Inferring Landmarks for Pedestrian Navigation from Mobile Eye-Tracking Data and Google Street View

Christian Lander
German Research Center for Artificial Intelligence
Saarland Informatics Campus
christian.lander@dfki.de

Frederik Wiehr
German Research Center for Artificial Intelligence
Saarland Informatics Campus
frederik.wiehr@dfki.de

Nico Herbig
German Research Center for Artificial Intelligence
Saarland Informatics Campus
nico.herbig@dfki.de

Antonio Krüger
German Research Center for Artificial Intelligence
Saarland Informatics Campus
krueger@dfki.de

Markus Löchtefeld
Aalborg University
Aalborg, Denmark
mloc@create.aau.dk

CHI'17 Extended Abstracts, May 06-11, 2017, Denver, CO, USA.
ACM 978-1-4503-4656-6/17/05.
http://dx.doi.org/10.1145/3027063.3053201

Abstract
While it has been well established that incorporating landmarks into route descriptions enhances understanding and performance of wayfinding, only a very few available systems make use of them. This is primarily due to the fact that landmark data is often not available, and the creation of the data is connected to tedious manual labor. Prior work explored crowd-sourced approaches to collect landmark data, but most of that work focused on explicit user input to gather the data. In this paper, we presented our work towards a system to automatically infer suitable landmarks for pedestrian navigation instructions from mobile eye-tracking data. By matching the video feed of the scene camera of a head-mounted eye tracker to Google Street View imagery, our system is able to cluster the visual attention of the users and extract suitable landmarks from it. We present early results of a field study conducted with six participants to highlight the potential of our approach.

Author Keywords
Landmarks, Eye Tracking, Google Street View, Navigation

ACM Classification Keywords
H.5.m [Information interfaces and presentation (e.g., HCI)]: Miscellaneous

Introduction

Navigating in partially familiar or completely unfamiliar environments is a complex task that requires spatial reasoning, memorization, and close examination of the surroundings. While turn-by-turn navigation can help in this process, its generic route descriptions, dependence on positional accuracy, existing experiences, and wrong human distance estimations can mislead users [2].

Incorporating landmarks – geographic objects that help structuring human mental representations of space [20, 13] – in the route description has been proven to enhance understanding and performance of wayfinding [3, 23]. May et al. found that landmarks, namely pubs, specific shops, restaurants, supermarkets, petrol stations, traffic lights and parks should be the primary means of providing directions to pedestrians [14]. Even using more specialized You-Are-Here maps that often contain such landmarks can support the process, compared to the one-size-fits-all approach of current online map providers [12, 21].

Even though these advantages are well known, very few commercial systems exist that include landmarks in the description process. The primary reason for that is the lack of available landmark data [3]. The main reason for this is, that the process of acquiring such data is very costly and often connected to manual labour, e.g. for detecting direct visibility of tall buildings from an intersection. Wakamiya et al. relied on a 3D geographic model of the city [24]. The thresholds for their proposed algorithm were determined by examining the actual visibility of a landmark in Google Street View.

To overcome the need for manual acquisition of landmark data, a variety of different approaches have been explored. Crowd-sourcing in particular has been proven to be a practical approach. Wolfensberger and Richter developed a mobile application that allows one to label objects in the environment as landmarks in-situ [26]. Helgath et al. extended this approach to a smartwatch application that was controllable via speech input [5] to lower the complexity of the interaction. Introducing gamification as a means of motivation for the crowd has been explored as well [1], and adapted to in-car applications for passengers [11].

But all of these approaches require active user input and only a few automatic selection techniques exist that correspond well enough with the human concept of landmarks [16] so far. Furthermore, the selection of landmarks is biased, when using a specific source such as social networks, or restricted by the available data for certain characteristics of objects, and they cannot be chosen freely based on their saliency. One possibility to overcome this need is to use eye tracking. In the area of geographic exploration, eye-tracking has been employed successfully before, i.e., to identify factors that influence the duration of the visual exploration of a city panorama [7]. Furthermore, eye tracking has been used to understand the process of self-localization with a map [8] and Giannopoulos et al. even developed a navigation system that incorporates the user's gaze at decision points to communicate the route [4]. This prior work makes eye tracking a promising approach to automatically identify landmarks as well. Since landmarks are normally characterized by a high visual saliency, they should attract the visual attention of the user [18, 25].

In this paper, we presented our work towards a system to automatically infer suited landmarks for pedestrian navigation instructions from mobile eye-tracking data. By matching the video feed of the eye tracker's scene camera to Google Street View imagery, our system is able to cluster the visual attention of the users on specific elements of the environment. From this aggregation, we can infer the saliency of

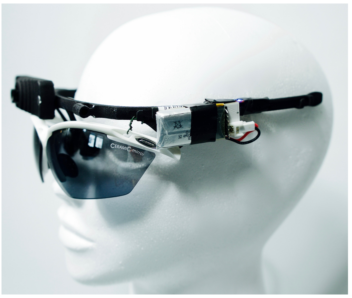

Figure 1: Eye tracking prototype extended with sunglasses and a wireless inertial measurement unit to detect the head orientation.

the environmental elements and the potential for use as landmarks for navigational instructions. Besides our current implementation, we present the early results of a field study with six participants.

Enabling Automatic Landmark Extraction

Sorrows and Hirtle distinguish between three kinds of landmarks: visual, semantic and structural landmarks [22]. Visual landmarks are salient due to their visual prominence, semantic landmarks due their historical or functional importance. Structural landmarks are characterized by the importance of their role in the environment. In this work we focus on extracting visual landmarks using eye-tracking data. By using natural gaze patterns of pedestrians, we want to overcome the need for manual input when crowd-sourcing landmark data. Due to the visual saliency of the landmarks, the users implicitly will focus more visual attention on them [25]. The described method refers to the visual attraction as a measure for the attractiveness of landmarks in the formal model of landmark saliency by Raubal and Winter [19].

The automatic extraction of visual landmarks based on a person's viewing and gazing behavior faces two key challenges. First of all, we have to continuously record a person's location, head orientation and gaze data. All the data have to be synchronized to accurately calculate the user's attention. Secondly, we have to map the user's visual attention onto the environment in an automated fashion. Especially the latter used to be problematic in the past due to the lack of holistic imagery of the environment. In our approach, we combine the recorded data with information gathered through Google Maps and Street View APIs [1].

Specifically, we correlate the user's GPS location and head orientation together with the eye tracker's scene camera video stream based on the recorded timestamps. The location data was captured through an iPhone SE that was manually synchronized by capturing a start-button press on the phone's screen with the scene camera. The location and head orientation data is used to get potential candidates the user's current field of view. This is done by querying Google Street View image data representing the current environment. To finally map the user's gaze onto the surrounding area, we make use of *GazeProjector* [9]. While this system was originally used to map the gaze point of a user on public displays, we adapted the approach to map the gaze on Google Street View imagery. More precisely, the eye tracker's scene camera image is used as a template that is searched for in the corresponding Google Street View image. Therefore, we re-implemented the feature tracking algorithm of *GazeProjector*. If the template matches, the software calculates a transformation matrix (i.e., a homography). This matrix represents the transformation from the recorded field of view image into the Street View image, retrieved from the API. Hence, the gaze point is transformed onto the Street View imagery, which after clustering the attention, can be used to identify the set of visually most attractive landmarks.

Hardware

To realize this, we developed a hardware prototype, that bundles different devices into a wearable solution to record all needed data at once. Figure 1 depicts the implementation of the device. It consists of the following components: (1) a head-mounted Pupil Labs eye tracker [2], (2) a 9 DOF IMU mounted on the eye tracker[3], (3) a device to record GPS location and (4) a laptop (MacBook Pro 13 inch) driv-

[1] https://developers.google.com/maps

[2] http://pupil-labs.com/pupil/

[3] Sparkfun 9DOF sensor stick with ADXL345 Accelerometer, HMC5883L 3-axis magnetometer, and ITG-3200 gyroscope with the AHRS firmware

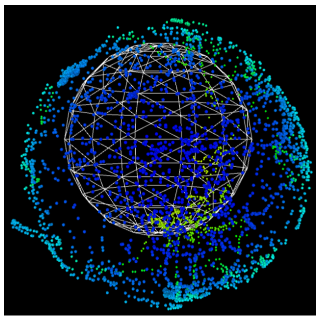

Figure 2: Matlab visualization of the 3-axes compass deviation table created with the AHRS firmware

ing Pupil Labs software. In total, four data streams are recorded: the eye tracker's scene camera stream, the user's gaze data and the user's head orientation are streamed to a laptop. For collecting GPS locations, an iPhone SE was used.

The mounted IMU is able to measure the user's absolute head orientation as yaw, pitch, and roll independently of the eye-tracking device. We used an RFDuino[4], to connect the IMU via Bluetooth LE to the MacBook. The way we mounted the sensor breakout shield ensured that the X axis was pointing forward, the Y axis pointing to the right and Z axis pointing down with respect to the viewing direction. The sensors were mounted on the left earpiece of the headset, where clearance to other cables and the cameras was sufficient in terms of magnetometer deviation. We took special care of the compass calibration to compensate for hard and soft iron errors. We experienced deviations from magnetic north due to electromagnetic induction when the eye-tracking system was running. Figure 2 visualizes the resulting compass calibration which we created with the complete setup switched on.

Software

As we use a Pupil Labs eye tracker to record a person's field of view and gaze data, we also use the Pupil Labs framework [6] to analyze the recorded data. This software is developed in Python and easily extensible through plug-ins. We extended its so-called *Pupil Player*[5], used to play back recorded data, by the following features: (1) Automatic correlation and playback of all recorded data. This is done by simply dragging&dropping a folder that contains the files recorded by the Pupil Labs software, a GPX track of the route walked, and a CSV file containing the head orientations, onto the UI; (2) highlighting the current location of the user on Google Static Maps; (3) displaying the user's view, approximated by using Google Street View imagery; and (4) extracting a sequence of images to create a set of landmarks.

Figure 3 illustrates the *Pupil Player*. In the leftmost image, the software is displaying the current field of view, captured by the scene camera. The center image depicts the described plugin, used to visualize the aformentioned data. Specifically, it overlays the scene camera image with the current Google Street View image, and a map indicating the current location of the user. The rightmost image highlights the matching algorithm, used to map the scene camera image to a Google Street View image by computing the homography matrix.

All our extensions are completely written in Python. We make use of the Google RESTful API for Static Maps and Street View. All necessary data can be queried by passing the location, heading, pitch and size of the target image for the Street View image, and the zoom level for the Static Map image. We developed an extra plugin to transfer *GazeProjector's* feature tracking into Python. For this, we use the OpenCV 3.1 library, which offers implementations of FREAK&FAST [17] for feature detection and description and FLANN [15] for key feature matching. For faster processing, we downscale the Street View images to 640 x 480 and the scene camera images to 360 x 180 pixels. We achieve up to 30 fps, i.e. the processing can be done in real time, and thus also during the data recording.

Preliminary User Study

We conducted a user experiment to verify our proof-of-concept implementation. To do this, we equipped six participants between 22 and 58 years old (M = 39.6, SD = 14.8

[4] http://www.rfduino.com/

[5] https://pupil-labs.com/blog/2014-02/pupil-player-release/

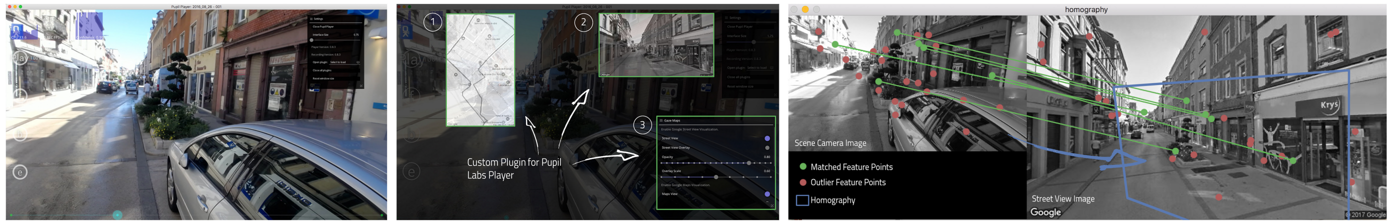

Figure 3: Playback and analysis of the recorded data. The standard Pupil Player is able to play back the data recorded by the eye tracker (left image). The developed custom plugin is able to correlate GPS location, head orientation and eye tracking data. For manual analysis, the path on a Google Map is shown (center figure (1)), as well as the Street View image (center figure (2)). The user is able to set some parameters for the visualization (center figure (3)). Using feature tracking, the plugin determines the transformation between the scene camera's image and the Google Street View image (left image)

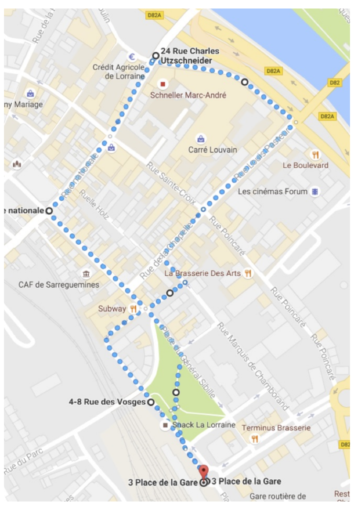

Figure 4: The route that was walked by the participants. The participants started through the park, beginning and ending in front of the railway station at the red marker.

years), 4 male and 2 female, with our device bundle. All participants were recruited from a local university campus and had corrected or normal vision; none reported any visual impairments (e.g., color blindness).

The task of the participants was to walk a pre-defined route through the nearby French town of Saareguemines. All participants rated their familiarity with this city as "rather unfamiliar" or "unfamiliar". The generated route went through the inner part of the city and was constructed as a 1.5 km circular course. The participants were guided using the off-the-shelf audio navigation part of the Komoot Mobile App[6]. As Wenczel et al. showed that the amount of visual attention on more salient landmarks is not affected by whether the user learned the route beforehand, or is incidentally learning it, we decided not to explain the route beforehand [25].

As our prior pilot tests showed that direct sunlight could affect the pupil tracking of the Pupil eye tracker, we added sunglasses to the head-mounted eye tracking device (as can be seen in Figure 1). We chose a pair of lightweight Alpina sports glasses with fracture-resistant polycarbonate lenses that absorb UV and infrared light.

Every participant was first asked to calibrate the head-mounted eye tracker while standing. We used the built-in nine-point 2D calibration procedure of the Pupil framework on a 15-inch laptop screen. After that, we synchronized it manually with the phone used for location tracking, i.e. we captured a start-button press on the phone's screen with the scene camera of the eye tracker. The participants were instructed to follow the audio instructions to finish the route.

During the task, the data was sampled in the following way: Right after the eye tracker calibration, each participant was asked to look straight ahead. We sampled data from the eye tracker for 6 seconds. This was done to have a set of samples to create a mapping between the gaze direction and head orientation relative to each other. On the way through the city, the scene camera video stream and gaze data were recorded through the Pupil eye tracking device at

[6] https://en.komoot.de/

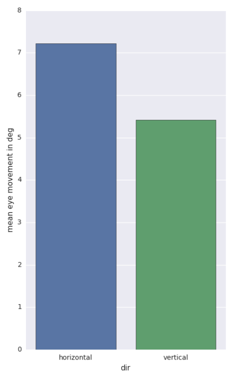

Figure 5: Mean eye movement in degrees, horizontal vs. vertical direction.

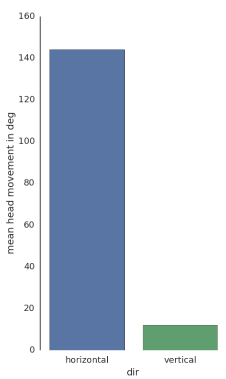

Figure 6: Mean head movements in degrees horizontal vs. vertical direction.

30 Hz; IMU data and location data were recorded at 40Hz and 2Hz respectively.

Results

To do a first evaluation of our proof-of-concept prototype, we aggregated all recorded data of each participant, using the Pupil framework together with the developed plugins. We then computed the amount of scene camera images, which we are able to match to Google Street View imagery. We found that it is possible to successfully match 31.99% of the scene camera's images to Street View image data (SD = 5.6%) on average. That means that we are also able to compute the homography matrix along a third of the walked route.

We further wanted to investigate the variability in eye and head movements: Figure 5 plots the mean eye movements we recorded during the experiment. We noticed a horizontal eye movement of $7.22°$ on average, compared to $5.42°$ in the vertical direction. Figure 6 shows the same plot for the observed head movements. Here we noticed $144.23°$ for horizontal head movements (yaw angle) on average. In the vertical direction (pitch angle), we observed $12.06°$ on average.

Conclusion & Future Work

In this paper, we presented an approach for the automatic extraction of a set of landmarks by combining different data sources with Google Street View imagery. We developed a wearable device by bundling a Pupil Labs head-mounted eye tracker with an IMU sensor and an iPhone SE for GPS logging. We developed a custom plugin for the Pupil Player that allows us to correlate all these data sources. In contrast to existing approaches, we use natural feature tracking, to automatically match the scene camera's image plane to the appropriate Google Street View imagery and map the user's gaze on it. From this we can allocate the attention and extract a set of landmarks.

The conducted experiment gives first insights into the feasibility of our method. We found that it is possible to extract landmarks for a person by matching almost a third of all image data. This seems to be relatively little. Note that we processed the raw data that was sampled. That means we did not take into account the fact that Google Street View images are usually taken from the center of the road. To increase the accuracy of the results, one would have to incorporate the offset between the user's position and the position of the Google Street View imagery and adapt the orientation accordingly. Further, it is very likely that Google Street View images differ from the current scene images, as they could contain other objects like cars, or were recorded in a different season. We noticed a very small variability in the eye movement data, compared to the head movement. The observed $7.22°$ for horizontal movements is within the macular region of the peripheral system. The large values for head movements indicate that people tend to move their head instead of their eyes. On the one hand this could be caused by the fact that people were walking through an unfamiliar city and tried to see as much as possible. On the other hand, it might be sufficient to use only the head orientation and location information, which would be a subject for further research.

In the future we are planning to extend the experiment. We want to investigate the suitability of the computed landmarks compared to existing methods. We also want to find out whether we can extract personalized landmarks and e.g. use them in a life-logging manner similar to [10]. Navigation might be even easier and more natural if we could extract a set of landmarks that fit the user's typical attention behavior.

References

[1] Florian Bockes, Laura Edel, Matthias Ferstl, and Andreas Schmid. 2015. Collaborative Landmark Mining with a Gamification Approach. In *Proceedings of the 14th International Conference on Mobile and Ubiquitous Multimedia (MUM '15)*. ACM, New York, NY, USA, 364–367. DOI: http://dx.doi.org/10.1145/2836041.2841209

[2] Barry Brown and Eric Laurier. 2012. The normal natural troubles of driving with GPS. In *Proceedings of the 2012 ACM annual conference on Human Factors in Computing Systems*. ACM, 1621–1630.

[3] Matt Duckham, Stephan Winter, and Michelle Robinson. 2010. Including landmarks in routing instructions. *Journal of Location Based Services* 4, 1 (2010), 28–52.

[4] Ioannis Giannopoulos, Peter Kiefer, and Martin Raubal. 2015. GazeNav: Gaze-Based Pedestrian Navigation. In *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '15)*. ACM, New York, NY, USA, 337–346. DOI: http://dx.doi.org/10.1145/2785830.2785873

[5] Jana Helgath, Simon Provinsky, and Timo Schaschek. 2015. Landmark Mining on a Smartwatch Using Speech Recognition. In *Proceedings of the 14th International Conference on Mobile and Ubiquitous Multimedia (MUM '15)*. ACM, New York, NY, USA, 379–383. DOI: http://dx.doi.org/10.1145/2836041.2841212

[6] Moritz Kassner, William Patera, and Andreas Bulling. 2014. Pupil: An Open Source Platform for Pervasive Eye Tracking and Mobile Gaze-based Interaction. In *Adjunct Proceedings of UbiComp 2014 (UbiComp '14 Adjunct)*. ACM, New York, NY, USA, 1151–1160. DOI: http://dx.doi.org/10.1145/2638728.2641695

[7] Peter Kiefer, Ioannis Giannopoulos, Dominik Kremer, Christoph Schlieder, and Martin Raubal. 2014b. Starting to Get Bored: An Outdoor Eye Tracking Study of Tourists Exploring a City Panorama. In *Proceedings of the Symposium on Eye Tracking Research and Applications (ETRA '14)*. ACM, New York, NY, USA, 315–318. DOI: http://dx.doi.org/10.1145/2578153.2578216

[8] Peter Kiefer, Ioannis Giannopoulos, and Martin Raubal. 2014a. Where am I? Investigating map matching during self-localization with mobile eye tracking in an urban environment. *Transactions in GIS* 18, 5 (2014), 660–686.

[9] Christian Lander, Sven Gehring, Antonio Krüger, Sebastian Boring, and Andreas Bulling. 2015. GazeProjector: Accurate Gaze Estimation and Seamless Gaze Interaction Across Multiple Displays. In *Proceedings of the 28th Annual ACM Symposium on User Interface Software & Technology (UIST '15)*. ACM, New York, NY, USA, 395–404. DOI: http://dx.doi.org/10.1145/2807442.2807479

[10] Christian Lander, Antonio Krüger, and Markus Löchtefeld. 2016. "The Story of Life is Quicker Than the Blink of an Eye": Using Corneal Imaging for Life Logging. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing: Adjunct (UbiComp '16)*. ACM, New York, NY, USA, 1686–1695. DOI: http://dx.doi.org/10.1145/2968219.2968337

[11] David R Large, Gary Burnett, Steve Benford, and Keith Oliver. 2016. Crowdsourcing good landmarks for in-vehicle navigation systems. *Behaviour & Information Technology* (2016), 1–10.

[12] Markus Löchtefeld, Sven Gehring, Johannes Schöning, and Antonio Krüger. 2010. PINwI: Pedestrian Indoor Navigation Without Infrastructure. In *Proceedings of the 6th Nordic Conference on Human-Computer Interaction: Extending Boundaries (NordiCHI '10)*. ACM, New York, NY, USA, 731–734. DOI: http://dx.doi.org/10.1145/1868914.1869016

[13] Kevin Lynch. 1960. *The image of the city*. Vol. 11. MIT press.

[14] Andrew J. May, Tracy Ross, Steven H. Bayer, and Mikko J. Tarkiainen. 2003. Pedestrian navigation aids: information requirements and design implications. *Personal and Ubiquitous Computing* 7, 6 (2003), 331–338. DOI: http://dx.doi.org/10.1007/s00779-003-0248-5

[15] Marius Muja and David G. Lowe. 2009. Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration. In *In VISAPP International Conference on Computer Vision Theory and Applications*. 331–340. DOI: http://dx.doi.org/10.1.1.160.1721

[16] Clemens Nothegger, Stephan Winter, and Martin Raubal. 2004. Selection of salient features for route directions. *Spatial cognition and computation* 4, 2 (2004), 113–136.

[17] Raphael Ortiz. 2012. FREAK: Fast Retina Keypoint. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR '12)*. IEEE Computer Society, Washington, DC, USA, 510–517. http://dl.acm.org/citation.cfm?id=2354409.2354903

[18] Clark C Presson and Daniel R Montello. 1988. Points of reference in spatial cognition: Stalking the elusive landmark. *British Journal of Developmental Psychology* 6, 4 (1988), 378–381.

[19] Martin Raubal and Stephan Winter. 2002. *Enriching Wayfinding Instructions with Local Landmarks*. Springer Berlin Heidelberg, Berlin, Heidelberg, 243–259. DOI: http://dx.doi.org/10.1007/3-540-45799-2_17

[20] Kai-Florian Richter and Stephan Winter. 2014. *Landmarks: GIScience for intelligent services*. Springer Science & Business.

[21] Johannes Schöning, Antonio Krüger, Keith Cheverst, Michael Rohs, Markus Löchtefeld, and Faisal Taher. 2009. PhotoMap: Using Spontaneously Taken Images of Public Maps for Pedestrian Navigation Tasks on Mobile Devices. In *Proceedings of the 11th International Conference on Human-Computer Interaction with Mobile Devices and Services (MobileHCI '09)*. ACM, New York, NY, USA, Article 14, 10 pages. DOI: http://dx.doi.org/10.1145/1613858.1613876

[22] Molly E Sorrows and Stephen C Hirtle. 1999. The nature of landmarks for real and electronic spaces. In *International Conference on Spatial Information Theory*. Springer, 37–50.

[23] Ariane Tom and Michel Denis. 2003. *Referring to Landmark or Street Information in Route Directions: What Difference Does It Make?* Springer Berlin Heidelberg, Berlin, Heidelberg, 362–374. DOI: http://dx.doi.org/10.1007/978-3-540-39923-0_24

[24] Shoko Wakamiya, Hiroshi Kawasaki, Yukiko Kawai, Adam Jatowt, Eiji Aramaki, and Toyokazu Akiyama. 2016. Lets Not Stare at Smartphones While Walking: Memorable Route Recommendation by Detecting Effective Landmarks. In *Proceedings of the 2016 ACM International Joint Conference on Pervasive and Ubiquitous Computing (UbiComp '16)*. ACM, New York, NY, USA, 1136–1146. DOI: http://dx.doi.org/10.1145/2971648.2971758

[25] Flora Wenczel, Lisa Hepperle, and Rul von Stülpnagel. 2016. Gaze behavior during incidental and intentional navigation in an outdoor environment. *Spatial Cognition & Computation* (2016), 1–22.

[26] Marius Wolfensberger and Kai-Florian Richter. 2015. *A Mobile Application for a User-Generated Collection of Landmarks*. Springer International Publishing, Cham, 3–19. DOI:http://dx.doi.org/10.1007/978-3-319-18251-3_1