An experience-based method for personalized routing

Özlem Çavuş[1], Sehnaz Cenani[2], Gülen Çağdaş[3]

ORCID NO: 0000-0002-8408-1981[1], 0000-0001-8111-586X[2], 0000-0001-8853[3]

[1] Istanbul Technical University, Graduate School, Department of Informatics, Architectural Design Computing, Istanbul, Turkey
[2] Istanbul Medipol University, Graduate School of Fine Arts, Design and Architecture, Department of Architecture, İstanbul, Turkey
[3] Istanbul Technical University, Faculty of Architecture, Department of Architecture, Istanbul, Turkey

Navigation devices that are tailored to the user's preferences offer personalized routes. When multiple users are involved, it can be hard to find a route that suits everyone's preferences and avoid conflicting interests. A decision support system can improve the quality of user decisions. Traditional systems typically consider only the predefined preferences of one user or a group with similar preferences. This study aims to develop a decision support system for a group of people with diverse preferences, using a method that considers their experiences regarding time and space. The method utilizes IoT, agent-based modeling, multi-objective optimization, and crowdsourced data to create a personalized navigation system for a group, such as a family car, that considers each group member's preferences. The study uses simulation to demonstrate how this method can be applied, and it is created using Grasshopper for Rhino and add-ons. The main original contribution of this research is to show how social aspects can be incorporated into personalized navigation systems for a heterogeneous group. The major challenge was the data-sharing policies.

Received: 16.01.2023
Accepted: 16.03.2023

Corresponding Author:
cavus19@itu.edu.tr

Çavuş, Ö., Cenani, Ş., Çağdaş, G. (2023). An experience-based method for personalized routing, *JCoDe: Journal of Computational Design, 4(1),*167-191.
https://doi.org/10.53710/jcode.1236875

Keywords: Agent-based modelling, Crowdsourced Data, IoT, Multi-user, Personalized Routing.

Kişiselleştirilmiş yönlendirme için deneyime dayalı bir yöntem

Özlem Çavuş[1], Sehnaz Cenani[2], Gülen Çağdaş[3]

ORCID NO: 0000-0002-8408-1981[1], 0000-0001-8111-586X[2], 0000-0001-8853[3]

[1] İstanbul Teknik Üniversitesi, Lisansüstü Eğitim Enstitüsü, Bilişim Anabilim Dalı, Mimari Tasarımda Bilişim, İstanbul, Türkiye
[2] Istanbul Medipol Üniversitesi, Güzel Sanatlar Tasarım ve Mimarlık Fakültesi, Mimarlık Bölümü, İstanbul, Türkiye
[3] istanbul Teknik Üniversitesi, Mimarlık Fakültesi, Mimarlık Bölümü, İstanbul, Türkiye

Kullanıcının tercihlerine göre uyarlanmış navigasyon cihazları, kişiselleştirilmiş rotalar sunar. Ancak, birden çok kullanıcı söz konusu olduğunda, herkesin tercihlerine uygun bir rota bulmak ve çıkar çatışmalarından kaçınmak zor olabilir. Bu bağlamda karar destek sistemleri kullanıcıların kararlar almalarını kolaylaştırabilir. Geleneksel sistemler tipik olarak yalnızca bir kullanıcının veya benzer tercihlere sahip bir grubun önceden tanımlanmış tercihlerini dikkate alır. Bu çalışma, farklı tercihlere sahip bir kullanıcı grubu için, zaman ve mekanla ilgili deneyimlerini dikkate alan, karar destek destek sistemine dayalı bir yöntem sunar. Bu yöntem, grup üyelerinin tercihlerini dikkate alan kişiselleştirilmiş bir navigasyon sistemi oluşturmak için Nesnelerin İnterneti, etmen tabanlı modelleme, çok amaçlı optimizasyon ve kitle kaynaklı verileri kullanır. Çalışma, bu yöntemin nasıl uygulanabileceğini göstermek için Grasshopper ve Rhino kullanılarak bir simülasyon geliştirir. Bu araştırmanın orijinal katkısı, heterojen bir grup için kişiselleştirilmiş navigasyon sistemlerine sosyal yönlerin nasıl dahil edilebileceğini göstermektir. Bu çalışmanın karşılaştığı en büyük sıkıntı veri paylaşım politikalarıdır.

Teslim Tarihi: 16.01.2023
Kabul Tarihi: 16.03.2023

Sorumlu Yazar:
cavus19@itu.edu.tr

Çavuş, Ö., Cenani, Ş., Çağdaş, G. (2023). Kişiselleştirilmiş yönlendirme için deneyime dayalı bir yöntem. *JCoDe: Journal of Computational Design,4(1),*167-191. https://doi.org/10.53710/jcode.1236875

Anahtar Kelimeler: Etmen Tabanlı Modelleme, Kitle Kaynaklı Veriler, Nesnelerin İnterneti, Kişiselleştirilmiş Navigasyon.

1. INTRODUCTION

Today, mobility has started to be affected radically by the idea of auto-driving cars due to improvements in the realm of information technologies. The applicability of these autonomous systems is still in its infancy in terms of the current technology's reliability, yet even their partial implementation changes users' in-vehicle experience. In fact, many car brands have started to incorporate sensor systems detecting the stress level of users in order to switch to auto-driving mode. Connecting with the in-vehicle systems, users can care more about the surrounding. In this regard, customized tools embedded in the vehicle system, such as navigation systems producing unique content specific to each user, become important. Personalization in navigation systems is not a new topic, yet it should be reviewed regarding an experience-oriented approach. The term experience is used here to indicate the convergence of time, place, and social dependence.

"Time" refers to the change in users' mood and the change in the use of recommended places over time. Location-based suggestions are determined by the preferences of users, and these preferences are tied to the mood of users. Therefore, user profiles and especially preferences should be reviewed in terms of the current mood of users. To illustrate, a user may like to visit a coffee shop just before arriving at work, yet he/she may not prefer drinking coffee when being nervous. The same place can also hold different activities such as yoga classes and live concerts in different periods. Therefore, a recommended place is beyond being a single location. It is the convergence of time, activity, and space. Therefore, the term "place" used in this paper stresses the functional use of a space in time. The third point, which is called "social dependence," indicates the dependency of decision-making based on our experience within a social group, such as the existence of a person or a situation that has priority. The presence of other people (the plurality of users) will have a voice in determining our preferences. For instance, having a priority person or a situation in a car (dominated agents) such as a baby, a patient, and a sleeping person affects preferences. In this regard, this paper discusses decision making in determining preferences from the social level because making decisions is also affected based on our personal experiences in the social environment in which we stand. The coexistence of above-mentioned three points is neglected in the existing routing systems. However, these points directly affect how we experience and therefore affect our decision making in defining preferences. In detecting common preferences, however, it may be difficult to get a consensus in a heterogeneous user group even if they know each other such as families. Because preferences might conflict with each other, a decision support system (DSS) for multi-user serves as a valuable start to improve quality of decisions in personalized routing devices. Therefore, the focus of this research is on the DSS concerning an experience-based approach in personalized routing for a heterogeneous group.

In collecting data in the proposed DSS, user moods are expected to be determined via in-vehicle sensors, and decision making takes consideration into the priority of agents. In defining common preference(s), multi-objective optimization is performed, and non-dominated preference(s) become keywords to search locations on social media because crowdsourcing offers up-to-date feedbacks increasing the variety of the content for which users seek. The pictures or tweets have geographic positioning information where they are taken, so they provide sources to determine routing nodes.

This paper discusses the relevant background to reveal the original contribution of this study, which introduces a DSS. The DSS depends on the Internet of Things (IoT) and agent-based modelling (ABM), and a relevant case study exemplifies the applicability of the method over a family car with a father, a mother, and a child. The results show that evaluation of personalized routing over social aspects opens the doors towards discoveries to improve digital tools. The major problem which is faced is reaching labelled data without access limitations. In consequence, the significance of the study is discussed together with the limitations and recommendations for future studies.

2. BACKGROUND

The relevant studies in the existing literature are examined in relation to personalized routing, crowdsourced data, simulation of human behaviour, multi-objective optimization, and recommender systems, respectively.

Sha et al. (2013) offer a system based on crowdsourcing. The system advocated by the author integrates driver-provided information into a vehicle navigation system in order to calculate personalized routing. It allows drivers to register into certain vehicle social network groups in order to share driving experiences with other drivers using voice tweets. A driver can instruct the social navigator to avoid or choose specific road segments in generating personalized routing. Also, Majid et al. (2013) evaluate personalized routing but based on geotagged pictures in social media to offer tourist locations on landmarks in a city. Similarly, Mermelstein (2017) evaluates a personalized navigation system depending on crowdsourcing. The author describes crowdsourced data collection as a participatory method of building a dataset with a large group of people. According to the author, navigation products and services provide the same content to everyone, present with the same navigation platform as everyone else, and a user must manually manipulate a proposed route to customize the route according to user preferences. Mermelstein (2017) suggests a need for a system allowing the crowdsourced collection of location-based data that can be used to update and maintain location-based data supplied to a navigation service that does not require direct input from the network of users to gather the updated data. There is also a need to provide customized navigational services that allow a user's preferences to affect the route provided by the service. Likewise, Wan et al. (2018) offer a method based on Bayes and Knn using geotagged Flicker pictures. The authors claim that the results of their study are better in accuracy than context-aware pattern recommendations.

Studies mentioned above offer unique content specific to a user or an event. However, they neglect one of the most significant concerns: human behaviour under different periods in a social context. Huang et al. (2014) offer a method for routing incorporating responses of people to the environment. The authors represent how these responses can be modeled and collected through two types of questionnaires. Video and map tasks are given to participants, and it is discussed how to improve automatic route planning based on their responses.

Similarly, Sopher et al. (2016) focus on simulation of human behaviour in built environments using an actor profiling method. This method is significant in discussing the translation of how social values in a digital environment. The convergence of social and numerical aspects is achieved by defining an event-based model defined over space, activity, and actors.

Another critical point in designing a recommender system is to find the optimal solution values of more than one desired goal. As it is intended to create a recommender system for multi-user, we have multi-criteria that should be achieved simultaneously in an optimal manner. In literature, Multi-objective Optimization (MOO) is preferred when problems have more than one objective. MOO has a multi-dimensional space of the objective function vector and the decision variable space of the solution vector. In every x solution in the decision variable space, there is a point on the objective function space (Gunantara, 2018). The following **Equation 1** shows the mathematical representation of MOO. Accordingly, x is a solution, n is the number of objective functions, U is a feasible set, *fn (x)* is the nth objective function, and min/max is combined object operations (Gunantara, 2018).

$$\text{Subject to: } x \in U; \quad \min/\max f1(x), f2(x), \ldots, fn(x) \qquad (1)$$

It has two main methods as scalarization and Pareto. The former creates multi-objective functions made into a single solution using weights, while the latter has a dominated solution and a non-dominated solution obtained by a continuously updated algorithm. According to Gunantara (2018), several reviews have been done regarding the methods and application of MOO, but these two methods do not require complicated mathematical equations making the problem simple in this manner. Zheng and Liao (2019) benefit from the Pareto method in order for heterogeneous tourist groups to design personalized tour routes. According to the authors, tourism activities are typically group-oriented, and the preferences and goals of group members may completely differ or even conflict with one another. Therefore, Pareto optimality is adopted to be used for the design with a nondominated sorting heuristic approach.

Another study in the field of recommendation systems for multi-user is conducted by Quan and Cho (2014). The authors offer a hybrid system depending on the analytic hierarchy process (AHP) and Bayesian networks for smart TV. Bayesian networks are used to infer each user's genre preference as well as program preference, while AHP is used to predict group genre preference and choose recommended programs. For instance, Kengpol et al. (2008) developed a DSS model integrating

AHP to achieve maximum satisfaction by considering the priorities of both customers and operators. The presented approach minimizes transportation costs while maximizing satisfaction in distributing goods between the distribution center and the customer. As the users, products, and spatial environment of the urban delivery models have huge variations, these models also need to solve similar problems, such as the existence of multi-user scenarios, varying priorities like primary users, sensitive cargo, and dynamic delivery addresses. According to Palanca et al. (2021), traditional vehicle fleets are transforming into more open fleets where members can decide whether or not to be part of the fleet and perform certain services. This distributed decision-making process makes management and control of open fleets complex, necessitating simulation tools.

As stated above, personalized routing in existing studies is studied mainly in the tourism sector and in-vehicle systems. In the tourism sector, actors are tourists who visit new places. That is, the environment is new to them. In this research, we recommend places not only in a new environment but also in a familiar environment. On the other hand, studies regarding in-vehicle personalized routing are conducted for a single user, and they focus on the development of the navigation system.

Moreover, preferences are set at once; still, they may change according to the conditions, such as the moods of people. In addition, suggestions have remained only as locations; however, a place can function differently in different periods. For example, a café may also serve as a concert hall on weekend evenings. In this regard, social media becomes significant to take up-to-date data based on experiences in order to tackle such problems. Therefore, this paper differs from the existing studies on personalized routing in encapsulating the following considerations at once.

- This study considers social aspects in defining agents in the decision support system.
- A place is defined, indicating its functional use in time rather than defining only a location.
- In-vehicle systems are studied for a heterogeneous group rather than for a single user or a homogeneous group.
- The preferences of users are associated with their current moods in time.

- It allows exploring places users have missed on their daily routes in which they are familiar.

The above-listed aspects are the significant points outlining the contribution of this paper to the relevant literature. The flow of the research method is evaluated based on these points.

3. METHODOLOGY

The main motive in outlining the research method in obtaining required data comprises optimization for multi-objective and processing crowdsourced data.

The former depends on the Pareto method to optimize preferences of a heterogeneous group of people in order to obtain a non-dominated preference. This preference will become a keyword to search into social media. The Pareto method is selected, each user has his/her different evaluation criteria in determining weights. For instance, two users can differently vote the same preference that they like almost equally. Since the Pareto method focuses on non-dominated preferences, it becomes useful not to eliminate a person who gives lower weights.

The latter uses content-based filtering from crowdsourcing. The determined non-dominated preference is searched within geotagged places shared in social media to obtain up-to-date data. In this manner, locations are associated with its current usage. Then, these obtained data become inputs to determine the nodes to be linked for routing.

Based on these two major points, it is proposed a flow having three main parts (**Figure 1**) as obtaining data from users in the car (in-car inputs), obtaining data from location-based services (environmental inputs), and processing all data to create a personalized route. In the first part, users determine their preferences in the user interface of a mobile application. All preferences are then voted by users in the range from 1 to 9. In case there are missing points, matrix factorization is used to estimate these missing parts. Then, MOO is performed for the non-dominated solution to further evaluate for crowdsourcing. MOO is associated with the moods of users who are present in the car. The priority in decision making changes according to this presence. The number of users determines which type of Pareto methods such as 2D or 3D Pareto should be implemented. If there is only a single user, there is no need to use Pareto optimality. In the second part, environmental data are obtained from social media by searching non-dominated preferences. Places becoming a point of interest (POI) are determined for processing. Finally, personalized routing is created based on Google Maps.

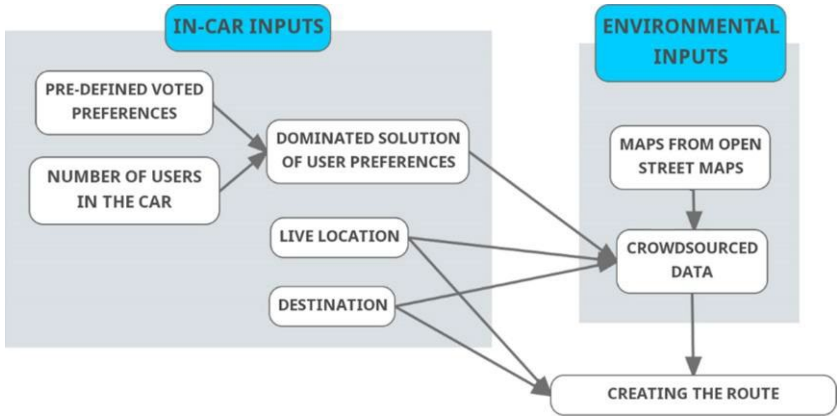

Figure 1: Conceptual scheme for the framework.

The presented scheme in **Figure 1** is extended in detail under two titles as setting up user profile and processing. The former defines the necessary information that should be acquired from users to initiate the recommendation system. The latter, on the other hand, defines how processing behind the system works. These two titles are explained, respectively.

3.1. Setting up user profile

The user setup starts with creating a personal identity comprising username, age, profession, and sex. It is required for collaborative filtering if it is needed. Once the profile is set, users and situations having priority, such as a child and a sleeping person, are defined as dominated agents, and their preferences are determined separately from the preferences of other users. Except for the dominant agents, all preferences are determined based on different moods, such as being tired, energetic, and stressed. All collected preferences obtained from all users sharing the same car are ready to be voted by each individual. In case users do not vote for all preferences, collaborative filtering based on matrix factorization (Ng, n.d.) predicts missing ratings. The logic behind this technique is to recommend products based on a similar customer rather than similar content. It is a machine learning technique where recommended things depend on patterns the machine has observed in other people whose preferences are similar to those of the user.

Figure 2: Flowchart indicating setup of user interface.

The reason to offer matrix factorization based on collaborative filtering is to reduce space in storing data and work with unlabelled data. For instance, using this technique, the system holds 300 thousand features instead of 2 million features when 2000 users and 1000 entries. As it is not easy to obtain labelled data, this technique also estimates latent features hidden in the preference data. The matrices of these feature data, which are determined randomly, are multiplied to estimate preferences. These estimated preferences are compared to the existing preference data, which are already voted. Based on this comparison, the error function of the test data is calculated, and the training data set is reformulated. The iterations continue until the error is equal to almost zero. This process is called stochastic gradient descent (Zhu et al., 2017), used for finding the right factorization.

All the preferences are expected to be weighted by using matrix factorization, yet the system may face a cold start problem indicating the circumstances which are not yet optimal for the engine to provide the best possible results. For instance, the training data set for machine learning might not be good enough to work in the desired manner. Hence, the proposed flow asks users whether they are willing to share their preferences with the recommender system to estimate better results and cope with the cold start problem. If they accept sharing their data, users' accessibility settings become open to the recommender system without exposing their identity to anyone else. All these mentioned steps outline user profile settings, which are depicted in the flowchart in **Figure 2**.

3.2. Processing

Once user profile settings are accomplished, processing behind the recommender system start whenever it perceives someone using the car. This presence of the users is determined by checking the GPS of the user's mobile phone. Whenever a user interacts with the car, sensors detect the current state of users' moods, and the system evaluates corresponding preferences based on the Pareto method to obtain non-dominated preference(s). Based on the number of users, the MOO technique changes slightly. The essence, however, is the same. If there is only one person, the system only concern with the pre-defined and already voted settings. In case settings have not been weighted, the system estimates weights based on similar users' patterns by using the matrix factorization technique.

Figure 3: Flowchart indicating processing behind personalized routing.

Once all ratings are determined, the number of users defines options to proceed as indicated in the following statements.

- A: the process for a single person. In this case, individual preferences are selected without using Pareto.
- B: the process for more than one person based on Pareto optimality.

For instance, if there are two people in the car, the 2D Pareto method is used. Similarly, the 3D Pareto method is used for three people in the car. 2D and 3D respectively mean two and three objectives that are tried to be achieved simultaneously. In this research, objectives indicate the preference of each user. The logic of Pareto optimization depends on the domination of alternatives when compared to each other. In this sense, preferences are optimized, and remaining results are ready to be searched as labels among the crowd-sourced data.

Non-dominated preference is indeed a keyword to be searched among relevant contents. This content-based filtering obtains a point cloud indicating locations where there is similar content. The point cloud already highlights the concentration areas whose locations serve as anchor points. Connecting these points, which are the nodes, the routing is defined. However, the location of the points should be listed, and the list should be sorted to cull the pattern. Otherwise, there would be a noisy point cloud causing difficulty in defining a precise route. Once the list is ready, routing is generated based on the live location and the destination point. Users can specify whether they desire to avoid certain roads such as highways. These roads are eliminated from the list in route generation.

The system needs labelled data of the roads to initiate routing. If the model is processing a large amount of data, it may take longer to respond. Data can be obtained from open-source platforms, such as Google Maps. There is also available software in the market allowing any user to search for crowdsourcing such as Mosquito for Grasshopper. All these mentioned steps are shown in the flowchart in **Figure 3** that will perform personalized routing for multi-user.

4. CASE STUDY

The case study's scope is to create a simulation to exemplify how the proposed flow of DSS works over a family car for three people: a father, a mother, and a child. The reason for selecting a family as a sample target user group is that a family has a social role diversity. A social role indicates the roles which individuals play in a society, such as being a father.

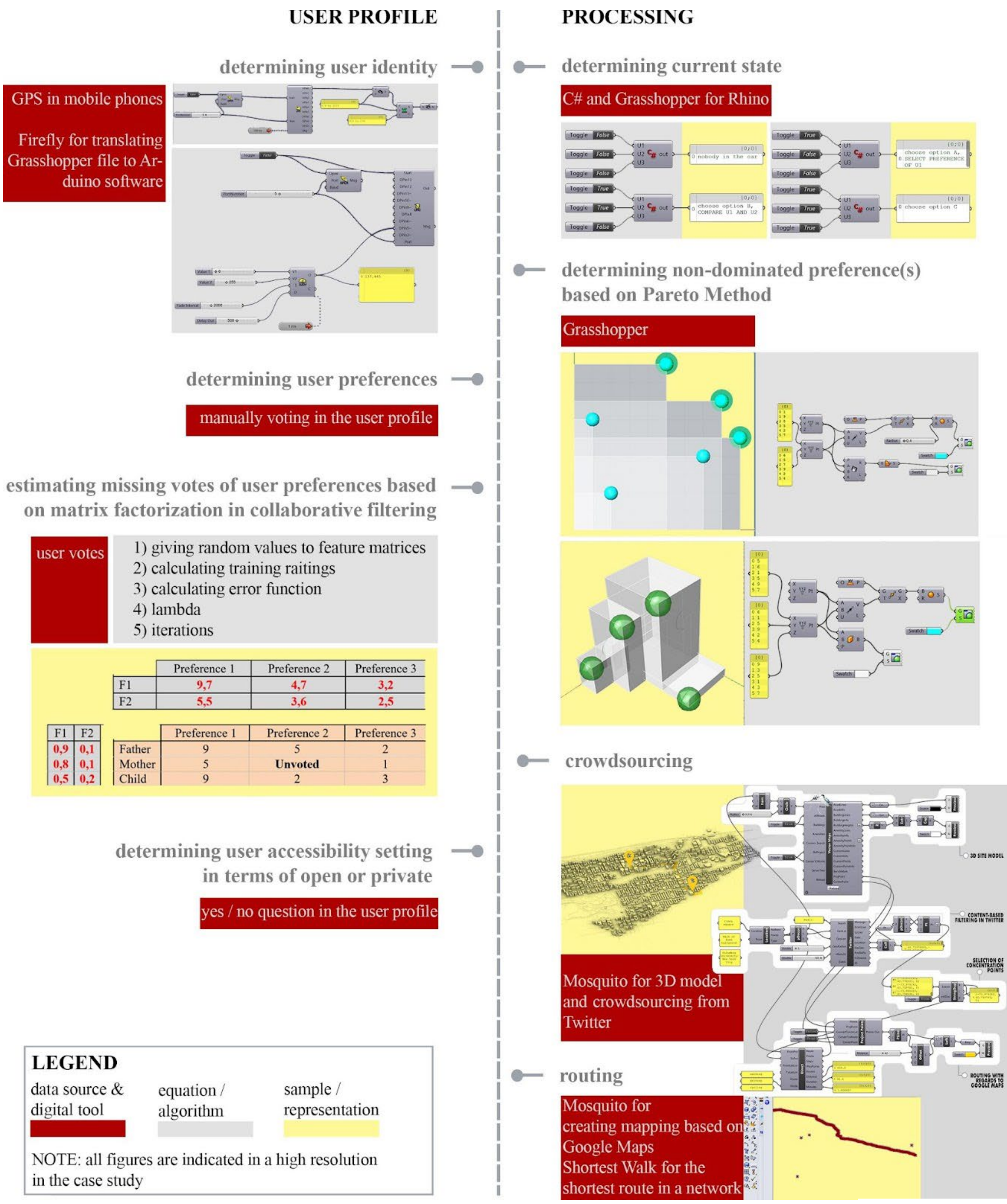

Figure 4: The link between DSS and the case study.

The simulation model is generated concerning the flowcharts (**Figures 2** and **3**). **Figure 4** indicates the relation between key parts in these flowcharts and the case study in regard to the data source, digital tool, equation or algorithm, and representation. The images in the figure are then explained in detail.

Since Rhino and its Grasshopper add-on are widely used in architecture and design, particularly for creating complex 3D models and parametric designs, they are used for simulation in this research. One potential use of these programs in personalized routing is for creating customized routes that consider user-specific factors, such as preferred modes of transportation, specific destinations, time constraints, and other relevant data.

Simulation in Grasshopper depends on the addons as Mosquito 0.5a (Smuts, n.d.), Firefly (Payne, 2016), Shortest Walk (URL1)), and C# script embedded in Grasshopper. The functions of these addons are briefly described as listed below.

- Mosquito for crowdsourcing from Twitter and creating mapping based on Google Maps.
- Shortest Walk (URL1) for Grasshopper to calculate the shortest path between two points in a graph.
- Firefly for translating Grasshopper file to Arduino software.
- C# script for calculation and determining options for Pareto optimality.

The simulation starts with the detection of existing users (**Figure 5**) in order to activate preferences accordingly. In case preferences have not been voted yet, missing parts are tried to be estimated via matrix factorization and stochastic gradient descent. **Table 1** exemplifies preference data (table with orange in colour) and feature data (table with grey in colour). *F1* and *F2* are the latent features indicating values in red. These values are the randomly determined ratings to initiate stochastic matrix factorization.

Table 1: Implementation of the matrix factorization: giving random values to the feature data.

		Preference 1	Preference 2	Preference 3
	F1	9,7	4,7	3,2
	F2	5,5	3,6	2,5

F1	F2			Preference 1	Preference 2	Preference 3
0,9	0,1		Father	9	5	2
0,8	0,1		Mother	5	Unvoted	1
0,5	0,2		Child	9	2	3

Based on the given random values, training ratings are calculated as indicated in **Table 2** using the **Equation 2**. It is calculated as a result of the multiplication of the feature matrices, denoted by *U* and *V* (Zhu et al., 2017).

*Training ratings= U*V* (2)

	Preference 1	Preference 2	Preference 3
Father	(0,9*9,7) + (0,1*5,5) = 9,28	(0,9*4,7) + (0,1*3,6) = 4,59	(0,9*3,2) + (0,1*2,5) = 3,13
Mother	(0,8*9,7) + (0,1*5,5) = 8,31	(0,8*4,7) + (0,1*3,6) = 4,12	(0,8*3,2) + (0,1*2,5) = 2,81
Child	(0,5*9,7) + (0,2*5,5) = 5,95	(0,5*4,7) + (0,2*3,6) = 3,07	(0,5*3,2) + (0,2*2,5) = 2,1

Table 2: Implementation of the matrix factorization: training ratings.

Depending on training ratings, the error's cost function is calculated using the **Equation 3**, where *x* and *y* denote predicted and actual ratings, respectively (Ng, n.d.).

Error=12mi=1m(xi-yi)2 (3)

Error=118((9-9,28)2+(5-4,59)2+(2-3,13)2+ (5-8,31)2+ (0-4,12)2+ (1-2,81)2+ (9-5,95)2+ (2-3,07)2+ (3-2,1)2)

Iterations perform until the value of error is equal to zero. For instance, the first column should decrease while the first row of the second column should increase. The unvoted preference is then estimated depending on the last training ratings, as exemplified above.

Once weights are predicted, the simulation starts according to the number of users. The model runs in C# Script in determining the options

(**Figure 5**). *U* values such as *U1* indicate users. Boolean toggle represents the presence of users in the car.

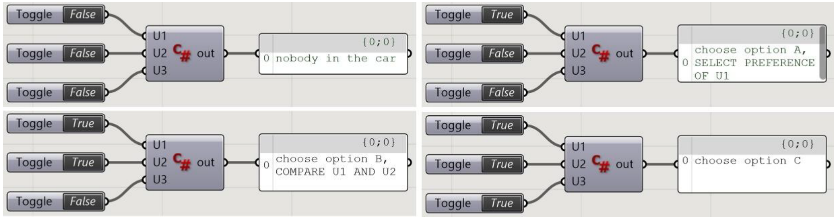

Figure 5: C# script for determining the options.

Then, the number of users in the car is set, and non-dominated preferences are obtained. In this case, both 2D and 3D Pareto method are exemplified, as illustrated in **Figure 6**. In the 2D Pareto chart in **Figure 6**, there are three dominated and three non-dominated alternatives. On the other hand, there are two dominated and four non-dominated alternatives in the 3D Pareto chart. These non-dominated alternatives are then searched in Mosquito, receiving data from Twitter. Mosquito works with a base map with labelled data in conjunction with Google Maps.

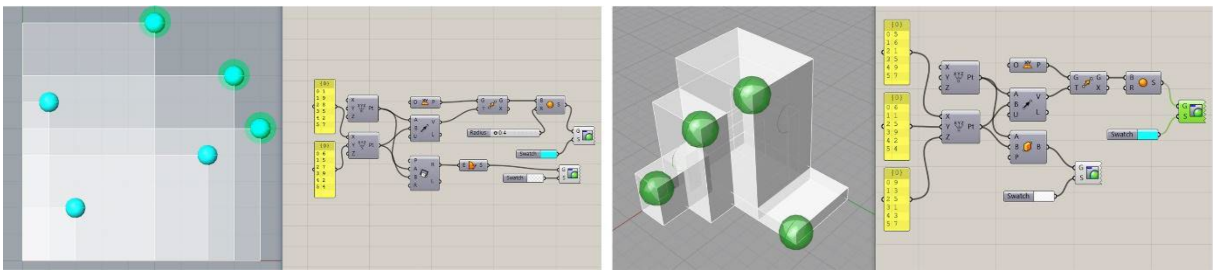

Figure 6: 2D (left) and 3D (right) Pareto chart.

Figure 7 indicates the algorithm in Grasshopper comprising the site model, crowdsourcing, and routing. It responds to any location within any range. Any keyword regarding a location turns into a specific coordinate through transforming Google's API into an address and point. Users can define the boundary lines of the site model. The data taken from Google Maps are converted into a 3D site model in this way. Within this range, any keyword can be searched. Accordingly, tagged tweets are listed based on location, the author's name, and the date when the tweets are published. However, Twitter allows to share a certain number of data, and some tweets do not include geographic information. Hence, the list of the location of tweets is culled. Among the remaining list, the duplicated locations are selected as concentration points, which serve as nodes for the routing path. Mosquito Direct command under Mosquito toolbar creates routing based on the destination, live location of users, transportation type such as driving and walking (**Figure 7**). However, the centre points of the 3D model and generated path may not be matched, so these models are juxtaposed via Map Projection. The coordinates coming out of Twitter are reversed in the Northern hemisphere, so X and Y coordinates are replaced.

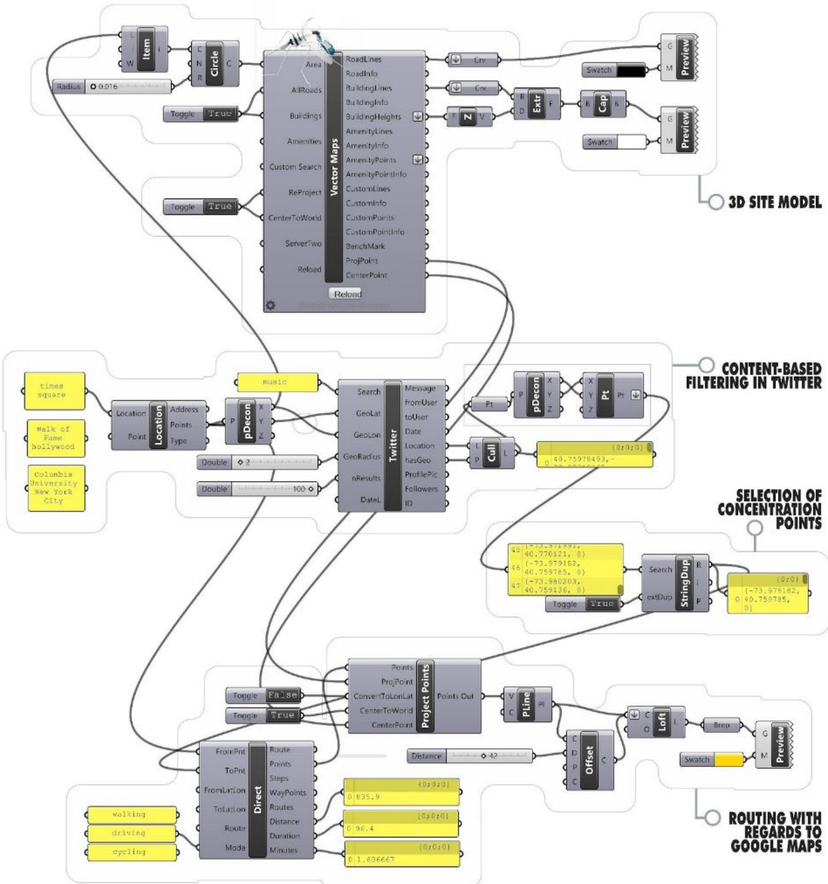

Figure 7: Algorithm in Grasshopper.

Figure 8 exemplifies the creation of routings in different locations with different scales in Grasshopper according to the definition in **Figure 7**. It is intended to select an area in the city centre where there are both business centres and tourist attractions.

Figure 8: 3 case studies created based on the algorithm in Figure 7.

The simulation also exemplifies automatic detection of moods of users, which is developed in Grasshopper and Arduino. **Figure 9** shows the codes translating languages between Grasshopper file and Arduino via Firefly addon to receive inputs from the environment in order to initiate the system.

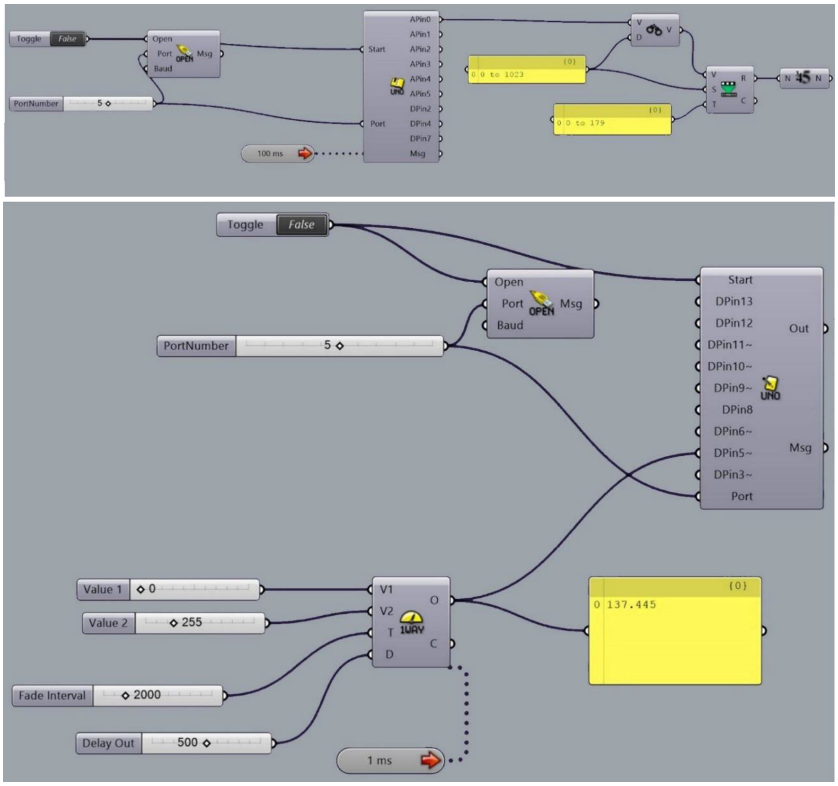

Figure 9: Top: Code for Arduino Uno Read via Firefly. Bottom: Screen shot of the code for Arduino Uno Write via Firefly.

5. DISCUSSİON AND CONCLUSION

The simulations in **Figure 9** are recreated using calculating the shortest path among initial and destination points (**Figure 10**). As seen, the paths in **Figure 10** are slightly different from the previous simulation in **Figure 9** in terms of the line lengths. This study's primary emphasis is to show how social values can be integrated into the computational design domain. The algorithm in **Figure 7** responds to any location with a user-defined boundary size. The generated GIS-based site maps in the simulations can be directly used in architectural projects as well. It is quite beneficial, especially in the analysis of the user flow in the project site.

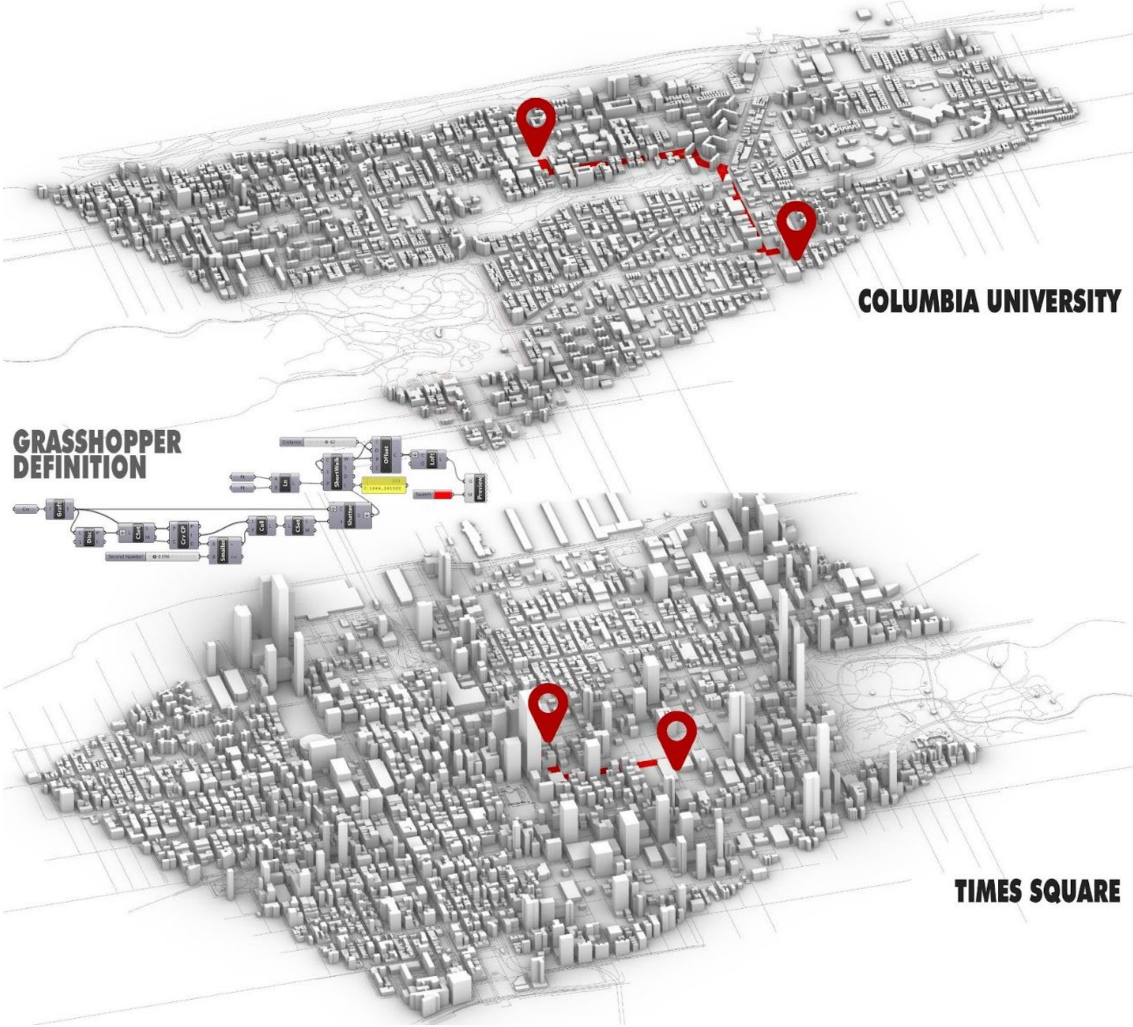

Figure 10: The shortest path algorithm.

The presented method for a decision support system is intended to use for location-based recommendations to increase the quality of decisions users take. The recommendations consider locations concerning functionality and time. The idea behind this model is significant for the literature, as it demonstrates how experiences may improve decision-making in DSS. It shows how social aspects can play a significant role in the computational design environment of personalized routing. These two grounds of the design, as social and computational, should be in dialogue with each other. This study shows the integration of these two sides and gives clues for the applicability of the proposed system. The results also show the dependency values for its applicability in real-life scenarios. Implementation of this system closely ties to how and where labelled data are obtained, how noises are removed, and how limited access of non-copyrighted images or tweets are overcome. It is also significant to get legal permission from the authorities to use crowdsourced data for commercial purposes.

The core idea advocated in this study can be applied to in-vehicle infotainment systems encapsulating personalized recommendations during the journey, such as music, news, and even games. It also offers location-based recommendations. Besides, the proposed method has the potential to adapt to other means such as cycling in addition to driving. The method served by this research also gives clues for similar studies in different fields such as e-commerce and tourism. The presented method uses both content-based and collaborative filtering because the use of content filtering alone could be simplistic and not very accurate. Still, the method needs to be improved to function as a whole integrated system.

The study proposed in this paper has limitations. First of all, the simulation is up to three users. It should be extended in terms of MOO based on objectives more than three. Also, the training data set should be improved to cope with the cold start problem. Automated determination of the preferences can be determined using K-Nearest Neighbours (KNN) algorithm if data are labelled. KNN is a supervised machine learning algorithm that can be used to solve both classification and regression problems. The labelled input data makes the algorithm learn a function that produces an appropriate output when given new unlabelled data. It works with the relative distance to its neighbours in order to classify the input data. The recommender based on KNN offers the closest contents to users themselves. In addition to the limitation regarding the autonomous determination of preferences, there is a child mode limitation. Unless a person has any mobile phone like a child, the system cannot perceive whether it is in the car due to lack of GPS. It needs to obtain data from multiple sources. It requires the connectivity of electronic tools. The connectivity of the tools not only solves the child mode problem but also increases the reliability of the proposed system. Indeed, it can serve as a control mechanism. On the other hand, the child mode problem can be solved by a weight sensor in the vehicle seats, as it is used to understand whether the seat is full or empty. If there is no GPS signal, but the weight sensor says that the seat is full, the system can accept as if there is a baby or a child. This sort of control mechanism can be added to the decision support system. In addition to the child mode problem, there is the access limitation to labelled data in crowdsourcing, as stated above. Legal procedures should also be considered in case the proposed method turns into a commercial navigation device.

In addition, the response speed of a model should be tailored to the specific processing needs and limitations of the data being processed to provide more efficient and effective results. The model should be optimized to prioritize processing the most important data first to provide a quick response time for critical tasks. It is also important for the model's scalability, and this can be achieved through distributed computing or cloud-based services providing additional processing power and memory resources.

In future works, it is intended to evaluate this research to make a complete infotainment system for multi-user, either it is served for in-vehicle devices or domestic usage such as a TV recommender system. In these systems, the design is generally for a single user or a homogeneous group of people without considering the state of being a person with priority or the change in people's mood. The next research aims to allow the union of user-centric and user-participated approaches in its design. User feedbacks are beneficial to evaluate the system in this regard. The logic behind feedback evaluation depends on deep reinforcement learning. It is a sort of machine learning technique working based on the analysis of a current state and taking an action that maximizes a future reward through continuous interaction with the environment. Last but not least, it is intended to improve the system in a manner that it can be easier to use and autonomous to enhance the lifecycle of the proposed system. Easy to use scenarios is significant, as it makes closer to reach a mainstream usage. Therefore, a user-friendly interface can be developed in this sense. All these discussed issues would carry the system one step further if they are implemented appropriately.

Acknowledgement

This study is developed based on the term project in Decision Support Systems in Architectural Design course given in Architectural Design Computing Graduate Program at Istanbul Technical University.

References

Gunantara, N. (2018). A review of multi-objective optimization: Methods and its applications. Cogent Engineering, 5(1). https://doi.org/10.1080/23311916.2018.1502242

Huang, H., Klettner, S., Schmidt, M., Gartner, G., Leitinger, S., Wagner, A., & Steinmann, R. (2014). AffectRoute – considering people's affective responses to environments for enhancing route-planning services. International Journal of Geographical Information Science, 28(12), 2456-2473. https://doi.org/10.1080/13658816.2014.931585

Kengpol, A. (2008). Design of a decision support system to evaluate logistics distribution network in Greater Mekong Subregion Countries. *International Journal of Production Economics, 115*(2), 388-399.

Liu, T. K., Moskowitz, P. A., Greenwood, M. C., Lieberman, L. I., & Wood, D. A. (2002). System for personalized mobile navigation information. U.S. Patent No. 6,349,257. Washington, DC: U.S. Patent and Trademark Office.

Majid, A., Chen, L., Chen, G., Mirza, H. T., Hussain, I., & Woodward, J. (2013). A context-aware personalized travel recommendation system based on geotagged social media data mining. International Journal of Geographical Information Science, 27(4), 662-684. https://doi.org/10.1080/13658816.2012.696649

Mermelstein, Y. Z. (2017). Method and system for providing personalized navigation services and crowd-sourced location-based data. U.S. Patent Application No. 15/187,400.

Ng, A. (n.d.). Machine learning. Coursera. https://www.coursera.org/learn/machine-learning

Palanca, J., Terrasa, A., Rodriguez, S., Carrascosa, C., & Julian, V. (2021). An agent-based simulation framework for the study of urban delivery. *Neurocomputing, 423,* 679-688.

Payne, A. (2016, October 25). Firefly. Food4Rhino. https://www.food4rhino.com/app/firefly

Quan, J. C., & Cho, S. B. (2014, June). A hybrid recommender system based on AHP that awares contexts with Bayesian networks for smart TV. In International Conference on Hybrid Artificial Intelligence Systems (pp. 527-536). Springer, Cham.

Sha, W., Kwak, D., Nath, B., & Iftode, L. (2013, February). Social vehicle navigation: integrating shared driving experience into vehicle navigation. In Proceedings of the 14th workshop on mobile computing systems and applications (pp. 1-6).

Smuts, C. (n.d.). Mosquito – Synthetic spaces. Synthetic Spaces-Conceptual Explorations of the Evolving Dimenions- Architecture, Industrial Design, Furniture. https://www.synthetic.space/synthetic/2443/

Sopher, H., Schaumann, D., & Kalay, Y. E. (2016). Simulating Human Behavior in (Un) Built Environments: Using an Actor Profiling Method. International Journal of Computer, Electrical, Automation and Information Engineering, 10(12), 2030-2040.

Wan, L., Hong, Y., Huang, Z., Peng, X., & Li, R. (2018). A hybrid ensemble learning method for tourist route recommendations based on geo-tagged social networks. International Journal of Geographical Information Science, 32(11), 2225-2246. https://doi.org/10.1080/13658816.2018.1458988

Zheng, W., & Liao, Z. (2019). Using a heuristic approach to design personalized tour routes for heterogeneous tourist groups. Tourism Management, 72, 313-325. https://doi.org/10.1016/j.tourman.2018.12.013

Zhu, X., Hao, R., Chi, H., & Du, X. (2017). FineRoute: Personalized and time-aware route recommendation based on check-ins. IEEE Transactions on Vehicular Technology, 66(11), 10461-10469. https://doi.org/10.1109/tvt.2017.2764999

URL 1 < https://www.food4rhino.com/app/shortest-walk-gh >