A Toolkit for Analysis and Prediction of Touch Targeting Behaviour on Mobile Websites

Daniel Buschek, Alexander Auch, Florian Alt
Media Informatics Group
University of Munich (LMU)
Amalienstr. 17, 80333 Munich, Germany
daniel.buschek@ifi.lmu.de, aucha@cip.ifi.lmu.de, florian.alt@ifi.lmu.de

ABSTRACT
Touch interaction on mobile devices suffers from several problems, such as the thumb's limited reach or the occlusion of targets by the finger. This leads to offsets between the user's intended touch location and the actual location sensed by the device. Recent research has modelled such offset patterns to analyse and predict touch targeting behaviour. However, these models have only been applied in lab experiments for specific tasks (typing, pointing, targeting games). In contrast, their applications to websites are yet unexplored. To close this gap, this paper explores the potential of touch modelling for the mobile web: We present a toolkit which allows web developers to collect and analyse touch interactions with their websites. Our system can learn about users' targeting patterns to simulate expected touch interactions and help identify potential usability issues for future versions of the website prior to deployment. We train models on data collected in a field experiment with 50 participants in a shopping scenario. Our analyses show that the resulting models capture interesting behavioural patterns, reveal insights into user-specific behaviour, and enable predictions of expected error rates for individual interface elements.

ACM Classification Keywords
H.5.2 Information Interfaces and Presentation (e.g. HCI): Input devices and strategies (e.g. mouse, touchscreen)

Author Keywords
Touch; Targeting; Mobile; Web; Toolkit; User Model

INTRODUCTION
Mobile handheld touchscreen devices challenge usable web designs with additional factors: For example, related research has studied the thumb's limited reach [6], occlusion of targets by the finger [5], the influence of varying finger pitch, roll and yaw [14, 15], and body movement [12] and encumbrance [19]. Other lines of research modelled typing behaviour [4, 10, 25] and touch-to-target distances [8, 9, 21], revealing patterns of targeting errors. We expect the insights gained in

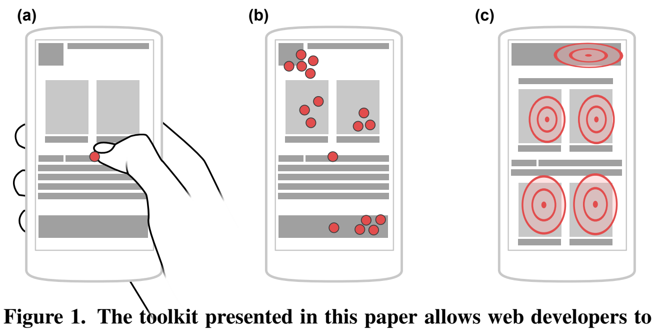

Figure 1. The toolkit presented in this paper allows web developers to record touch interactions on mobile websites *(a)*, to visualise the collected touch data *(b)*, and to derive touch targeting models which can predict expected touch distributions and targeting error rates for graphical interface elements in different layouts *(c)*. Hence, our toolkit can help to inform design choices for improving mobile websites with respect to touch interaction.

these studies to influence touch interactions with mobile websites as well. However, it still remains unclear how developers of mobile websites could benefit from information, and how they could employ the developed modelling techniques, which have not been applied to interactions on websites yet.

We consider this an important direction to advance mobile web development: For example, current responsive websites can consider a range of physical device properties (e.g. resolution, aspect ratio, orientation). However, when it comes to user characteristics, websites are mostly only aware of *strategic* behaviour (e.g. shopping recommendations based on items bought in the past). In contrast, current mobile web design neglects *physical* aspects of user behaviour, such as touch targeting patterns under different conditions. Insights from prior work (e.g. respecting the thumb's reach [6] or expected offsets [9, 24]) reveals unexplored potential to improve touch interaction.

Therefore, we aim to narrow the gap between the rich line of research addressing touch behaviour, and the lack of practical applications of these insights, in particular in a mobile web context. We focus on touch targeting and the resulting behavioural patterns: We enable observing, analysing, and modelling physical aspects of user behaviour within an easy-to-deploy framework (Figure 1) to support web developers in improving websites for touch interaction on mobile devices.

To this end, we contribute: 1) concepts for analysis and prediction of touch targeting behaviour for mobile websites; 2) a toolkit which implements these concepts to be used by web developers; and 3) example analyses and predictions with an application of the toolkit to data collected in a field study.

EICS'15, June 23–26, 2015, Duisburg, Germany.

ACM 978-1-4503-3646-8/15/06...$15.00.
DOI: http://dx.doi.org/10.1145/2774225.2774851

This paper is structured as follows: First, we review related literature with regard to web usability, touch behaviour, and touch modelling. We then present the concept of touch offset modelling and describe a linear model that can map between touch and target locations. Afterwards, we describe our toolkit, focussing on data collection and analysis, as well as on the prediction of user behaviour. Finally, we present a field study observing touch interactions with a mobile website; we discuss findings with respect to analyses and predictions of users' touch behaviour, employing our toolkit with models trained on the collected data.

RELATED WORK

A number of approaches have been presented which aim to improve usability of websites by observing user behaviour. Research in the desktop domain focused on tracking and analysing a user's click stream, mouse movement activity, and gaze.

Click locations and activated elements on a page (often referred to as click analytics [11]) can be collected from different sources, most notably from web analytics tools, such as *Google Analytics*, *ClickHeat* or *CrazyEgg*. This data can inform web development: For example, Arendt and Wagner [1] used such data to enhance the usability of a library website. A different project, *WebQuilt* [16], tracks the user's navigation path across a website using a proxy. In this work, the authors particularly focus on the visualisation of the obtained data. Furthermore, Müller and Lockerd presented a tool to track mouse movement activity on webpages [17]. A more comprehensive way of assessing usability issues on websites was presented by Atterer et al. [2, 3]: Their *UsaProxy* allows for detailed tracking and later analysis of user actions, such as mouse navigation, scrolling, clicks, and text input. Finally, Reeder et al. presented two tools to analyse the gaze of users browsing the web [20].

These projects gathered useful behavioural information about users of desktop systems, and support usability testing of websites. However, directly applying these methods to the mobile web is challenging: Users browsing the web on a touchscreen device usually do not leave any cursor traces, and gaze is difficult to assess, since the users' fingers constantly obscure parts of the screen. Nevertheless, researchers interested in the mobile domain can analyse touch behaviour, similarly to click streams in desktop settings. This approach motivates our work: We present a toolkit to support such analyses of touch behaviour on websites.

Recent related research by Nebeling et al. [18] presented a method to adapt websites based on collected mobile touch data, from which they derived adaptation rules. For example, font sizes could be increased if the user's hit rate for navigation links falls below 50%. In this paper, we present a toolkit with a similar goal. However, in contrast to the related work, we derive statistical models of users' touch behaviour. This allows us to capture behavioural patterns across the screen, instead of relying on point estimates (e.g. averages) of interaction behaviour. Furthermore, our models allow for predictions of expected behaviour when evaluating changes, say, to the website's layout. We see these approaches as complementary: the rules from related work and our models could be combined, for example by increasing font size pre-emptively, if the *predicted* hit rate falls below a certain threshold.

Related research concerned with touch targeting behaviour mainly used such models to correct sensed touch locations and thus improve touch accuracy [9, 13, 22, 24]. These projects did not target websites in particular, but rather abstract targets (e.g. crosshairs, circles), or keyboards [23]. Although we can expect that touch offsets on websites could be corrected with this approach as well, our toolkit rather aims to improve web designs with respect to touch usability. To this end, we focus on the *analysis* of touch targeting patterns, and *predictions* of future expected touch behaviour, using similar touch models as in the related work. Our tools aim to support web developers and designers in their task to improve websites and thus mitigate problems related to touch targeting errors before they occur. This offers more flexibility and control to the developer than automatic adaptations after deployment, which may, for example, break the developers intended design and layout.

Further recent related work in the mobile domain has optimised websites with respect to accessibility by magnifying font sizes as much as possible without introducing layout problems [7]. Our work also aims to inform possible changes to websites to make them more accessible, but with respect to touch input. Moreover, as stated above, we target pre-emptive optimisation by supporting web developers, not reactive automatic adaptations on already deployed websites.

TOUCH TARGETING MODELS

In this section, we introduce the formal foundations for predictions with our toolkit. In particular, we present touch offset modelling based on related research [9, 22, 24]: First, we explain our general formal notation for touch input. Second, we describe a linear offset model used to map these touch locations to target locations (and vice versa). The particular mapping functions learned from given touch data in this way allow us to capture patterns of user behaviour for different contexts and conditions (e.g. hand posture, screen size).

Touch Input

We use the following vector notation for touch input. Each touch $\mathbf{t}$ is represented as a two-dimensional vector containing its screen coordinates:

$$\mathbf{t} = (x, y)^T \tag{1}$$

Additionally, we measure the offset vectors $\mathbf{o}$ between the touch location and the corresponding target location (e.g. centre of the target whose bounding box contains the touch). Offsets are computed as the euclidean distances between touches and targets along each dimension x and y.

Offset Models

For the purpose of the toolkit presented in this paper, we employ linear touch targeting models with non-linear basis functions, similar to related work [9].

Basis functions map the touch locations to higher dimensions to allow for non-linearity in the resulting functions. Note that the model parameters are still linear, which keeps the training costs low (i.e. linear regression).

In the following, we denote basis functions with Φ. For a quadratic model, similar to the related work [9], we set:

$$\Phi(\mathbf{t}) = (1, x, y, x^2, y^2)^T \tag{2}$$

A touch location vector is thus extended by a constant 1 (bias), and the quadratic terms of the coordinates (x^2, y^2). These extra dimensions give the model additional flexibility, which has been shown to prove useful to describe users' touch targeting behaviour [9].

In summary, the touch inputs $\mathbf{t}$ and their high-dimensional mapping $\Phi(\mathbf{t})$, together with offsets $\mathbf{o}$ provide the foundation for training our targeting model.

Training

A training set contains touch locations, and target locations or offsets. To train a linear offset model from a given dataset, we need to solve the linear regression problem defined by the training examples: Fist, we arrange the transformed touch locations $\Phi(\mathbf{t})$ of the N training examples as N rows of a design matrix $\mathbf{X} \in \mathbb{R}^{N \times 5}$.

Furthermore, let $\mathbf{o}_x$ and $\mathbf{o}_y$ denote the vectors comprising of the offsets of all training examples along the x and y dimension, respectively. We can then solve for the parameters of the model, namely $\mathbf{w}_x, \mathbf{w}_y$:

$$\mathbf{w}_x = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\mathbf{X}^T\mathbf{o}_x \tag{3}$$

We can compute $\mathbf{w}_y$ analogously (i.e. using $\mathbf{o}_y$). Therein, $\lambda \in \mathbb{R}$ denotes the regularisation parameter. Regularisation helps to restrict the model in order to avoid overfitting, in other words, learning noise in the training data instead of the users' actual, systematic touch targeting patterns.

Figure 2 visualises three examples: The plots show learned horizontal targeting patterns, that means the functions defined by $\mathbf{w}_x$, for three models trained on data from our study.

Prediction

The resulting weight vectors $\mathbf{w}_x, \mathbf{w}_y$, and the transformation Φ, allow us to make predictions for future touch events. In particular, for a new touch location $\mathbf{t}'$ the model predicts the offset μ as follows:

$$\begin{aligned} \mu &= (\mu_x, \mu_y)^T \\ \mu_x &= \mathbf{w}_x^T\Phi(\mathbf{t}') \\ \mu_y &= \mathbf{w}_y^T\Phi(\mathbf{t}') \end{aligned} \tag{4}$$

We can then add this predicted offset μ to the touch location $\mathbf{t}'$ to correct it. In other words, this addition yields a prediction for the true intended target location for the given touch location.

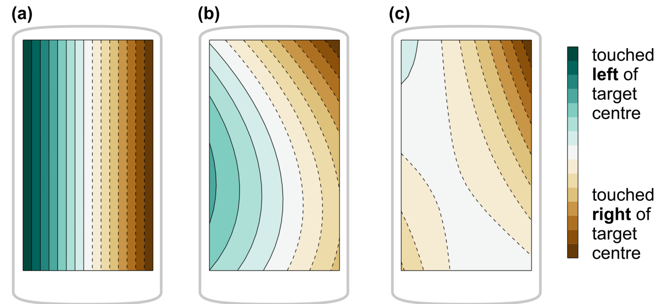

Figure 2. Horizontal offset predictions of models trained on collected thumb touches for the three target types of our study website (compare to Figure 7): *(a)* full width buttons, *(b)* a 3×2 grid of thumbnails, and *(c)* text links in articles. Colour scales are defined per plot to facilitate the perception of patterns. While the full width targets *(a)* lead to a trivial horizontal offset pattern, thumbnails *(b)* and text links *(c)* reveal interesting model shapes. For example, white regions reveal the most precise areas in horizontal direction, as learned from the corresponding touches and targets. Especially for the smallest targets *(c)*, we further observe a tendency to touch to the right of the target centres in most screen regions, explained by the use of the right thumbs in the study.

Moreover, we can compute variances associated with these predictions along each dimension, for example for x:

$$\begin{aligned} {\sigma'_x}^2 &= \hat{\sigma}_x^2\Phi(\mathbf{t}')^T(\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I})^{-1}\Phi(\mathbf{t}') \\ \hat{\sigma}_x^2 &= \frac{1}{N}\sum_{i=1}^{N}(o_x^{(i)} - \mathbf{w}^T\Phi(\mathbf{t}_i))^2 \end{aligned} \tag{5}$$

Finally, ${\sigma'_y}^2$ and $\hat{\sigma}_y^2$ are computed in the same way. Together, these predictions define a bivariate Gaussian:

$$\mathcal{N}(\mathbf{t}' + \mu, \mathbf{\Sigma}) \tag{6}$$

with a covariance matrix defined as follows:

$$\mathbf{\Sigma} = \begin{bmatrix} {\sigma'_x}^2 & 0 \\ 0 & {\sigma'_y}^2 \end{bmatrix} \tag{7}$$

Inverse Offset Models

In contrast to the use of offset models in related work [8, 9, 13, 22, 23, 24], in this paper we are particularly interested in applying *inverse* offset models – in other words, models trained to *predict touch locations given target locations*.

To train such models, we simply need to swap the roles of touch and target locations in the explained equations for training and prediction. Note that the model equations themselves stay exactly the same as above. As a result, inverse offset models predict distributions of likely touch locations for a given target location. We use these distributions to simulate touch behaviour, as explained in the following section.

THE TOOLKIT

We have described modelling touch targeting behaviour. Next, we introduce our application concept and its implementation in our toolkit, which employs these models to learn about users' behaviour from collected data. Figure 1 provides a visual overview of the general concept. In the following, we present the toolkit in two parts: First, we describe functionality related to touch data collection and analysis. Second,

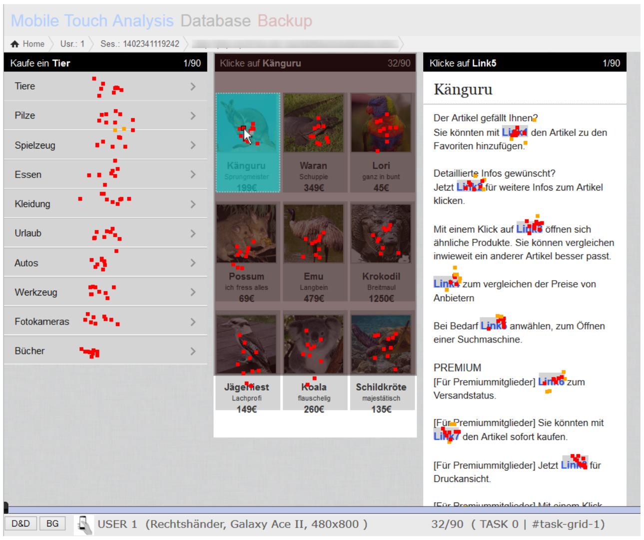

Figure 3. Graphical user interface of the toolkit for analysis of the website used in the field study: In this screenshot, it shows the three main pages of the study website with the touches of a selected user. Context information is shown at the bottom, such as hand posture, device model, and page resolution of the website as it was rendered on the device. Hovering over indivudal touches highlights the target area, as well as the viewport at the time of the touch.

we explain our approach of using targeting models to generalise patterns in the collected data. This enables predictions of expected touch behaviour for other layouts or websites.

Data Collection and Analysis

Our toolkit provides means to collect touch events on websites, in particular including the exact touch locations, the viewport, and target elements. This information is extracted from the browser's touch events and the website via JavaScript, and then sent to a basic PHP/MySQL-server. We chose this simple framework since it is easy to integrate into an existing website.

As a fundamental analytical perspective on touch behaviour, our toolkit provides interactive visualisations of the collected touches. These can be examined, for example, by a web developer on a desktop computer. Figure 3 shows an example view for the "shopping" website from our user study: It displays the three subpages used in the experiment, with touches overlayed on top. The visualisation is interactive: Hovering over a touch with the cursor highlights the full area of the target graphical interface element, as well as the viewport as it was set (e.g. via scrolling) on the device at the time of the selected touch.

Simulation and Prediction

So far, we have described how to analyse user behaviour by collecting and visualising touches. In the next step, we aim to learn and generalise patterns in this collected data to predict and simulate future expected behaviour. Here, we employ the described touch targeting models. The resulting system helps web developers to evaluate changes to their websites prior to deployment, without having to collect new user data. Note, that we do not aim to replace actual user studies entirely. However, our system can help to judge changes effortlessly

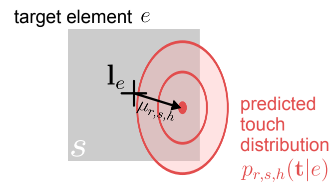

Figure 4. Prediction of a distribution of likely touch locations $p_{r,s,h}(\mathbf{t}|e)$ for a target interface element e of shape s at location $\mathbf{l}_e$, assuming hand posture h.

to inform further iterations before user testing. We first describe the formal approach, before showing the implemented functionality as seen by a developer.

Approach

To enable predictions of touch locations for new or changed websites based on past behaviour, we derive inverse offset models from the collected data. In particular, our toolkit trains and stores several models to capture touch behaviour and predict likely touch locations under different interaction conditions. In this paper, we split the collected touch data by the following conditions:

- *page resolution* r, which mostly coincides with device and screen size in our study
- *target shape* s; our study observes three target shapes: full width buttons, quadratic images, and rectangular text links
- *hand posture* h; in particular thumb and index finger input in our study

We chose these conditions for this project since they cover an interesting range of influences related to device, website, and user. Note, that this choice presents no conceptual limitation, as our approach may include further values or additional types of context (e.g. walking vs standing), as long as we can collect touches (e.g. in a user study) to train corresponding models. Complementary, we can train less specific models, if some of the factors are not measured.

After grouping the data by r, s, h, we train and store one inverse offset model, denoted $m_{r,s,h}$, for each of the resulting subsets of touches. This model can then predict users' touch behaviour for the corresponding conditions. Formally, it predicts a bivariate Gaussian distribution $p_{r,s,h}(\mathbf{t}|e)$ of likely touch locations $\mathbf{t}$ for a graphical interface element e:

$$p_{r,s,h}(\mathbf{t}|e) = m_{r,s,h}(\mathbf{l}_e) = \mathcal{N}(\mathbf{l}_e + \mu_{r,s,h}, \Sigma_{r,s,h} + n\Sigma_n) \qquad (8)$$

where $\mu_{r,s,h}$ (see Equation 4) and $\Sigma_{r,s,h}$ (see Equation 7) result from the prediction of the model $m_{r,s,h}$ for the target element e at location $\mathbf{l}_e$ (see Equation 6). Figure 4 illustrates these values.

Moreover, we consider "noise" with covariance $n\Sigma_n$. This covariance matrix is set proportional to the target's width and height with a scaling factor n. Hence, it enables predictions for elements with known shapes, but different sizes compared to the ones in the training data. For example, the size of a text link might change due to a change of the text in a new version of the website. By including variance related to the element's size, our approach can account for such cases.

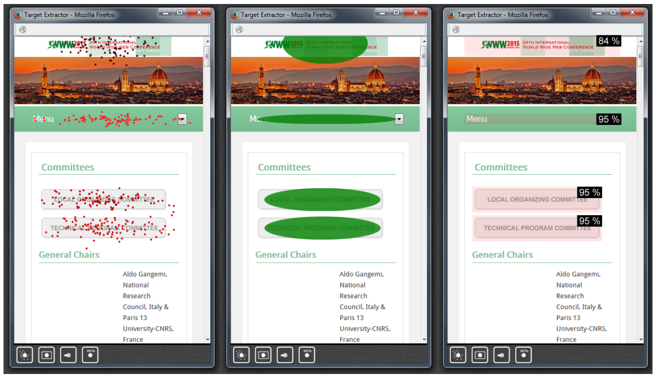

Figure 5. Three views showing estimated touch behaviour for a web conference website, computed by our toolkit when trained on data from our user study. The left view shows simulated touches for the header, the menu dropdown list, and the two committee buttons. The middle view plots the expected distribution of touches for these elements (two sigma ellipses). Finally, the right view shows estimated hit rates for these targets. The high predicted accuracy for the main elements suggests that this website is suitable for mobile use with touch input.

The bivariate Gaussian distributions $p_{r,s,h}(\mathbf{t}|e)$ are then used to sample touch locations $\mathbf{t}$ for the corresponding target e. In other words, we simulate touch interactions with this interface element.

Furthermore, we can compute expected error rates for elements, meaning the ratio of sampled touches that fall outside of the bounding box of this interface element (i.e. Monte Carlo approximation of the "overlap" of the predicted touch distribution and the target's area).

Implementation
We implemented the described approach using JavaScript for the computations, and HTML, CSS, and JavaScript for the front end, which visualises the resulting predictions. The toolkit is intended to be used on a desktop computer. It simulates viewing a given website on a mobile device for a chosen page resolution and assuming a chosen hand posture. Figure 5 shows such prediction views in our toolkit for a part of a web conference website.

For each target on the page, an overlay shows the simulated touches (red dots) and their underlying predicted distributions (green ellipses). Furthermore, each target is annotated with the expected error rate (i.e. ratio of mistouches). We chose dots and ellipses as a simple visualisation, but this can easily be extended. For example, heatmaps[1] could be directly generated from the predictive model by visualising the predicted density of touches at each pixel.

These predictions are computed with an inverse offset model trained on touches collected for the chosen page resolution, hand posture, and target type, as described previously.

The view is scrollable, and all predictions (touches, distributions, error rates) are updated accordingly, since the location of targets on the screen changes when scrolling. Therefore, web developers can gain insights into expected touch behaviour and error rates dynamically, while exploring their website with our toolkit.

[1]e.g. https://heatmaps.io/

Figure 6. The three targets used in the study website. From left to right, users targeted 1) at list entries, spanning the full width of the screen; 2) at images with captions, arranged in a 3×2 grid; and 3) at links in short paragraphs of text.

USER STUDY AND DATA COLLECTION
We conducted a field study with a custom website. Our goal was to collect touch interactions with a mobile website to demonstrate possible analyses and predictions when applying our toolkit. Participants performed several tasks within a shopping scenario: In particular, they had to select certain categories, products, and text. This covered targeting at full width list buttons, quadratic images, and textual hyperlinks (see Figure 6), which we consider to be common and important elements of mobile websites today.

Participants
We recruited participants via social networks and email. In total, 64 individual users participated in the study. We did not ask users for extensive demographic information to keep the required effort as small as possible. However, since we mainly advertised the study among undergraduate university students, we can expect that many participants belong to this demographic group. Due to the nature of this field study, we do not have detailed participant information, such as hand size, which may influence touch offset models. However, this paper studies *applications* of these models to website analysis - and does not aim to repeat detailed quantitative evaluations from related work [8, 9, 22, 24].

Apparatus
In this study, we used a custom mobile website to collect touch targeting data in a shopping scenario. The general design of this website was inspired by large online shopping websites (e.g. Amazon).

Front End
On the front end, our study website has two main views:

The *preparation page* explained the study procedure to the participant. This explanation included a visualisation and explanation of the device orientation (always portrait), and the hand posture for the individual participant (either thumb or index finger input, alternating between participants). The page also asked for the participant's explicit consent to take part in the experiment and the data collection.

The *task page* showed the tasks of the study (see Figure 7): The current task was displayed in a status bar at the top of the screen (e.g. *"Buy a tool."*). The rest of the screen showed the actual content page. It switched through three states, corresponding to the three studied target types: 1) a list of full width buttons to choose a product category (e.g. *tools*); 2) a 3×2 grid of quadratic images, with captions beneath each image, to choose a product; and 3) a product page with title and several paragraphs with ten text links in total.

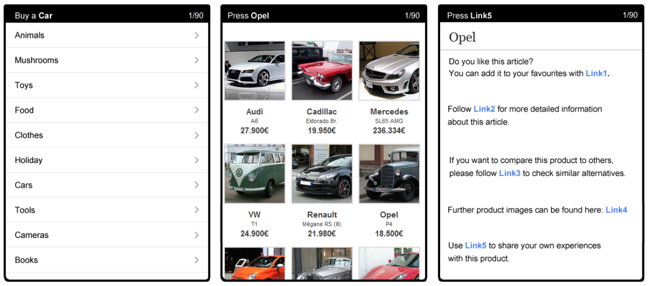

Figure 7. The three screen types of the study website. In each shopping task, participants navigated through these three types, selecting the given category, then the product, and finally the link.

Back End

We used a PHP server and a MySQL database to store the touch data collected with the study website. For each touch, we stored timestamp, touch locations (at up, down, cancel events, as far as supported by the user's browser), target locations (x, y), target types (button, image grid, link) and sizes (width, height), visible targets, intended target, actually hit target, and viewport offset and size.

Procedure

The study was distributed via a link to our website. If the site could not detect a mobile web client, it showed a message asking the visitor to return on a mobile device. All mobile visitors were then directed to the preparation page.

Preparation Page

This page informed the participant about the study procedure, as described above. Moreover, users provided information about their dominant hand and their mobile device model by filling in a short form. They were further asked to confirm their consent with the study and data collection by marking a check box. Finally, participants submitted their information with a button.

A user ID was assigned to the participant when pressing the button. All users with even IDs were then asked to hold the device in the right hand, touching with the right thumb. Users with odd IDs were asked to hold the device in the left hand, touching with their right index finger. Visualisations were used to help explain how to hold the device.

Task Page

Each participant had to complete 90 shopping tasks. For each such shopping task, the user had to navigate through the three screens described above: 1) a list of ten product categories, 2) a grid of nine product images, and 3) a product page with ten text links.

The task description at the top of the screen was updated after each of these three steps, for example:

buy a ***tool*** → *buy a* ***hammer*** → *read* ***link 5***

The order of the 90 shopping tasks was randomised. For each participant, these tasks resulted in nine touches per target for the list and the text links, and in ten touches per target for the image grid. After completing all tasks, the website displayed a message to inform the participant about the end of the study.

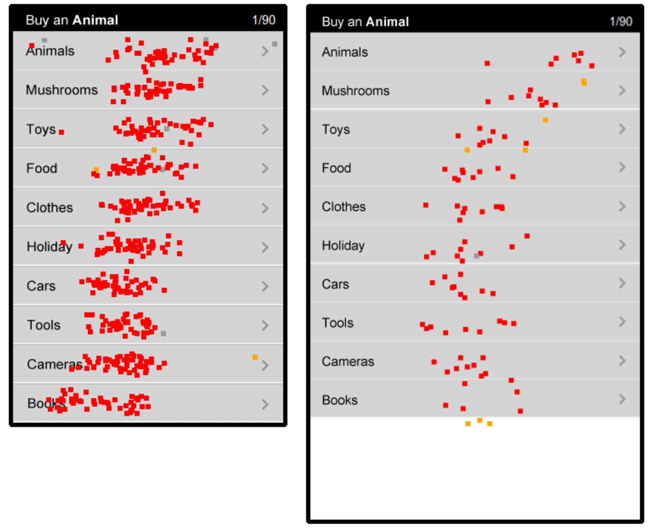

Figure 8. Analysing thumb touches for the full width list entry targets (left): The distribution of touch locations indicates comfortably reachable areas for different device and screen sizes. For example, an arc-like shape becomes particularly apparent for users of a large device (right).

RESULTS

Since 14 of our 64 participants only completed a few tasks, we exclude these users from the following analyses. Hence, we focus on the data of the remaining 50 individuals.

Analysing Touch Behaviour

We apply our toolkit to analyse the touch data from our user study, revealing possibilities for improving the design of the website. All figures in this section show screenshots of our toolkit analysing the website of the study.

Revealing Physical Constraints

Analysing touch locations allows us to gain insights into possible physical constraints, for example revealing comfortably reachable areas for thumb input, as shown in Figure 8.

These patterns suggest that users do not like to change their grip on the device to always touch on a certain region of a target element (e.g. the text, the centre, the arrow); they will rather reach as far as necessary to hit the target comfortably.

While limits in reach due to hand anatomy and grip are known to HCI [6], our findings suggest that studying them through observed touch patterns can inform mobile web design beyond current standards: For example, the touch distributions in Figure 8 confirm that targets at the top of the screen should indeed span the full screen width, as thumb users will otherwise likely have to change their grip to reach small targets in the top corners.

Note that this finding contradicts some current mobile designs offering a small home-button in a top corner (e.g. the popular bootstrap[2] framework). The arc-like shape further suggests that a radial navigation menu may be a more comfortable alternative to a simple list layout on larger devices.

[2] http://getbootstrap.com/

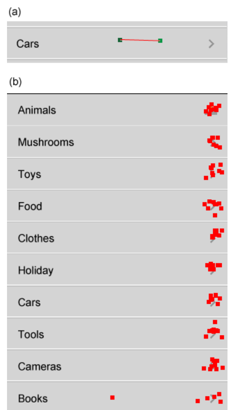

Figure 9. Analysing touch events can reveal a user's assumption or mental model of the website. This example shows *(a)* a user's first touch interaction with the list. The user performed a swipe gesture instead of activating the list button with a single touch. Another user *(b)* only used the small arrows to the right. These insights can help to inform design choices regarding graphical representation and implemented behaviour.

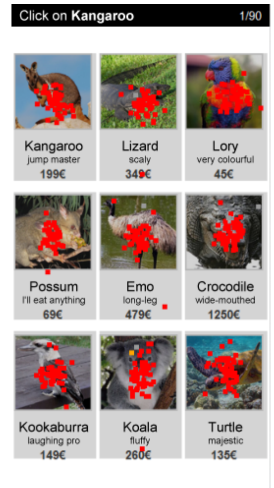

Figure 10. Distribution of thumb touches for a grid of images. With this visualisation, our toolkit reveals that most users touched on the image itself, while the caption was not perceived as a touchable area.

Revealing Mental Models

Complementary to patterns of physical origin, our analyses also revealed patterns most likely resulting from mental models of individual users: For example, one user first tried to swipe through the list view, as indicated by a left-to-right drag event (Figure 9a). Another user misinterpreted the small arrows on the right end of the list buttons as the only touchable areas (Figure 9b).

To avoid possible frustration through unimplemented swiping functionality or small perceived target areas, a future iteration of the website could thus try to adapt the presentation: For example, it may be helpful to remove the arrow or to add gaps or a raised 3D look to highlight the actual bounding boxes of the buttons. Complementary, swiping functionality could be implemented as well.

Another related pattern can be observed in the distribution of touches aiming at the grid of images, see Figure 10: Here, the toolkit reveals a strong preference of users to target the image itself. In contrast, they avoided to touch on the caption, although both image and caption were linked to the product page in the study.

Only 5.4% of all touches were aimed at the image captions. We explain this result with the visual dominance of the image compared to the caption. Moreover, the images' bounding boxes might be perceived more clearly as a single touchable area, compared to the combined area of image and caption. While touching the correct images was not an issue for users in our study, a different caption design may have offered even larger perceived touch areas. For example, we could use rectangular images with the caption overlaid on top of the bottom of each image.

Predicting Touch Behaviour

Complementary to the analytical perspective, we next discuss example applications and derived insights with our predictive tools as well. A first example is shown in Figure 5. Here, we further apply the tools to other websites.

Text Links in a Wikipedia Article

As a first example application of our prediction tool, we analyse text links, and compare two versions of a wikipedia page, which differ by the used font size. Figure 11 shows the resulting predictions for text links on this website. The toolkit predicts slightly higher hit rates (i.e. fewer mistouches) for the larger font, and higher rates for links near the screen centre. In contrast, lower hit rates occur for targets near the corners. These predictions match our observations from the user study, as well as expectations based on related work [6].

Note the relatively low predicted hit rates for the table of contents links aligned at the left of the screen (Figure 11, left). In consequence, we suggest to use full width elements for such lists of links instead, which are easier to reach and offer a larger touch area in general.

Navigation on a Non-Mobile Website

In this second example, we apply our prediction tool to a website which is clearly not optimised for mobile use. Figure 12 shows the resulting predictions for right-hand thumb touches aiming at a navigation bar on the left side, simulating a larger mobile device.

The toolkit predicts relatively low hit rates (i.e. many mistouches). To investigate this further, we can examine individual samples, in other words single simulated touches (see Figure 12). This reveals that the overall distributions are shifted to the right of the target. We explain this result with limited reachability. In conclusion, we can consider these predictions as correct in that they match expectations for a website which was clearly not designed for mobile use.

Note, that while this result may possibly seem expected, this pattern was discovered by our model "on its own": It was learned from the collected data of the study. As a result, this example application of our toolkit shows that our approach can learn and transfer patterns observed on one website to another one. Again, these predictions also match the findings for thumb reachability from related work [6].

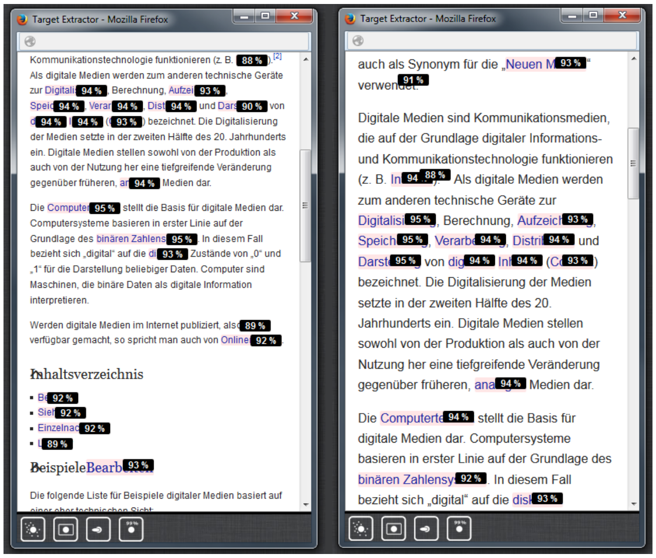

Figure 11. Predictions of expected hit rates for thumb input aiming at links in a wikipedia article with two font sizes: Our toolkit simulates touches, and predicts and annotates resulting hit rates for each target element. Predictions are updated accordingly when scrolling. Hence, web developers can explore expected touch behaviour interactively. This example analysis reveals slightly higher hit rates with a larger font, and near the centre of the screen, compared to the border regions.

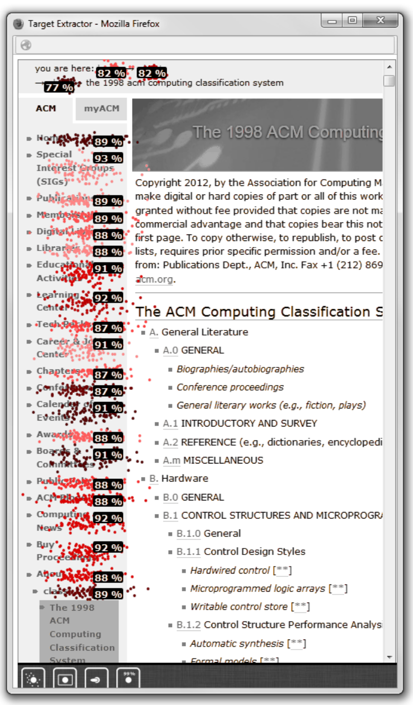

Figure 12. Simulated touch interactions and predicted hit rates for thumb input aiming at the navigation of the ACM Keywords page, which is not optimised for mobile use. Our toolkit predicts low hit rates for a large device. Looking at individual sampled touches yields the explanation: users tend to touch too far to the right. This pattern was learned by the underlying model trained on data from our user study. This example shows how collected data from one page can be transferred to assess possible issues with different layouts and websites.

DISCUSSION

Using our toolkit, we were able to: 1) gain insights into physical constraints (thumb reach); 2) reveal aspects possibly related to mental models (e.g. assumed touchable areas); 3) estimate expected touch behaviour for different layouts and websites (e.g. conference website, wikipedia, ACM website).

Some of these results can be directly translated into recommendations to improve websites: For example, our shopping website could remove the small arrow to avoid possibly confusions regarding touchable areas and activation behaviour (touch, not swiping). In summary, our list elements should more clearly be presented as buttons. Moreover, since users only targeted the product images themselves, not their captions, the images could be extended (with captions overlaid on the bottom of the images). Finally, the observed thumb touches indicate that users avoid reaching to the far corner of the screen, if possible. This should be supported, for example with full width elements at the top of the screen, or by exploring alternative navigation layouts.

Complementary to these insights from our study data analysis, predicting touch behaviour with our models allows us to suggest possible improvements for several other websites: For example, the wikipedia page could increase font size for mobile devices, and use full width buttons for table of contents sections, instead of small text links aligned to the left. Furthermore, our toolkit correctly indicated an unsuitable navigation bar when testing predictions for a website not designed for mobile use. We can thus recommend a more responsive design in this case, for example using a full screen navigation drawer.

In summary, we have gained interesting first insights with our toolkit, even for our very basic study website. These are based on the use of precise touch locations: our toolkit allows to observe and predict touch coordinates, not just activated graphical interface elements. We comment on limitations of this evaluation in the next section.

LIMITATIONS

Prediction accuracy: Our evaluation focused on insights on a higher semantic level, not on repeating quantitative model evaluation schemes known from related work [8, 9, 22, 24]. Therefore, we did not evaluate the accuracy of the models' predictions with new data from a second study. Hence, while our observations match expectations based on related work, we can not assess the precision of predicted absolute values. However, insights derived from relative comparisons of our predictions (e.g. hit rates for different target locations) match observations in the field study.

Hand postures and generalisation: By asking participants to use specific hand postures, we assumed known postures throughout this work. This matches use-cases in which web developers use our toolkit to analyse data from a user study. In contrast, if data is collected on a website in every-day use, we cannot assume known postures. This limits the possible conditions for which we can derive different models. However, it presents no limitation to our modelling approach in general: We can simply train models on data from unknown (and thus possibly multiple) hand postures. In such cases, we lose the ability to compare data between hand postures, but we can still compare models between other factors (e.g. device type, size, screen orientation). Finally, related work showed that pre-trained touch offset models can also be used to predict hand postures [8, 9]. Hence, automatically inferring postures from collected touches could be included in our toolkit in future work, since we already use these models.

Usage by web developers: Some findings may not seem very surprising to a web developer, and our study design can not inform us about how web developers would make use of our toolkit in everyday practice. While this paper focussed on the novel concept and toolkit, a future long-term study with web developers could investigate this. However, our basic evaluation provides a first promising assessment of potential benefits of analysing and modelling touch behaviour on (mobile) websites, and we were able to derive concrete implications for improving the websites in our example application cases.

CONCLUSIONS AND FUTURE WORK

In this paper, we have presented a toolkit for collecting and analysing touch interactions with websites to support mobile web development. In contrast to previous approaches, we also offer predictive tools: We model patterns in users' touch behaviour to simulate expected touch interactions for other layouts or websites. Besides analysing user behaviour, this enables estimations of touch distributions and hit ratios for individual graphical interface elements. A field study indicated that our toolkit's visualisations and predictions support revealing usability issues on the studied website as well as others, and can yield concrete recommendations to improve these pages with respect to touch interaction.

Future work can extend this evaluation to quantify accuracy of touch predicitions in the following way: 1) predict touch distributions based on data from a study; 2) match those touch distributions with actual touches from another set of participants for the same interface.

As a main direction for future development, the toolkit could automatically annotate possibly problematic regions or elements of the analysed website. For example, it could graphically indicate whether specific buttons are hard to reach. Using the predictive power of our toolkit, this could further incorporate automatically generated suggestions for web developers: For example, these suggestions could be generated by iteratively changing the site (e.g. changing css parameters) and checking the result with the models' predictions.

REFERENCES

1. Julie Arendt and Cassie Wagner. 2010. Beyond description: Converting web site usage statistics into concrete site improvement ideas. *Journal of Web Librarianship* 4, 1 (2010), 37–54.

2. Richard Atterer and Albrecht Schmidt. 2007. Tracking the interaction of users with AJAX applications for usability testing. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 1347–1350.

3. Richard Atterer, Monika Wnuk, and Albrecht Schmidt. 2006. Knowing the User's Every Move: User Activity Tracking for Website Usability Evaluation and Implicit Interaction. In *Proceedings of the 15th International Conference on World Wide Web*. 203–212.

4. Shiri Azenkot and Shumin Zhai. 2012. Touch Behavior with Different Postures on Soft Smartphone Keyboards. In *Proceedings of the 14th International Conference on Human-Computer Interaction with Mobile Devices and Services*. 251–260.

5. Patrick Baudisch and Gerry Chu. 2009. Back-of-device interaction allows creating very small touch devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 1923–1932.

6. Joanna Bergstrom-Lehtovirta and Antti Oulasvirta. 2014. Modeling the Functional Area of the Thumb on Mobile Touchscreen Surfaces. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 1991–2000.

7. Jeffrey P. Bigham. 2014. Making the web easier to see with opportunistic accessibility improvement. *Proceedings of the 27th annual Symposium on User Interface Software and Technology* (2014), 117–122.

8. Daniel Buschek and Florian Alt. 2015. TouchML: A Machine Learning Toolkit for Modelling Spatial Touch Targeting Behaviour. In *Proceedings of the 20th International Conference on Intelligent User Interfaces*.

9. Daniel Buschek, Simon Rogers, and Roderick Murray-Smith. 2013. User-Specific Touch Models in a Cross-Device Context. In *Proceedings of the 15th International Conference on Human-Computer Interaction with Mobile Devices and Services*. 382–391.

10. Daniel Buschek, Oliver Schoenleben, and Antti Oulasvirta. 2014. Improving Accuracy in Back-of-Device Multitouch Typing: A Clustering-based Approach to Keyboard Updating. In *Proceedings of the 19th International Conference on Intelligent User Interfaces*. 57–66.

11. Tabatha A Farney. 2011. Click analytics: Visualizing website use data. *Information Technology and Libraries* 30, 3 (2011).

12. Mayank Goel, Leah Findlater, and Jacob O. Wobbrock. 2012. WalkType: Using Accelerometer Data to Accommodate Situational Impairments in Mobile Touch Screen Text Entry. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2687–2696.

13. Niels Henze, Enrico Rukzio, and Susanne Boll. 2011. 100,000,000 Taps: Analysis and Improvement of Touch Performance in the Large. In *Proceedings of the 13th International Conference on Human Computer Interaction with Mobile Devices and Services*. 133–142.

14. Christian Holz and Patrick Baudisch. 2010. The Generalized Perceived Input Point Model and How to

Double Touch Accuracy by Extracting Fingerprints. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 581–590.

15. Christian Holz and Patrick Baudisch. 2011. Understanding Touch. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2501–2510.

16. Jason Jong, Jeffrey Heer, Sarah Wterson, and James Landay. 2001. WebQuilt: A Proxy-based Approach to Remote Web Usability Testing. *ACM Trans. Inf. Syst.* 19, 3 (July 2001), 263–285.

17. Florian Mueller and Andrea Lockerd. 2001. Cheese: Tracking Mouse Movement Activity on Websites, a Tool for User Modeling. In *CHI '01 Extended Abstracts on Human Factors in Computing Systems*. 279–280.

18. Michael Nebeling, Maximilian Speicher, and Moira Norrie. 2013. W3touch. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2311.

19. Alexander Ng, Stephen a. Brewster, and John H. Williamson. 2014. Investigating the effects of encumbrance on one- and two- handed interactions with mobile devices. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2014), 1981–1990.

20. Robert W. Reeder, Peter Pirolli, and Stuart K. Card. 2001. WebEyeMapper and WebLogger: Tools for Analyzing Eye Tracking Data Collected in Web-use Studies. In *CHI '01 Extended Abstracts on Human Factors in Computing Systems*. 19–20.

21. Daryl Weir. 2014. *Modelling Uncertainty in Touch Interaction*. Ph.D. Dissertation. University of Glasgow.

22. Daryl Weir, Daniel Buschek, and Simon Rogers. 2013. Sparse Selection of Training Data for Touch Correction Systems. In *Proceedings of the 15th International Conference on Human-Computer Interaction with Mobile Devices and Services*. 404–407.

23. Daryl Weir, Henning Pohl, Simon Rogers, Keith Vertanen, and Per Ola Kristensson. 2014. Uncertain text entry on mobile devices. *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2014), 2307–2316.

24. Daryl Weir, Simon Rogers, Roderick Murray-Smith, and Markus Löchtefeld. 2012. A user-specific machine learning approach for improving touch accuracy on mobile devices. In *Proceedings of the 25th annual Symposium on User Interface Software and Technology*. 465–476.

25. Ying Yin, Tom Y. Ouyang, Kurt Partridge, and Shumin Zhai. 2013. Making Touchscreen Keyboards Adaptive to Keys, Hand Postures, and Individuals - A Hierarchical Spatial Backoff Model Approach. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. 2775–2784.