye tracking experiment, they conclude the feasibility of using the users' fixation behavior to infer the product which they are inclined to critique within a set of recommendations. They discover that fixation count and total fixation duration are more indicative of a user's interest in one product for critiquing, while average fixation duration acts more accurately than the former two metrics for inferring a user's critiquing criteria on a product's attributes. Their conclusions also include that the product attributes with relatively high average fixation duration can more likely be improved and those with low average fixation duration are more likely to be compromised. Furthermore, they discuss that users' attribute fixations can be combined with their value comparison behavior for deriving high-confident association rules. These findings suggest that the implicit critiquing feedback based on eye movements can be exploited to enhance the accuracy/relevance of system-suggested critiques and that recommender systems following this approach can automatically refine users' preference model [CWP17].

3 Tools and Methods

In this chapter, we review our proposed closed loop content-based recommender system by introducing the employed methods and the utilized tools for the realization of the proposed system. We analyze the individual components and their contributions to the system in general and regarding the current application setting, which is the generation and refinement of a clothing design in brief. We realize the recommender system by constructing the core component (Simulation/Visualization and User Interaction Component) with Unity and coupling all individual components together. The components building our recommender system are as follows:

- User Feedback Component
- Evaluation/Quantification Component
- Simulation/Visualization and User Interaction Component
- Database Component

The following components are proposed for an extension of the system:

- Optimization Component
- User Scanning Component

We analyze these components in detail regarding their implementation and/or integration, their roles and advantages; while also discussing the possible limitations and alternatives for these components. In addition to the detailed analysis of the individual components, we review their relations and their cooperation with each other. We further on introduce the current pipeline, which consists of two phases: **Baseline Design Generation Phase** and **Design Attribute Refinement Phase**. Moreover, we review the extended pipeline of the proposed system and the enhancements realized by it's adoption.

3.1 Individual Components

3.1.1 User Feedback Component

The user feedback component handles the retrieval of implicit user feedback to infer the target user's preference over a set of designs or over specific attributes of a single design. The component is only responsible for the capturing of the user feedback and not for the interpretation, which is handled by the evaluation/quantification component. Considering the many advantages of eye tracking, mainly unobtrusiveness/noninvasiveness and versatility, we employ an eye tracking system in our recommender system in order to capture the implicit user feedback. While there are many devices and software products offering eye tracking solutions, we choose our solution according to it's accuracy, convenience and prevalence among eye tracking research. For that purpose we use Tobii Pro Nano, a highly mobile and convenient screen-based eye tracker. The eye tracker unit applies the corneal reflection and dark/bright pupil combination eye tracking techniques, while achieving an accuracy of 0.3° at optimal conditions. It also offers a Unity software development kit (SDK) for building analytical applications, allowing straightforward integration with Unity. Figure 3.1 shows a sample scene using the Tobii Pro Unity SDK, where the green dots represent the previous eye fixation points by the user. The eye tracker's fast calibration, straightforward integration, high portability and accuracy makes it a suitable tool for our user feedback component.

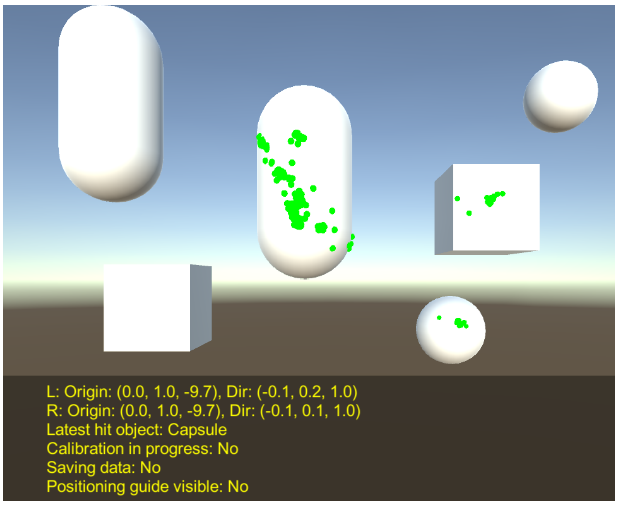

Figure 3.1: Screen capture from a sample Tobii SDK scene with visualized gaze data

Our proposed user feedback component collects the following metrics for each design object/region: fixation count, total fixation duration, average fixation duration, average pupil diameter. Fixation count represents the total number of times the user fixates on the item and total fixation duration represents the sum of these fixations' duration, while the average fixation duration and average pupil diameter represent the average duration of the user's fixation on the item and average pupil diameter of the user during a fixation on the item, respectively. The captured user feedback is sent over to the evaluation/quantification component for further processing in form of collected metrics, alongside item characteristics such as attribute/region type and visible surface area or volume for each item of potential interest.

Another alternative feedback that the component can incorporate, instead of or supplementary to eye tracking, is user emotion. Potentially, the target user's emotions during examination of an item can be identified and extracted for refining the user preference model. Possible instruments for identifying the target user's emotion during item examination are electroencephalography (EEG) devices or galvanic skin response (GSR) sensors. The former are usually in the shape of headsets and measure the electrical activity of the brain while the latter can be wearable units usually attached to finger tips, measuring the electrical conductivity of the skin. Both type of instruments can potentially be employed to measure psychological arousal and identify human emotions. However, our preliminary research shows that the utilization of this type of feedback to identify product emotions and to infer user preference can be quite challenging, especially with EEG [FB+07][Mot09][GE13]. Although there are studies showing encouraging results of neural response exploitation for applications such as experience good and aesthetic preference recognition [Ma+18][CTM15], 3D user interface interaction [FHL17]; both EEG and GSR approaches suffer from challenges such as invasiveness/obtrusiveness, idiosyncrasy and high sensitivity to noise. Given these challenges, we decide that capturing and analyzing eye tracking feedback is a more suitable option for our proposed recommender system.

In addition to implicit user feedback, we propose the utilization of low-effort explicit user feedback by the system. In the second phase of the recommendation process, the target user's interest on design regions of a single simulated design is measured, instead of measuring interest on design pictures with varying attributes. During this phase we propose suggesting the user a set of updated design attributes based on the region with the highest interest. Thus, the user can explicitly choose one of the displayed design suggestions, updating the currently simulated design in turn. The process of capturing low-effort explicit feedback is conducted in cooperation with user interaction component. While user feedback component is responsible for any kind of implicit feedback including the one leading to the explicit inquiry, acquisition of the actual explicit feedback is conducted by the user interaction component. In Subsection 3.1.3, we discuss the advantages of utilizing low-effort explicit feedback and provide more details on this approach.

3.1.2 Evaluation/Quantification Component

The evaluation/quantification component is responsible for processing the captured feedback transferred by the user feedback component. The component exploits the implicit feedback data by converting the received measures (fixation count, total/average fixation duration and average pupil diameter) to a more tangible metric, interest score. The component calculates the interest score with a weighted combination of the measures it receives from the user feedback component, also exploiting the volume or visible surface area of the respective item in order to have equitable and accurate results. Using the calculated interest scores, the component determines the changes to be made on the baseline design and sends this information to the simulation/visualization component to realize these changes. For the aforementioned low explicit feedback, a direct evaluation is possible and the feedback does not need to be further processed. Instead, the changes resulting from the explicit feedback are realized directly by the simulation/visualization component.

3.1.3 Simulation/Visualization and User Interaction Component

Simulation/visualization and user interaction component is the core component connecting the other individual components and building the foundation for the overall workflow. The component is responsible for visualizing the design representations and simulating the existing design among other user interactions such as low explicit feedback. By visualizing and simulating the designs, the component creates the opportunity to capture user feedback, in turn refining the user preference model and producing more relevant recommendations for further simulation. The component visualizes/simulates the designs until feedback is captured and initializes or updates the designs according to the results from the evaluation component, realizing the basis for the closed loop cycle of the recommender system.

The whole component is constructed with Unity, a game engine allowing the development of 2D/3D games and simulations. Offering a wide variety of features such as animation, rendering and simulations tools, Unity is adopted by many different industries besides game development. These industries include engineering, automotive and architecture, given the game engine's ability to represent and simulate full-scale industry models in real-time, possibly in augmented reality (AR) and virtual reality (VR). Due to similar reasons, we employ Unity for our core component, specifically version 2018.4.9.

For the actual simulation of the design, we employ a cloth simulation subcomponent along with a human model in order to produce a realistic representation of the design. Our preliminary research results with three candidates for the realization of this subcomponent: NVIDIA FleX for Unity, Obi Cloth and Unity's built-in cloth component. All three are real-time cloth simulation components utilizing particle-based physics and have their respective edges/drawbacks. Upon further review of these subcomponents, we determine that Obi Cloth is the most suitable option for our proposed recommender system due to NVIDIA FleX's requirement of high processing power and Unity Cloth's very low adaptability. While Obi Cloth has lower performance requirements in comparison to NVIDIA FleX, it still produces far more accurate and flexible simulations than Unity's built-in cloth component. Furthermore, Obi Cloth has a far more comprehensive documentation compared to the other alternatives, making the integration and adjustment of the subcomponent more straightforward. Obi Cloth is an highly flexible simulator, allowing for many constraint adjustments such as aerodynamic, bend and skin constraints which are depicted in Figure 3.2 along with it's particle editor. Global constraints along with particle constraints make it possible to refine the cloth further and produce realistic simulations. The simulator can also be employed to add cloth simulation to skinned meshes, such as characters or other humanoid models. While the cloth simulator is an highly adaptable one, it's application with skinned meshes still requires an additional integration process; which is crucial for our recommender system, since we simulate the generated clothing design worn by a human model to present more realistic results. We use Blender for this additional integration, an open source 3D computer graphics software for modeling, rigging, animation and simulation among other tasks. The process starts with setting up the armature and generating the rig for the human mesh, which can be skipped if the human mesh is already rigged. Furthermore the human mesh and the cloth to be simulated have to be skinned to the rig, which is followed by export from Blender and import into Unity. Figure 3.3 displays the part of the process after the rigging of the human mesh. This complex process introduces some limitations for our recommender system, such as the need for skinning different types of clothes to the human model beforehand. Another limitation is that there is no optimal solution for dynamically scaling a Unity gameobject with a skinned mesh renderer component, which makes it challenging for our recommender system to refine individual design attributes such as sleeve length. However, the exploitation of per-particle defined skin constraint parameters of Obi Cloth, such as skin radius and backstop, still gives potential for refining the tightness/looseness of the generated design.

The simulation/visualization and user interaction component also handles the capture of low-effort explicit user feedback by employing a user interface for the selection of system-suggested critiques. While the users do not explicitly state which design aspect to critique, they have the freedom to choose any of the suggested critiques based on the highest interest design region. An example to this feedback process is depicted in Figure 3.4, where the user is presented with different options of collars because the

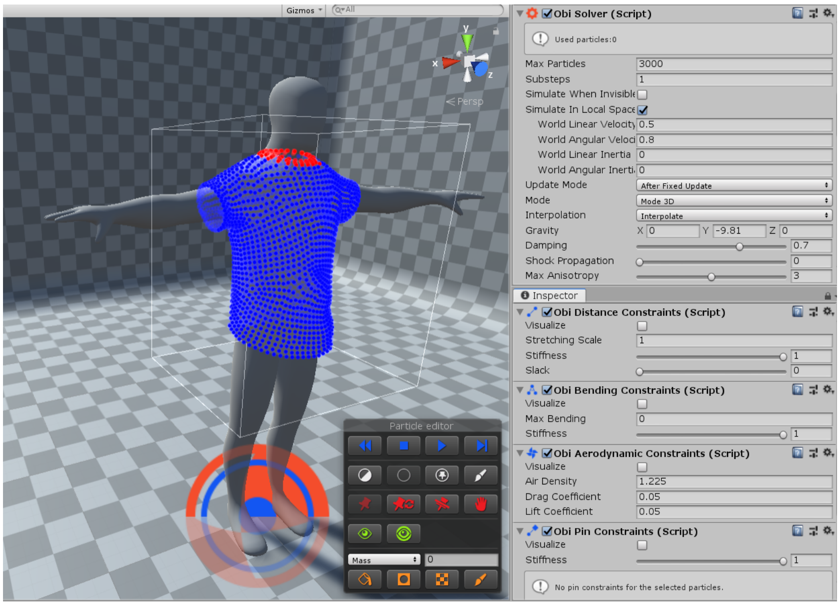

Figure 3.2: Screen capture of Obi Cloth simulator displaying solver parameters, constraints and particle editor within Unity

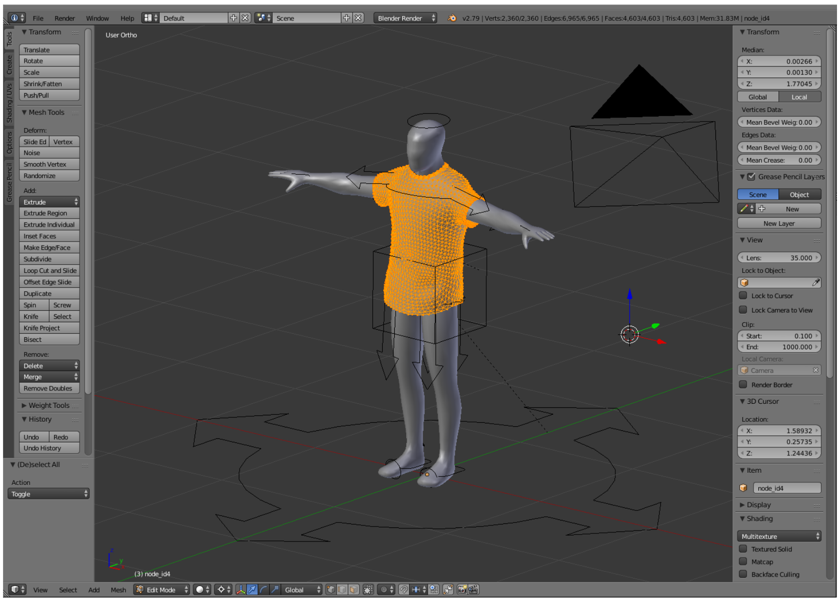

Figure 3.3: Screen capture from Blender depicting the skinning of a cloth mesh to a rigged humanoid mesh

current design region with the highest user interest is the neck region. We propose this approach due to the challenge in refining an individual design attribute during a single design simulation. Adapting the design solely based on user interest on the design region, without retrieving the explicit feedback may produce irrelevant recommendations; since a reference data regarding the same design aspect is not present during the simulation of a single design item. By adopting the current approach, we obtain higher accuracy and relevance in the recommended design with respect to the target user's preferences, while still keeping the required effort to minimum. Instead of explicitly choosing the design attribute to critique, the user can confirm one of the suggestions made by the system, avoiding higher effort on their part.

Figure 3.4: An example of system-suggested critiques

3.1.4 Database Component

The database component stores sets of design representations along with individual design attributes defined by discrete parameters. The stored data can be in the form of images such as design visualizations used for the inference or in the form of gameobjects such as different collar options for the design. Additionally, the database component stores numerical data representing different design attribute values including logo size and design color values. The stored data is used by the simulation/visualization component to generate new designs or modify existing designs. While the continuous parameter design attributes such as design color and logo scale/position can be modified solely by simulation/visualization component, the discrete parameter design attributes are stored in database component and get transferred to the simulation/visualization component accordingly. The discrete parameter attributes include design fabrics, collar styles and the logos among others. While we currently exploit both continuous and discrete parameters, our future work focuses more on continuous parameters due to the further enrichment of design options they provide.

The component is in closed loop interaction with the simulation/visualization component to generate the designs and realize the changes in the existing designs. It also handles the extraction of data such as eye tracking metrics, interest scores and current design parameters, which are necessary for our current studies and the future integration of the optimization component. Besides storing design data, cooperating with simulation/visualization component and extracting existing data, the database component is also responsible for parsing the data outputted from the optimization component. By parsing the data from the optimization component, which we thoroughly discuss in next section, and transferring this data to the simulation/visualization component, the database component plays a critical role in further optimizing the existing design in addition to it's other responsibilities.

3.1.5 Extension Components

Optimization Component

We envision the optimization component for the further and more accurate refinement of our proposed recommender system's recommendations. The recommender system currently produces recommendations according to the internal evaluation of implicit and low-effort explicit user feedback. In order to improve the relevancy of the recommendations and potentially replace low-effort explicit feedback, we propose employing Hierarchical Evolutionary Engineering Design System (HEEDS) for this component. HEEDS is a design space exploration software that automates and improves the search for better solutions within a design space. The software makes it possible to uncover new design concepts that improve products, leveraging optimization algorithms such as genetic and adaptive algorithms along with machine learning techniques. Given it's features including process automation, efficient search and distributed execution, it is utilized across many industries such as automotive, energy and aerospace. Our recommender system can utilize HEEDS to increase the accuracy of the recommendations and to completely remove the explicit user feedback. There are multiple potential solutions to replace the low-effort explicit feedback during the design refinement. One approach is randomly applying changes on the high interest region and feeding HEEDS the current parameters such as eye tracking metrics and design parameters in each refinement iteration, in turn decreasing randomness in each iteration and refining the design towards more optimized solutions. HEEDS can also potentially be utilized to generate a set of optimized design variations instead of a single one and the recommender system can suggest these optimized designs to the user, again relying on low-effort explicit feedback.

While our current recommender system is not integrated with HEEDS design optimization, the required infrastructure is already prepared. This allows the system to communicate with HEEDS and conduct design of experiments (DOE). Current integration supports the utilization of predefined design parameters by HEEDS in cooperation with the database component, which handles the communication by parsing the incoming data from HEEDS and extracting internal data to HEEDS. By employing the optimization component fully integrated with HEEDS, the proposed recommender system can produce more relevant recommendations and also further reduce the required effort for the target user.

User Scanning Component

We propose the addition of this future component to the recommender system in order to have more realistic simulations and individualized results. The user scanning component can be used at the start of the design process to extract a 3D body scan of the user, replacing the generic human model our current recommender system has. The component comes into effect during the second phase of the design process, where a single design is simulated in order to infer regional interest by capturing implicit user feedback. Furthermore, the scanned user model can move around wearing the generated design by using predefined animations or tracking the user's real body, in turn giving the user an opportunity to examine the design in more detail. While the suggested component may cause some privacy concerns and introduce invasiness to the recommender system, it can still enhance the user experience and allow the system to produce more individualized recommendations.

3.2 Overall Pipeline

While the core concept can be applied for many different application fields, our proposed closed loop generative design recommender system focuses on the generation of individualized clothes for the target user. The idea behind the recommender system is to generate a baseline design for the target user by exploiting implicit user feedback and simulating this design to capture further user feedback on following iterations. As depicted in Figure 3.5, this iterative design process consists of simulation of the design along with additional feedback retrieval, evaluation of the captured feedback and refinement of the existing design for further simulation; hence building a closed loop generative design process.

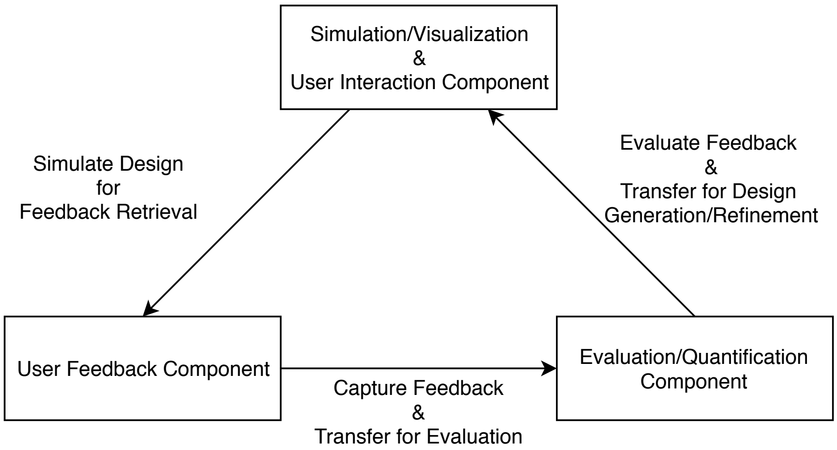

Figure 3.5: Design cycle representing the communication between member components

As shown in Figure 3.6, the overall workflow of our recommender system consists of two phases: **Baseline Design Generation Phase** and **Design Attribute Refinement Phase**. The former phase is constructed with a series of design displays in the form of images and it results with the generation of the baseline design for the target user. Each set of displays focuses on a single design aspect with varying attributes within the same display set in order to infer the most suitable option for the current design aspect, based on the target user's interest on the design option representations. In this phase we primarily focus on non-regional design attributes such as design fabric and color, since the inference of user preference for these attributes proves to be particularly challenging during the next phase, where we measure the user's interest on design regions. During **Baseline Design Generation Phase**, *User Feedback Component* captures the implicit user feedback for each set of design images visualized by *Simulation/Visualization and User Interaction Component* and transfers the user feedback data to *Evaluation/Quantification Component* for further processing. The latter component infers the preference of the target user and sends this information to *Simulation/Visualization and User Interaction Component* which simulates the baseline design in cooperation with *Database Component*. At this stage, **Design Attribute Refinement Phase** starts and the recommender system modifies the simulated design in each refinement cycle. During this phase, *User Feedback Component* and *Evaluation/Quantification Component* measures the target user's interest on different design regions such as neck, torso and sleeve; followed by the transfer of these results to *Simulation/Visualization and User Interaction Component* for the presentation of system-suggested critiques. Utilizing a user interface, target user's explicit feedback is retrieved and the refined design is simulated once again for next iteration of the cycle. The generative design cycle can be limited to a certain number of iterations or alternatively, users can have the option to terminate the interaction when they are satisfied with the existing design recommendation which is an approach commonly adopted by critique-based recommender systems.

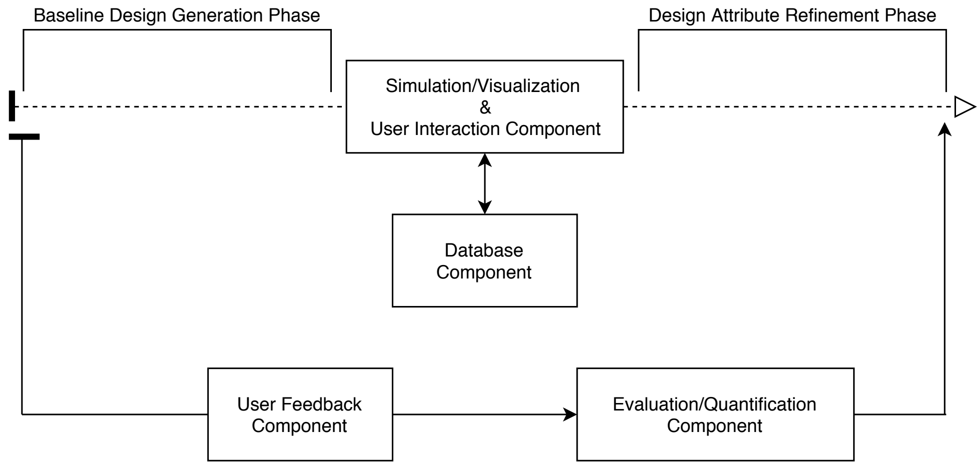

Figure 3.6: Overall pipeline displaying the design phases and component connections

We also propose an extended pipeline with the goal of enhancing the accuracy, diversity and the overall user experience of our closed loop generative recommender system. Alongside the addition of extension components, we propose an additional phase for the pipeline: **Initialization Phase**. This is a very simple phase, consisting of the explicit choice of clothing type by the user at the start of the design process. Since we currently focus on only a single type of clothing, the option to generate and recommend other types of clothes would severely increase the diversity of our recommender system. Figure 3.7 represents the newly proposed extended workflow with the additional phase and the components, which is still closely related to the existing pipeline. As seen in the figure, *Optimization Component* is directly connected with *Evaluation/Quantification Component* in order to increase the accuracy of the system and the *User Scanning Component* begins having an effect at the start of the **Design Attribute Refinement Phase**, replacing the previous human model with the user's scanned body mesh. In addition to these changes, the explicit feedback from the user during the examination of system-suggested critiques can be replaced by implicit feedback as well. The system can adopt an approach similar to that in **Baseline Design Generation Phase**, measuring the user's interest on system-suggested critiques and realizing the changes accordingly and in turn lowering the overall user effort required for the design process. Realizing these changes can further increase the relevancy of produced recommendations, leading to an improved experience for the target user.

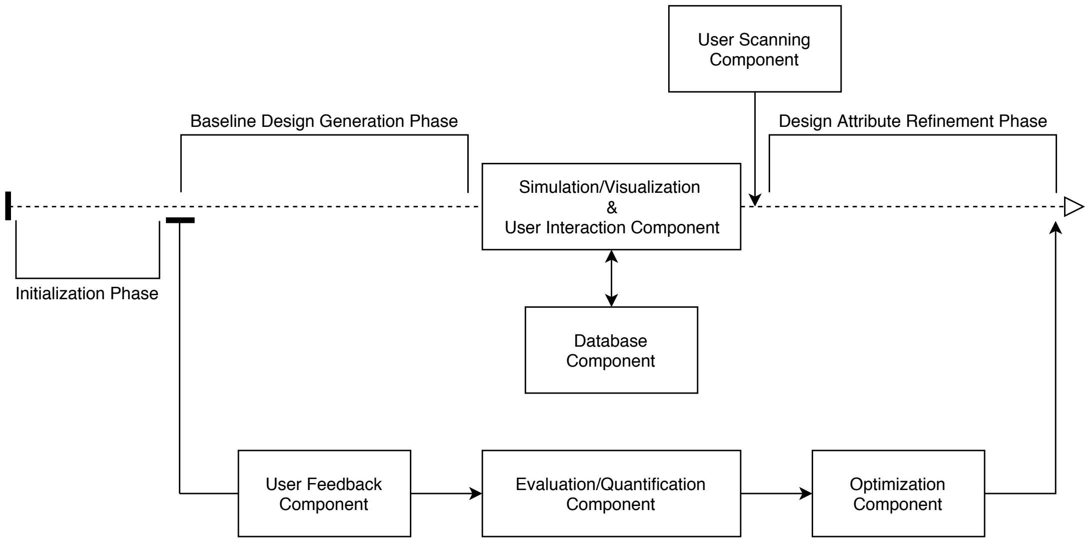

Figure 3.7: Overall pipeline displaying the design phases and component connections with extensions

4 Results and Discussion

In this chapter, the developed workflow presented in Chapter 3 is evaluated using two different case studies. First study focuses on the validity of user preference inference by using eye tracking metrics, whereas second study relates to the complete workflow and it's outcome. In this section, we introduce the conducted case studies and analyze their results.

The goal of our initial study is to verify our preliminary eye tracking concepts. The evaluation was realized by displaying sets of design pictures to the participants, collecting their eye tracking metrics, testing different inference methods and comparing the results with the explicitly given values. Upon the completion of the experiment, the users are asked to explicitly list their rankings for the proposed designs based on their preference, providing us with the comparison values. The study consists of ten sets of design images, each set having four different designs, and all ten iterations of design examination lasts for 20 seconds. A brief calibration is conducted at the start of the study and a resting interval is kept between each iteration in order to increase the precision of our study. Instant transitions or very brief resting intervals between the iterations may affect the precision negatively, while too long waiting time may be boring for the user and lead to a loss of interest. Given these premises, we choose an interval of 3 seconds between each iteration in order to balance user engagement and precision. The images in our data set consist of different clothing designs with varying types such as t-shirt, jacket, dress and each image belonging to the same set are of the same type. Figure 4.1 depicts a sample examination iteration where four t-shirts are displayed for the user to observe. We try to keep a level of similarity between the images in the same set and avoid using distracting designs in order to get more accurate results. Catchy design features such as bright colors or written texts can draw more attention from the user, not necessarily because they are more preferable, but simply because they are more distracting or because the texts are hard to read. Therefore, we avoid using such distracting designs in our sets with the goal of acquiring more equitable and accurate results. Twelve participants, with different educational backgrounds and ages between 20-30 were asked to examine each set of design images and after the participants complete examining each ten sets of images, they were asked to rank the four designs in each set from most favorite to least favorite by showing them the images again without a time limit.

Figure 4.1: Screen capture from the initial study displaying one step of examination

Four preference inference functions, each utilizing only one of the collected eye tracking metrics (fixation count, total/average fixation duration and average pupil diameter), are defined and the results of these functions are compared with the explicitly given user preference ranking of the designs. All four functions simply order the designs in each set by it's utilized comparison metric, from highest metric value to the lowest. Total and average fixation duration inference functions use milliseconds as the comparison unit, whereas average pupil diameter inference function uses millimeters. In order to acquire a more extensive evaluation, four additional functions are defined and are named "inverse" functions for clarity purposes. These inverse functions each exploit only one of the eye tracking metrics similarly to the previously defined functions. Although they also use the same metrics as before, they differ from the previous functions in the direction of ordering. Instead of ranking the designs in a descending order, the inverse functions rank them in an ascending order, the design with the lowest metric value inferred as the most favorite and the design with the highest value as the least. Figure 4.2 displays the results of the comparison between (inverse) fixation count inference and explicitly given preference rankings. Each bar in the chart represents the accuracy of one of the two inference functions (fixation count and inverse fixation count) for a given preference rank. The accuracy is defined as the ratio between the number of correctly inferred designs for that rank and the total number of design sets. The fixation count inference function reaches an accuracy of 75,5% for the first rank preference, meaning that the design attracting the highest number of fixations among a set is actually the favorite design of a user 75,5% of the time. Moreover, the forth rank accuracy of the same inference function has an accuracy of 40%, which indicates that the design with the least number of fixations also is the least preferred user design with a probability 40%. Furthermore, the inverse fixation count inference has an accuracy of only 2,22%, indicating that the design with the least fixation count is very rarely the most preferred design of the user.

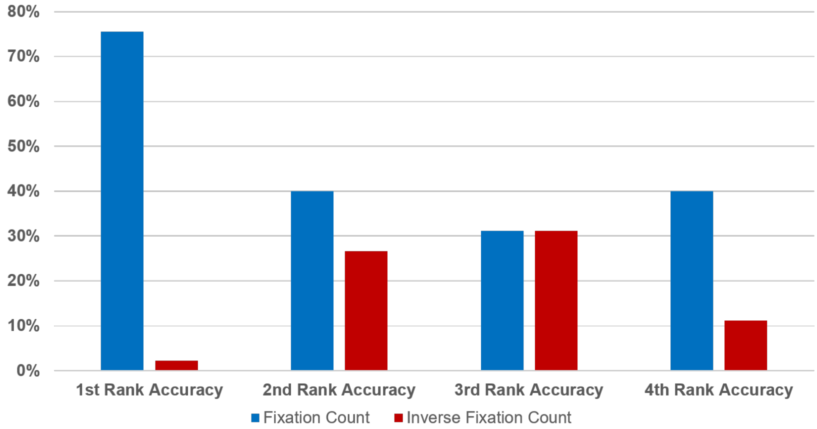

Figure 4.2: A chart depicting the accuracy of fixation count and inverse fixation count inference functions for each preference rank

Figure 4.3 displays the accuracy of total fixation duration and inverse total fixation duration inference functions, with first rank fixation duration having the highest accuracy (71,3%) among others. Moreover, the inverse function of the same metric has a significantly low accuracy for both the first rank and the forth: 2,8% and 5,6%. These low accuracy rates reveal that it is infrequent for a design with the shortest total fixation duration to be the favorite design of a given user. They also suggest that the design with the longest total fixation duration is rarely a least favorite.

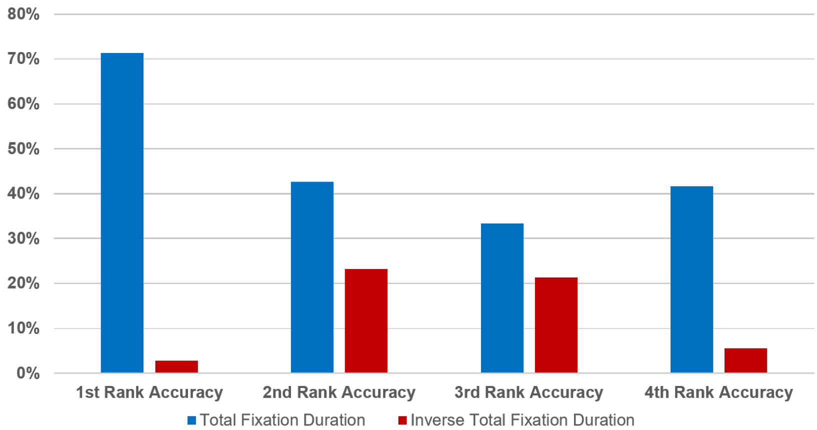

Figure 4.3: A chart depicting the accuracy of total fixation duration and inverse total fixation duration inference functions for each preference rank

While Figure 4.4 suggests that the correlation between the user preference and the average fixation duration is not as strong as the correlation with the previous two metrics, it still conforms to our preliminary assumptions regarding this metric. The accuracy of the average fixation duration inference is higher than inverse's for each rank, with a maximum difference of 36,4% and a minimum difference of 8,1%. These differences and the accuracy values of the two functions support the argument that a design with greater average fixation duration is likelier to be a more preferred design than one with a smaller average fixation duration.

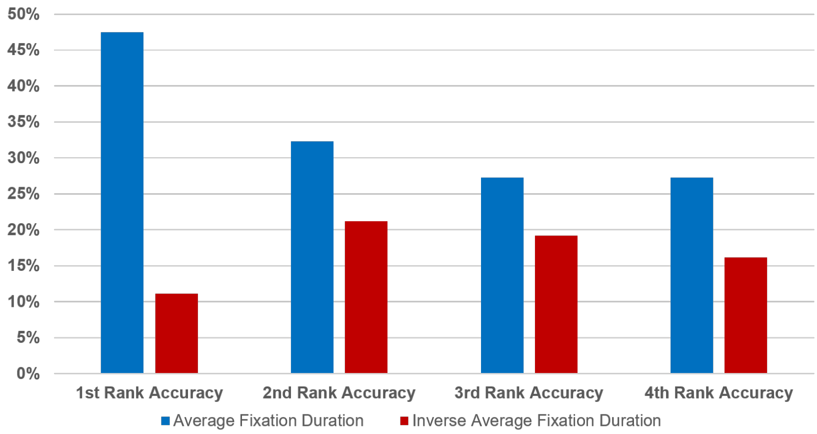

Figure 4.4: A chart depicting the accuracy of average fixation duration and inverse average fixation duration inference functions for each preference rank

Finally, the accuracy of average pupil diameter inference is depicted in Figure 4.5. Even though our preliminary research suggests that a larger pupil diameter correlates with a higher aesthetic rating of an image or preference of a design [BS12][HL14], the resulting data for this metric is rather inconsistent. Potential issues causing this inconsistency might be the bright design colors and environmental lighting. Both of lighting and bright colors can induce pupillary responses that are not necessarily correlated with the design preference, and therefore can result in such inconsistencies. Although the highest accuracy is once again reached by a non-inverse function, it only goes up to the value of 35,8%, which is the lowest observed value for a non-inverse inference function in first preference rank. Furthermore, this inference method is the only one, where the inverse inference function surpasses the original function in terms of accuracy. A difference of 9,2% for the second rank and a difference of 2,8% for the third rank are observed in favor of the inverse average pupil diameter inference. Moreover, the inverse inference function reaches an accuracy of 22% for the 4th rank, which is the highest reached accuracy by an inverse function. The results regarding the average pupil diameter metric contradicts with our initial assumptions and they manifest that this metric is the least accurate for preference inference among all four metrics.

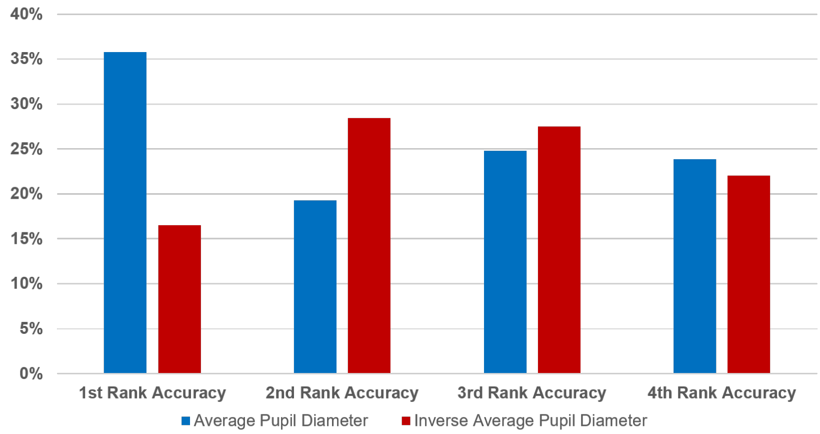

Figure 4.5: A chart depicting the accuracy of average pupil diameter and inverse average pupil diameter inference functions for each preference rank

In addition to previous accuracy comparisons, a comparison of the first rank accuracy of all four eye tracking metrics is shown in Figure 4.6 to provide a clearer overview. Although fixation count inference method seems to be the most accurate of all four methods for the first rank, it should be noted that the respective function has a number of samples significantly smaller than the other three functions. This is due to the high number of designs which share the same number of fixations with another design in the same set. While the fixation count function is highly prone to ties between designs and can not infer a complete ranking of the set occasionally, it is uncommon for the designs to tie with each other under the comparison of the other metrics. Despite average pupil diameter metric proving to be rather unreliable, our results show that the remaining three metrics can be exploited to infer user preference within a set of designs. These results validate our preliminary assumptions that the fixation count and total/average fixation duration metrics have a correlation with the user preference. They also indicate that the exploitation of these three metrics is suitable for the inference of the favorite design, but not as reliable for the inference of other ranks. However, the inference of the favorite user design alone already provides valuable information for the individualized design generation. It is also worth mentioning that the inverse inference functions for the first three metrics score significantly low for the favorite design inference, especially the fixation count and total fixation duration. This information reveals that the designs with lowest number of fixation and shortest total fixation duration are rarely the most preferred option, which can also be exploited during the individualized design generation.

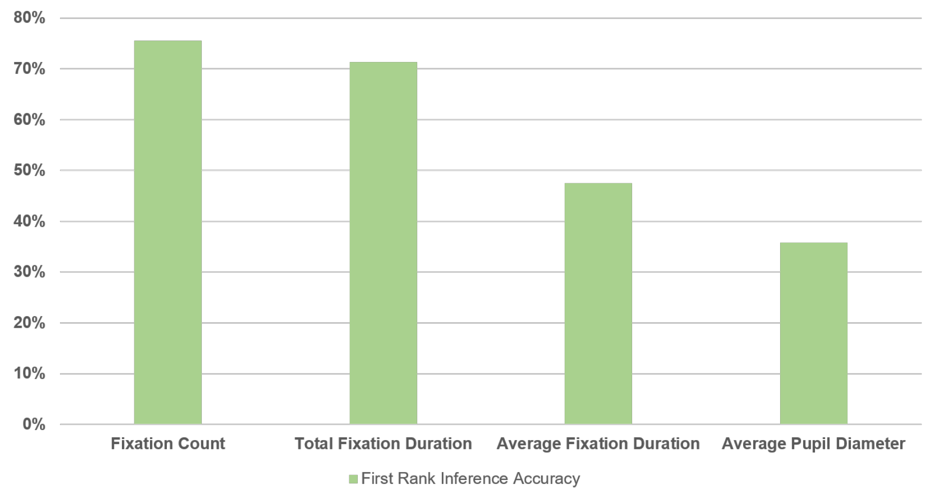

Figure 4.6: A chart comparing the first rank accuracy of four different eye tracking metrics

Furthermore, we define another inference function, constructed from a combination of fixation count and total fixation duration metrics. Since both of these metrics prove to be in correlation with the user preference, the combination inference method also utilizes them in a similar way to previous methods and additionally introduces a threshold. The inference function is only aimed at inferring the most preferred design of the user and begins the evaluation by giving priority to total fixation duration. The design with the highest duration value is directly returned, if the difference between the highest and the second highest value is greater than the threshold. Otherwise, the function considers those designs within the range of the threshold as equally ranked and continues the evaluation with the second metric. The function ranks the designs, which previously had equal rankings, according to their fixation counts. If two or more designs still share a rank after this step, the design with the longest fixation duration is returned, ignoring the threshold. This threshold is defined as one-tenth of the design examination duration, which corresponds to 2 seconds for our initial study settings. The favorite design inference accuracy of the combination method is compared with the total fixation duration metric's in Figure 4.7. We choose the total fixation duration for comparison instead of fixation count, due to the limited sample number of the latter metric. Even though fixation count has a higher favorite design accuracy than total fixation duration according to our previous results, the metric is highly prone to equal rankings caused by multiple designs with same number of fixations. Therefore, total fixation count metric alone is not robust enough for the inference, especially with short examination phases. The total fixation duration method has an accuracy of 71,3% for the inference of the favorite design, whereas the combination method reaches a value of 72,5%. Although the difference is a minor one, it is still promising. The combination method is more robust in comparison to using solely fixation count, because this method can produce inferences for most design sets, where as the fixation count alone is highly prone to multiple equally ranked designs. More research still has to be done to further test the validity of our assumption and to produce better results. Eventually, more optimized and reliable combinations of these metrics will support our recommender system, in turn increasing the accuracy further.

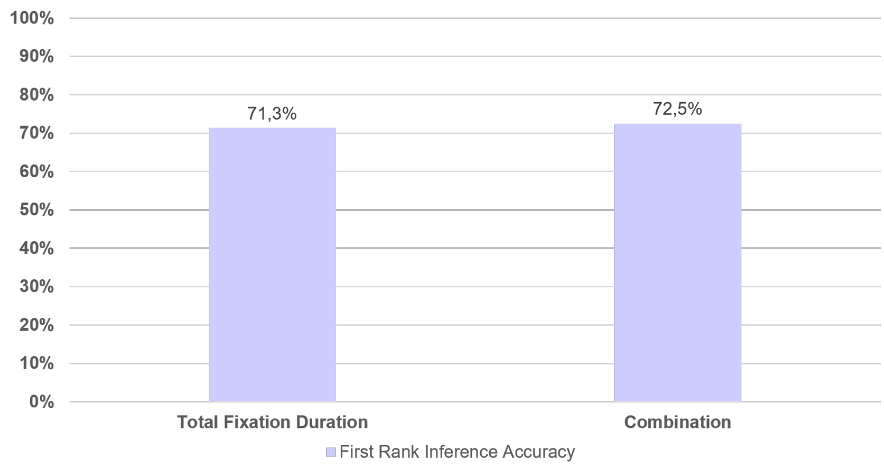

Figure 4.7: A chart comparing the first rank accuracy of total fixation duration inference and the combination inference

Our second study consists of the testing of the proposed recommender system by users and a user survey for the evaluation of the workflow. The study aims to evaluate the overall user satisfaction of the proposed recommender system and is in the form of a brief workflow representing the baseline design generation phase of the design process. The inference of the user preference is completely based on eye tracking feedback, as proposed for the baseline design generation phase. For the study, we have a total of twenty-three participants mostly consisting of simulations experts between the age of 30-55 and students from disciplines such as information technology, computer sciences and electrical engineering, between the ages 21-25. The demonstrated design process starts with a brief calibration, which is followed by the display of design aspect options divided into different steps. In each step, the user investigates a set of design pictures representing varying options for specific design aspects such as design color and logo for a 10 second interval. Figure 4.8 shows one of the examination steps where implicit user feedback is captured, while Figure 4.9 shows a generated baseline design as a result of the design process. After the examination steps, the generated baseline design is simulated for the user, worn by a generic human model that can be rotated by user interaction in order get a better view of the design. Besides allowing the users to see different parts of the design, the rotation of the model also helps the users to get a better idea about how the generated design looks while moving.

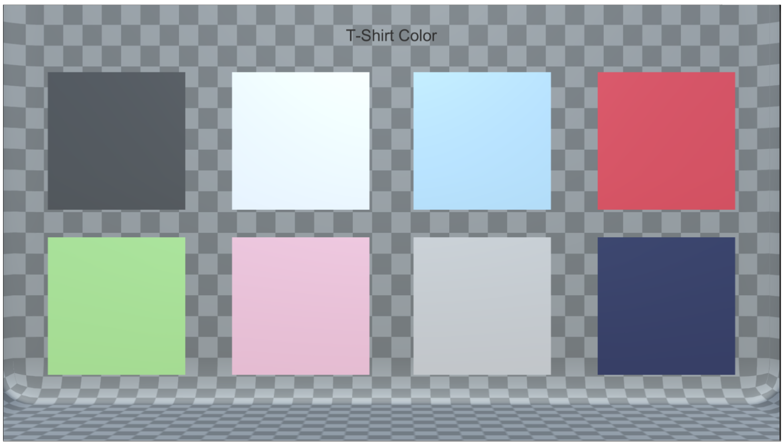

Figure 4.8: An examination step during baseline design generation displaying different color options for the design

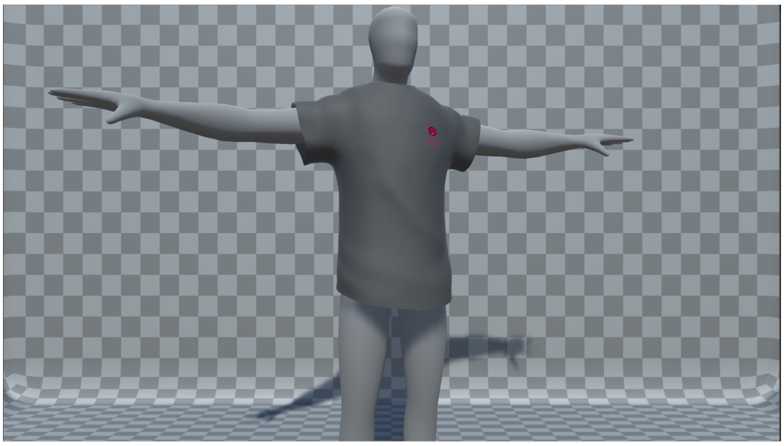

Figure 4.9: A sample from the design process depicting generated baseline design

Figure 4.10 depicts the overall workflow of the case study with specific design attribute inference steps. As mentioned, an eye tracking calibration is performed for the user at the start of the study, before the design aspect examination steps. Each examination step is designated for the inference of a specific design attribute, starting with the design fabric inference. The examination steps of the workflow continue with logo size/color inference and conclude with the design color inference. Following the completion of the examination steps, the baseline design is generated and simulated for the user observation. After this stage, the participants are asked to fill out the user survey consisting of questions aimed at deducing user satisfaction and gathering feedback for our workflow.

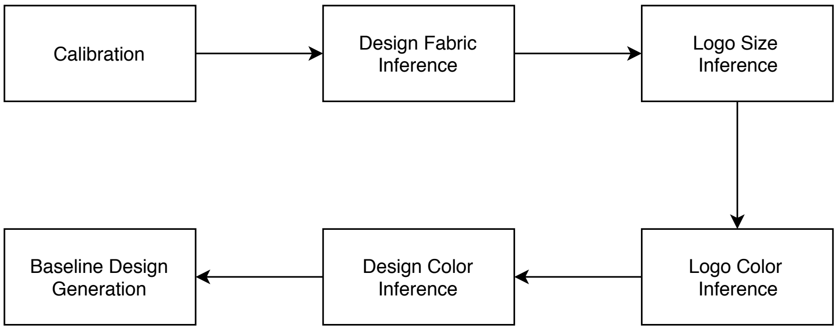

Figure 4.10: A diagram depicting the overall workflow of the study

The results derived from user evaluations reveal valuable information regarding the accuracy of the proposed recommender system, along with it's edges and drawbacks. Out of twenty-three total participants, seventeen confirm that the proposed design matches their preference, while the remaining six are not fully satisfied due to certain design aspects not fitting their preference. These unfitting design aspects include t-shirt color, fabric, looseness/tightness and the most common among those, logo color. Although the participants indicate positive impressions for the lack of required user effort, the precision of eye tracking and cloth simulation, the data collected from the users also suggests improvements for the workflow. The two most commonly addressed issues are the need for additional parametrizable design features and a more neutral background. Feedback from participants indicate that the current background creates a challenge for the examination of some design aspects and should be changed to a more neutral one. Within this study the users were asked to provide feedback about potential commercialization of this solution. They were also asked if they had any concerns regarding the privacy, despite all collected data being anonymized. The results of the user surveys suggest that the users are divided regarding the privacy concerns using eye tracking for a real-world application. While a majority of participants express that they do not have any privacy concerns as long as consent was given beforehand, there are still some participants having privacy concerns for the use of eye tracking. In general, the participants indicate their desire for complete awareness of the tracking and per occasion allowance for the utilization of this technology.

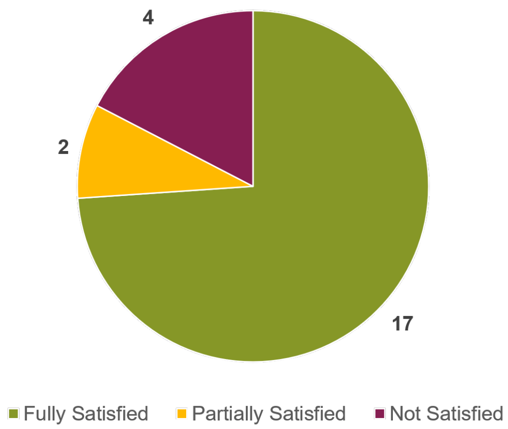

Figure 4.11: A chart displaying the user evaluation results for the baseline design generation study

The results from these studies show that the core utilized concepts of our proposed recommender system are valid and the system itself is viable. While there is a room for improvement, the recommender system produces encouraging results. The system is capable of producing relevant individualized designs, fitting the users' preference. With more optimized inference methods and minor modifications to the workflow, our proposed recommender system can be further enhanced to improve the user experience and generate even superior recommendations.

5 Conclusion and Future Work

This work presents a novel approach for generating user-relevant individualized clothing designs based on implicit and low-effort explicit user feedback. We propose a closed loop generative design recommender system with an emphasis on eye tracking for the generation of custom products. The proposed system makes use of realistic cloth simulations with the goal of accurately capturing user feedback and refining the generated designs incrementally. The results from the preliminary studies are encouraging for the validity of exploiting eye tracking feedback for the inference of user design preference. The results conclude that by combining multiple eye tracking metrics, the proposed recommender system is able to infer the favorite user design among a set of designs with an accuracy of 72,5%. They also display that our design generation workflow matches the user preferences with a confirmation rate of 74%. User evaluations show that our proposed recommender system is capable of producing relevant design recommendations and generating individualized clothing designs accurately while still requiring minimal user effort. The proposed recommender system can assist the users through the design process of customized products by addressing the information overload and significantly lowering the required effort.

Current studies only evaluate a fraction of the whole concept. Further improvements can be applied to optimize the evaluation of the user feedback. The potential improvements include defining more suitable evaluation methods and fully integrating the recommender system with the optimization component HEEDS. Further user studies and machine learning approaches can lead to improved feedback evaluations and to an increased relevancy of produced recommendations. Moreover, the recommender system can be enhanced by utilizing additional design parameters with an emphasis on continuous parameters, in turn further expanding the design options for the product generation. The utilization of VR technologies along with more realistic simulations would potentially allow the recommender system to create an immersive and improved user experience. Furthermore, technologies such as facial electromyography and facial recognition can be researched and experimented with in order to acquire supplementary implicit feedback and increase the accuracy of our recommender system to a greater extent. In addition to these changes, all existing explicit user feedback can be replaced with gaze interaction techniques with the aim of further reducing the required user effort and enhancing the overall user experience.

List of Figures

List of Tables

Bibliography

[BFG11] R. Burke, A. Felfernig, and M. H. Göker. "Recommender Systems: An Overview." In: *AI Magazine* 32.3 (June 2011), pp. 13–18. DOI: 10.1609/aimag.v32i3.2361.

[BS12] K. Blackburn and J. Schirillo. "Emotive hemispheric differences measured in real-life portraits using pupil diameter and subjective aesthetic preferences." In: *Experimental brain research. Experimentelle Hirnforschung. Expérimentation cérébrale* 219 (Apr. 2012), pp. 447–55. DOI: 10.1007/s00221-012-3091-y.

[BS97] M. Balabanović and Y. Shoham. "Fab: Content-based, Collaborative Recommendation." In: *Commun. ACM* 40.3 (Mar. 1997), pp. 66–72. ISSN: 0001-0782. DOI: 10.1145/245108.245124.

[CJP10] S. Castagnos, N. Jones, and P. Pu. "Eye-tracking Product Recommenders' Usage." In: *Proceedings of the Fourth ACM Conference on Recommender Systems*. RecSys '10. Barcelona, Spain: ACM, 2010, pp. 29–36. ISBN: 978-1-60558-906-0. DOI: 10.1145/1864708.1864717.

[ÇM17] E. Çano and M. Morisio. "Hybrid Recommender Systems: A Systematic Literature Review." In: *Intelligent Data Analysis* 21.6 (Nov. 2017), pp. 1487–1524. DOI: 10.3233/IDA-163209.

[CP10] L. Chen and P. Pu. In: *Eye-Tracking Study of User Behavior in Recommender Interfaces*. June 2010, pp. 375–380. DOI: 10.1007/978-3-642-13470-8_35.

[CP12] L. Chen and P. Pu. "Critiquing-based recommenders: survey and emerging trends." In: *User Modeling and User-Adapted Interaction* 22.1 (Apr. 2012), pp. 125–150. ISSN: 1573-1391. DOI: 10.1007/s11257-011-9108-6.

[CTM15] L. H. Chew, J. Teo, and J. Mountstephens. "Aesthetic preference recognition of 3D shapes using EEG." In: *Cognitive Neurodynamics* 10 (Nov. 2015). DOI: 10.1007/s11571-015-9363-z.

[CWP17] L. Chen, F. Wang, and P. Pu. "Investigating users' eye movement behavior in critiquing-based recommender systems." In: *Ai Communications* 30.3-4 (2017), pp. 16. 207–222. DOI: 10.3233/Aic-170737.

[Duc07] A. T. Duchowski. *Eye Tracking Methodology: Theory and Practice*. Berlin, Heidelberg: Springer-Verlag, 2007. ISBN: 1846286085.

[ERK11] M. D. Ekstrand, J. T. Riedl, and J. A. Konstan. “Collaborative Filtering Recommender Systems.” In: *Foundations and Trends® in Human–Computer Interaction* 4.2 (2011), pp. 81–173. ISSN: 1551-3955. DOI: 10.1561/1100000009.

[FB+07] O. C. Fenech, J. C. Borg, et al. “Exploiting emotions for successful product design.” In: *DS 42: Proceedings of ICED 2007, the 16th International Conference on Engineering Design, Paris, France, 28.-31.07. 2007*. 2007, pp. 825–826.

[FHL17] J. Frey, M. Hachet, and F. Lotte. “EEG-based neuroergonomics for 3D user interfaces: opportunities and challenges.” In: *Le travail humain* 80.1 (2017), pp. 73–92.

[FJ11] A. M. Franco-Watkins and J. G. Johnson. “Decision moving window: using interactive eye tracking to examine decision processes.” In: *Behavior Research Methods* 43.3 (Apr. 2011), p. 853. ISSN: 1554-3528. DOI: 10.3758/s13428-011-0083-y.

[GE13] G. Guo and M. Elgendi. “A New Recommender System for 3D E-Commerce: An EEG Based Approach.” In: *Journal of Advanced Management Science* 1 (Feb. 2013). DOI: 10.12720/joams.1.1.61-65.

[HL14] C.-H. Ho and Y.-N. Lu. “Can pupil size be measured to assess design products?” In: *International Journal of Industrial Ergonomics* 44.3 (2014), pp. 436–441. ISSN: 0169-8141. DOI: https://doi.org/10.1016/j.ergon.2014.01.009.

[JC80] M. A. Just and P. A. Carpenter. “A theory of reading: From eye fixations to comprehension.” In: *Psychological review* 87.4 (1980), p. 329.

[JK03] R. J. Jacob and K. S. Karn. “Commentary on Section 4 - Eye Tracking in Human-Computer Interaction and Usability Research: Ready to Deliver the Promises.” In: *The Mind’s Eye*. Ed. by J. Hyönä, R. Radach, and H. Deubel. Amsterdam: North-Holland, 2003, pp. 573–605. ISBN: 978-0-444-51020-4. DOI: https://doi.org/10.1016/B978-044451020-4/50031-1.

[JLZ18] D. Jannach, L. Lerche, and M. Zanker. “Recommending Based on Implicit Feedback.” In: *Social Information Access: Systems and Technologies*. Ed. by P. Brusilovsky and D. He. Cham: Springer International Publishing, 2018, pp. 510–569. ISBN: 978-3-319-90092-6. DOI: 10.1007/978-3-319-90092-6_14.

[JSK10] G. Jawaheer, M. Szomszor, and P. Kostkova. “Comparison of Implicit and Explicit Feedback from an Online Music Recommendation Service.” In: *Proceedings of the 1st International Workshop on Information Heterogeneity and Fusion in Recommender Systems*. HetRec ’10. Barcelona, Spain: ACM, 2010, pp. 47–51. ISBN: 978-1-4503-0407-8. DOI: 10.1145/1869446.1869453.

[KK14] T. Kliegr and J. Kuchař. "Orwellian Eye: Video Recommendation with Microsoft Kinect." In: *Proceedings of the Twenty-first European Conference on Artificial Intelligence*. ECAI'14. Prague, Czech Republic: IOS Press, 2014, pp. 1227–1228. ISBN: 978-1-61499-418-3. DOI: 10.3233/978-1-61499-419-0-1227.

[KT03] D. Kelly and J. Teevan. "Implicit Feedback for Inferring User Preference: A Bibliography." In: *SIGIR Forum* 37 (Sept. 2003), pp. 18–28. DOI: 10.1145/959258.959260.

[LGS11] P. Lops, M. de Gemmis, and G. Semeraro. "Content-based Recommender Systems: State of the Art and Trends." In: *Recommender Systems Handbook*. Jan. 2011, pp. 73–105. DOI: 10.1007/978-0-387-85820-3_3.

[Lop+19] P. Lops, D. Jannach, C. Musto, T. Bogers, and M. Koolen. "Trends in content-based recommendation." In: *User Modeling and User-Adapted Interaction* 29.2 (Apr. 2019), pp. 239–249. ISSN: 1573-1391. DOI: 10.1007/s11257-019-09231-w.

[LPP08] T. Q. Lee, Y. Park, and Y.-T. Park. "A time-based approach to effective recommender systems using implicit feedback." In: *Expert Systems with Applications* 34.4 (2008), pp. 3055–3062. ISSN: 0957-4174. DOI: https://doi.org/10.1016/j.eswa.2007.06.031.

[Lü+12] L. Lü, M. Medo, C. H. Yeung, Y.-C. Zhang, Z.-K. Zhang, and T. Zhou. "Recommender systems." In: *Physics Reports* 519.1 (2012). Recommender Systems, pp. 1–49. ISSN: 0370-1573. DOI: https://doi.org/10.1016/j.physrep.2012.02.006.

[Ma+18] Y. Ma, J. Jin, W. Yu, W. Zhang, Z. Xu, and Q. Ma. "How Is the Neural Response to the Design of Experience Goods Related to Personalized Preference? An Implicit View." In: *Front. Neurosci.* 2018.

[MC17] K. Madadipouya and S. Chelliah. "A Literature Review on Recommender Systems Algorithms, Techniques and Evaluations." In: *BRAIN. Broad Research in Artificial Intelligence and Neuroscience* 8.2 (2017), pp. 109–124. ISSN: 2067-3957.

[Mot09] D. Motte. "Using brain imaging to measure emotional response to product appearance." In: *Proceedings of International Conference on Designing Pleasurable Products and Interfaces, Compiegne, France*. 2009.

[OK01] D. Oard and J. Kim. "Modeling Information Content Using Observable Behavior." In: *Proceedings of the Annual Meeting of the American Society for Information Science* 38 (Mar. 2001).

[OK98] D. Oard and J. Kim. “Implicit Feedback for Recommender Systems.” In: *Proceedings of the AAAI Workshop on Recommender Systems*. 1998, pp. 81–83.

[Pal+08] D. Pallez, P. Collard, T. Baccino, and L. Dumercy. “Eye-Tracking Evolutionary Algorithm to minimize user’s fatigue in IEC applied to Interactive One-Max problem.” In: *arXiv preprint arXiv:0803.3192* (2008).

[PB06] A. Poole and L. J. Ball. “Eye tracking in human-computer interaction and usability research: Current status and future prospects.” In: *Encyclopedia of Human Computer Interaction*. Jan. 2006, pp. 211–219.

[PB07] M. J. Pazzani and D. Billsus. “Content-Based Recommendation Systems.” In: *The Adaptive Web: Methods and Strategies of Web Personalization*. Ed. by P. Brusilovsky, A. Kobsa, and W. Nejdl. Berlin, Heidelberg: Springer Berlin Heidelberg, 2007, pp. 325–341. ISBN: 978-3-540-72079-9. DOI: 10.1007/978-3-540-72079-9_10.

[Ped13] A. Pedersen. *Data Expedition: Mapping the garment factories*. May 2013. URL: https://blog.okfn.org/2013/05/20/data-expedition-mapping-the-garment-factories/ (visited on 12/12/2019).

[Pri05] C. Price. *Fragile prosperity? Fragile social peace? Notes on the US*. July 2005. URL: http://libcom.org/library/fragile-prosperity-fragile-social-peace-collective-action-notes (visited on 12/12/2019).

[Riv19] O. Rivlin. *Reinforcement Learning for Real-World Robotics*. May 2019. URL: https://towardsdatascience.com/reinforcement-learning-for-real-world-robotics-148c81dbdcff (visited on 12/12/2019).

[RN05] F. Ricci and Q. N. Nguyen. “Critique-based mobile recommender systems.” In: *ÖGAI Journal* 24.4 (Dec. 2005), pp. 1–7.

[RT08] K. Rao and V. Talwar. “Application Domain and Functional Classification of Recommender Systems—A Survey.” In: *DESIDOC Journal of Library & Information Technology* 28 (May 2008). DOI: 10.14429/djlit.28.3.174.

[Sch+03] M. Schiessl, S. Duda, A. Thölke, and R. Fischer. “Eye tracking and its application in usability and media research.” In: *MMI-interaktiv Journal* 6 (2003), pp. 41–50.

[Udd18] J. Uddin. *Bangladesh mulls retaining China DFQF access, not APTA scheme*. Jan. 2018. URL: http://www.newagebd.net/article/32491/bangladesh-mulls-retaining-china-dfqf-access-not-apta-scheme (visited on 12/12/2019).

[XJL08] S. Xu, H. Jiang, and F. C. Lau. "Personalized Online Document, Image and Video Recommendation via Commodity Eye-tracking." In: *Proceedings of the 2008 ACM Conference on Recommender Systems*. RecSys '08. Lausanne, Switzerland: ACM, 2008, pp. 83–90. ISBN: 978-1-60558-093-7. DOI: `10.1145/1454008.1454023`.

[Zha+16] Q. Zhao, S. Chang, F. M. Harper, and J. A. Konstan. "Gaze Prediction for Recommender Systems." In: *Proceedings of the 10th ACM Conference on Recommender Systems*. RecSys '16. Boston, Massachusetts, USA: ACM, 2016, pp. 131–138. ISBN: 978-1-4503-4035-9. DOI: `10.1145/2959100.2959150`.

[ZN09] A. Zenebe and A. F. Norcio. "Representation, similarity measures and aggregation methods using fuzzy sets for content-based recommender systems." In: *Fuzzy Sets and Systems* 160.1 (2009). Theme: Aggregation Operations, pp. 76–94. ISSN: 0165-0114. DOI: `https://doi.org/10.1016/j.fss.2008.03.017`.