EmojiZoom: Emoji Entry via Large Overview Maps 😄🔍

Henning Pohl, Dennis Stanke, Michael Rohs
FG Mensch-Computer Interaktion
University of Hannover
Hannover, Germany
{henning.pohl, michael.rohs}@hci.uni-hannover.de

ABSTRACT
Current soft keyboards for emoji entry all present emoji in the same way: in long lists, spread over several categories. While categories limit the number of emoji in each individual list, the overall number is still so large, that emoji entry is a challenging task. The task takes particularly long if users pick the wrong category when searching for an emoji. Instead, we propose a new zooming keyboard for emoji entry. Here, users can see all emoji at once, aiding in building spatial memory where related emoji are to be found. We compare our zooming emoji keyboard against the Google keyboard and find that our keyboard allows for 18 % faster emoji entry, reducing the required time for one emoji from 15.6 s to 12.7 s. A preliminary longitudinal evaluation with three participants showed that emoji entry time over the duration of the study improved at up to 60 % to a final average of 7.5 s.

ACM Classification Keywords
H.5.2 Information Interfaces and Presentation: User Interfaces—*Input devices and strategies, Interaction styles*

Author Keywords
Text entry; emoji; soft keyboard; zooming user interfaces; mobile input; interaction technique; spatial memory

INTRODUCTION
Over the last years, the use of emoji has risen in popularity and dedicated emoji keyboards are now available on all major mobile platforms. Emoji are pictographic Unicode characters, such as 😘, 🚣, or 🐰. While they are shown on the screen as graphics, they are still text in nature and can thus be used in text messages, filenames, tags, or comments. Yet, there has so far been no research on text entry methods for emoji. So while researchers have come up with a large number of layouts and methods to optimize text entry speed for the set of latin characters (for a survey see, e.g., [9]), emoji keyboards all follow just one principle: selection from large lists (one list per category of emoji). This makes emoji entry a linear search task, which is increasingly problematic as the number of emoji grows.

MobileHCI '16, September 06–09, 2016, Florence, Italy
ACM 978-1-4503-4408-1/16/09…$15.00
DOI: http://dx.doi.org/10.1145/2935334.2935382

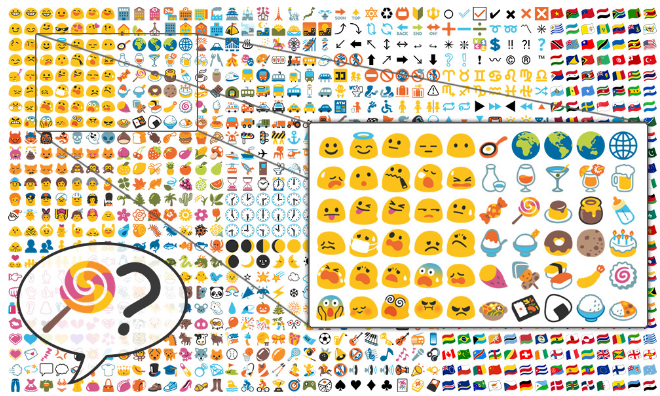

Figure 1. We present a novel keyboard for entry of emoji characters that is built around zooming. Instead of swiping through a list, users zoom to the area of the emoji and then select it from the zoomed-in view. Users can still easily explore all available emoji by panning the zoomed-in view.

Instead, we propose applying the principles of zooming keyboards to the problem of emoji text entry. As shown in Figure 1, users are presented with a zoomed out view of all emoji. They then zoom to the area where they spot or assume their target emoji and select it in this zoomed-in view. Such a technique is now possible with the increasing prevalence of high-resolution screens in phones. Even when we render many small emoji, each emoji still occupies dozens of pixels and is distinguishable from others. Our proposed method, EmojiZoom, has several advantages over selection from emoji lists:

- It shows all emoji at once, giving users an immediate idea of how many emoji are available.
- It builds spatial memory as groups of related emoji are always found in the same region (e.g., all smileys are found in the top left corner).
- It allows for exploration at multiple scales—users can zoom in slightly to explore the general overview or zoom in all the way to more closely explore a certain region.

In this paper, we first give a short introduction to emoji, followed by a description of EmojiZoom. We evaluate EmojiZoom in two ways: (1) a lab study with 18 participants that compares it against the Google keyboard, and (2) a preliminary longitudinal evaluation with three participants. In the lab study we find that zooming emoji entry allows for 18 % faster emoji entry, yet has no higher error rate than the Google keyboard. EmojiZoom also was preferred by most participants in a post-study poll. Our longitudinal evaluation shows that users can further improve performance. While participants in our lab study averaged 12.7 s per emoji with our keyboard, participants in our longitudinal evaluation only needed 7.5 s at the end with one participant as fast as 5.8 s.

EMOJI 101

In the late 90s the first emoji were created in Japan for *NTT DoCoMo*. Emoji allowed sending small pictograms to other phones by only transmitting two bytes—the corresponding character code. The Japanese origin of emoji still manifests itself in emoji such as 🏣, 🗾, 🎏, or 🎍. In 2010, the Unicode consortium standardized 916 emoji (excluding flags) in version 6.0 of the Unicode standard, promoting many earlier characters to the status of emoji. With Unicode standardization, interoperability between systems was secured—a necessary prerequisite for the rise of emoji to broad popularity.

Example data for emoji uptake is available from Instagram. On their platform, they saw a sharp rise in emoji usage from 0% to 20% within less than half a year of the introduction of the iOS emoji keyboard[1]. Currently, about 40% of Instagram messages contain emoji, and this number is even higher in some markets (e.g., more than 60% in Finland).

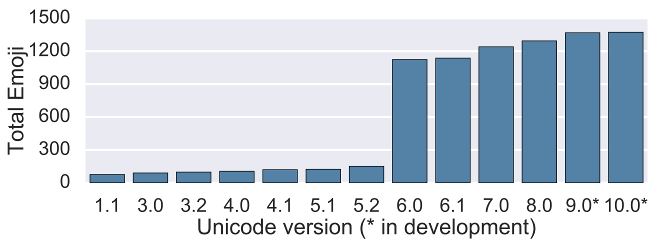

Figure 2. Emoji were first specified in version 6.0 of the Unicode standard (with some characters retroactively promoted to emoji). Since then, that number has continuously grown. The total given here is a conservative number as it does not include every possible combination of compound emoji). For a list of included emoji per version see http://emojipedia.org/unicode-{VERSION}.

The space of emoji is currently moving rapidly. Which emoji are available on a given mobile phone can change with every update. Overall, the number of specified emoji has been rising steadily (see Figure 2). This is partly due to compound emoji such as the country flags. Here region code emoji characters are entered, but a flag is rendered: 🇺 + 🇸 ⇒ 🇺🇸. There are currently 249 official country codes that are potential emoji flags and some systems support additional ones, such as the EU flag. Some platforms also support skin tone modifiers or family combinations (such as 👨‍👩‍👧‍👦 or 👨‍👦‍👨), further increasing the number of emoji that can be entered.

Emoji keyboards on mobile platforms currently all present the available emoji in large, scrollable lists. Instead of having one big list, emoji are split into multiple categories and users pick the list to select from using, e.g., tabs. While keyboards currently only provide a subset of the specified emoji, that subset is also growing with every update. For example, the *iOS* 9.1 keyboard introduced an additional 150 emoji[2]. With ever larger numbers of emoji to choose from, selection via a list is bound to become problematic (imagine having to pick Chinese character from a list when composing a text).

[1]http://instagram-engineering.tumblr.com/post/117889701472/emojineering-part-1-machine-learning-for-emoji
[2]http://blog.emojipedia.org/ios-9-1-includes-new-emojis

RELATED WORK

Emoji text entry so far is an area not well explored. However, communication with emoji is part of the more general area of emotional communication. There are also links to other pictographic systems and emoticons (which can be seen as a precursor to emoji).

Several papers by Cho et al. explore aspects of pictogram use in a *"pictogram email system"* (set up specifically for communication between children) [6, 5, 4]. Their work focuses on semantic relevance of pictograms, which they establish from tags associated with the pictograms (gathered from a survey). By enabling users to find the pictogram that best matches their *intended* message, they hope to reduce ambiguity in pictogram communication. Using pictograms for communications is also a central aspect of the *picoTrans* system [16]. Here users select pictogram sequences which the system translates to text in a target language. This is similar to the use of picture dictionaries for travelers, which allow for limited communication (by pointing at images) in the absence of a shared language.

There has been much interest in how emotion is communicated over textual channels. For example, Hancock et al. studied chat conversations and found that participants were able to detect emotion state from just text [7]. They found that participants did only rarely use emoticons and saw no difference in emoticon use between happy and sad conditions. Emoticon frequency was also investigated by Tossell et al., who logged text messages of 21 participants over a period of half a year, but found that only about 4% of messages contained any emoticons [17]. In an earlier study, Rivera et al. observed impact of emoticons on group-decision making over chat, but did not find an effect [15]. However, more recently, Janssen et al. investigated whether using more emoticons would increase *perceived intimacy* between chat participants, which was indeed the case [8]. While results on emoticon usage are mixed, it is not clear whether this directly translates to emoji. As described earlier, e.g., Instagram has seen fast uptake of emoji use when they were made available.

Enabling emoji use in text is but one approach to give users more expressiveness in their communication. Another approach is kinetic typography, where text is animated to convey emotion [11, 13, 10]. Such animations can, e.g., be used to convey that the sender of a text is shouting at the recipient by making the text *jump at* the reader. In more recent work, Buschek et al. use sensor readings gathered from a phone during text entry to distort the typed text [2]. This enables users to personalize their texts, and readers to, e.g., infer how active the sender was during composition (writing while moving results in more *shaky* text).

We are not the first to explore zoom interactions for keyboards. Starting with *ZoomBoard* [14], there have been several zooming keyboards [3, 12]. The focus of those keyboards has so far been to allow entry of latin characters on very small keyboards (such as on smartwatches). Instead of showing few characters on a small screen, we show many characters on a high-resolution screen. Both approaches solve a mapping problem where the number of symbols and the available space are mismatched.

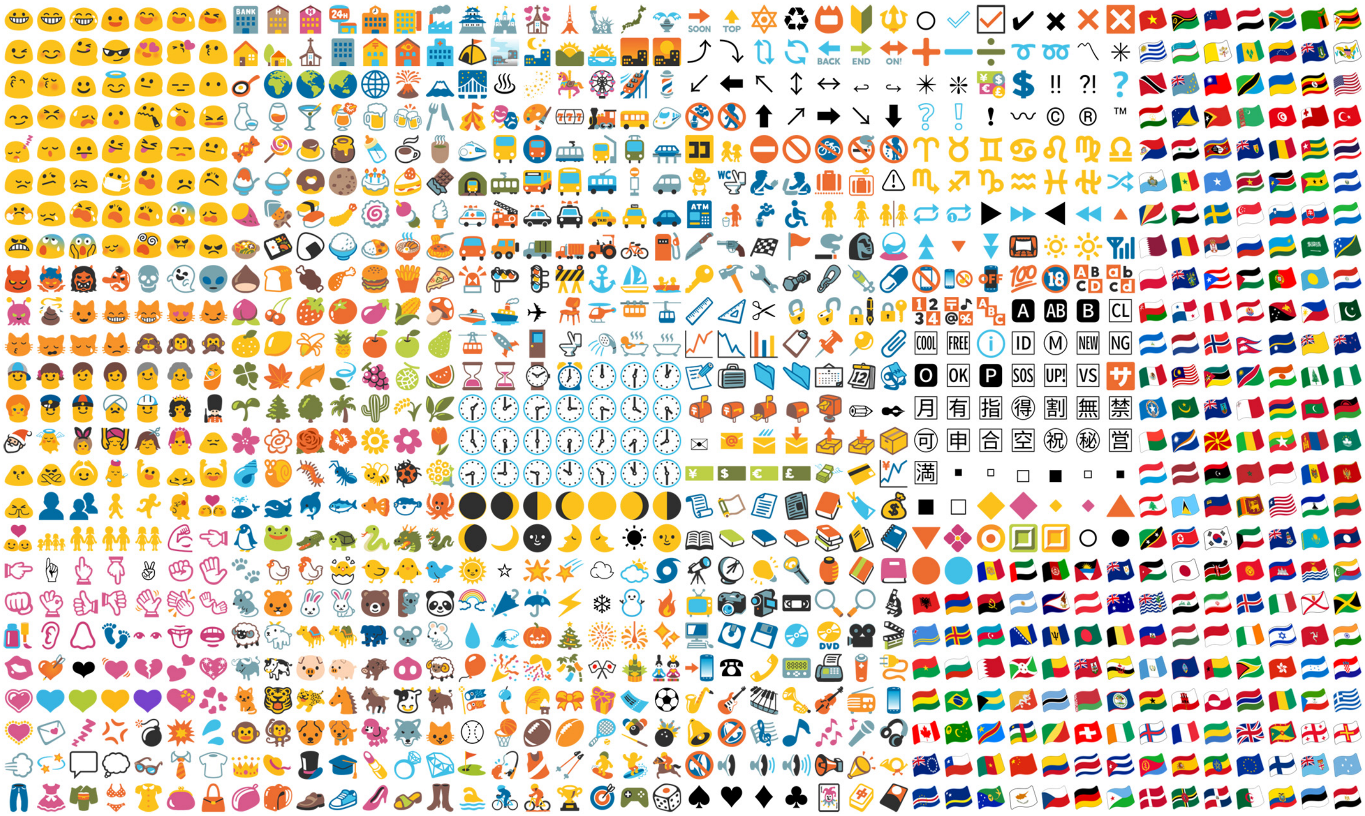

Figure 3. All emoji as shown in the zoomed out view of EmojiZoom. Emoji are arranged in a 42×25 grid along a snaking path, as illustrated in Figure 5. On the Nexus 5 we used in the study this whole grid is 6 cm wide: each emoji occupies about 1.4 mm. As the Nexus 5's screen has 445 ppi this gives about 25 px per emoji—enough to identify many details.

EMOJIZOOM: A ZOOMING EMOJI KEYBOARD

For our design, we forked the Google keyboard[3] (shown in Figure 4) and replaced the emoji components with our own. While this restricts the range of possible designs, it allows for a fair comparison as we retain the overall size. We also render emoji at the same size when zoomed in, as not to introduce a confounding factor in emoji selection. The QWERTY view stays the same, so users see an identical initial view when they activate the keyboard. In the emoji view, however, we remove the list control and categories and only render a grid of all available emoji (see Figure 3). We keep the lower bar, but add a *zoom out* button and the backspace button to it (see Figure 6). Users can zoom into the grid by either tapping on it or using two-finger gestures as in, e.g., map apps. Zooming out is possible via pinching, or by tapping on the *zoom out* button. While tapping zooms in a preset amount, the two-finger zoom allow users to pick any zoom factor.

When zoomed in, users can pan the view by dragging inside the emoji grid. If at or beyond 75 % zoom, a tap on an emoji selects it and returns the grid to the zoomed out view. We set 100 % zoom so that emoji are rendered at about the same size as on the Google keyboard. However, we reduce whitespace between emoji and can thus fit more emoji on the screen. Where the Google keyboard shows 7 emoji in a row, we can fit up to 9 emoji in a row.

[3]https://android.googlesource.com/platform/packages/inputmethods/LatinIME

When users tap to zoom in, we apply a zoom region interpolation as in *ZoomBoard* [14]. This tries to resolve the ambiguity of a zoom action: should the selected point after zooming in be under the finger or in the center of the screen? The applied interpolation is a compromise between both approaches and yields a center position between the two extremes. However, this also means that if users press exactly on an emoji in the zoomed-out view, that emoji is not under their finger in the zoomed-in view. They have to move their finger again to select the desired emoji. We believe this is not much of a tradeoff though, as the large number of emoji makes it hard to precisely select one in the zoomed-out view anyway. Instead, the more viable strategy is to tap in the vicinity of the desired emoji. Users then need to reacquire the emoji in the zoomed-in view (where it should be close to the center) and touch it again.

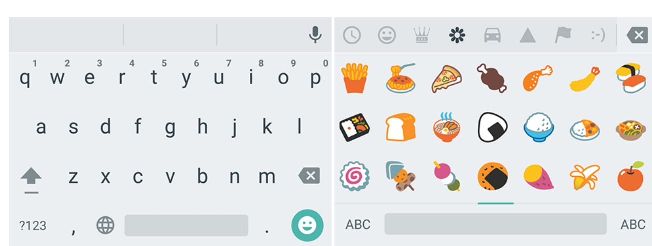

Figure 4. We compare against the *Google keyboard* on the *Nexus 5*. Here, emoji mode is activated with a button in the lower-right. Once in emoji mode, emojis are shown in a paging control where users swipe left/right to move to the next page. Tabs enable jumping to different categories.

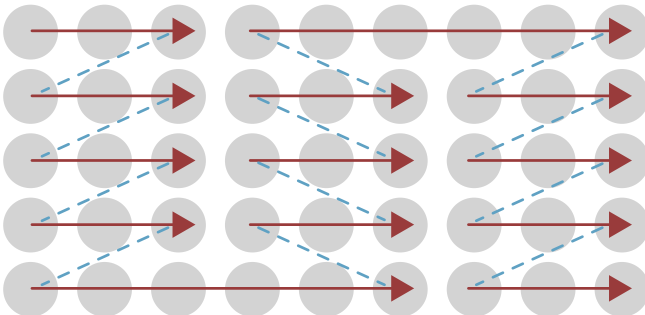

Figure 5. In order to map emoji to our grid, we use a snaking space-filling algorithm. Emoji are first ordered according to the Unicode standard's definition. Columns are then filled either from top to bottom or from bottom to top, alternating every column.

One critical aspect in EmojiZoom is the arrangement of emoji inside the grid. The Unicode standard itself defines an ordering for emoji, yet this only prescribes a one-dimensional order. We hence chose to arrange emoji in columns according to this ordering. The grid is then filled in a snake-like pattern (see Figure 5), thus column direction is reversed every column. Inside each column, emoji are just ordered from left to right.

EVALUATION

With EmojiZoom implemented, we set out to determine whether emoji entry is faster with it. For comparison, we picked the default Google keyboard (version 4.1.23153.2501950) on a Nexus 5, running Android 5.1.1. The screen resolution on the Nexus 5 is 1920×1080 px with 445 ppi. The default keyboard offers 1050 emoji, split into 6 categories. Note that while our keyboard gives no preference to common or recent emoji, the Google keyboard employs a mechanism to facilitate entry of common emoji. For this purpose the Google keyboard maintains a list of recently used emoji and also remembers the last used page per category. If users only enter emoji from a small number of pages, this approach often presents them with the target page when opening the keyboard or switching categories.

Participants

We recruited 18 participants (4 female, age 21–26, $\bar{x} = 23.4$, $\sigma = 1.5$) for the study via social media. All but one participant owned a smartphone and 11 of them ran some version of Android on their phone. Participants using an iPhone all ran iOS version 9.2.1, while Android use was split between 4.x (5 participants), 5.x (5 participant), and 6.x (1 participant). No participant used a custom keyboard, thus all were most familiar with the default keyboard on their phone. Emoji design can vary between different phones, even when running the same operating system. While we used the Google emoji style, all iOS users instead used the Apple style on their phones. In fact, the Google emoji style was only used on five of the participants' phones. Four participants using Android instead used the Samsung design, one participant used the LG design, and one Android user had the Apple style emoji on his phone. Note that while this shows a wide range in different keyboards and designs between participants, the basic principle of selection from an emoji list is identical in all those devices.

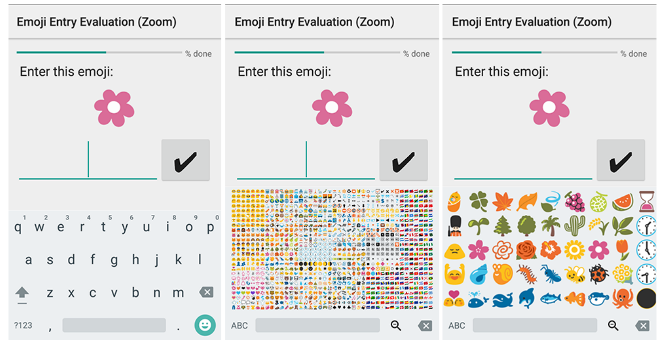

Figure 6. Layout of our evaluation application while testing EmojiZoom. Participants are shown the emoji to enter at the top of the app. In each trial, they need to activate emoji mode, find the respective emoji and click the commit button. Committing also switches the keyboard back to QWERTY mode.

Thirteen participants stated that they often or very often use emoji in their daily life. The primary use (according to 16 participants) is in chats and instant messaging. Two participants said they also use emoji in social media and one participant respectively indicated use in emails and forums. Overall, emoji were described as useful or very useful by 14 participants. Only two participants stated to see no use for emoji.

Procedure

Evaluation of emoji entry comes with some unique challenges. As the size of the test set is very large (1050 emoji in our case), requiring each participant to enter each of them once, or even multiple times, is not feasible. We can thus only test a small subset of emoji. This, of course, begs the question which emoji to test. To gather data on general emoji entry behavior this leaves us with random sampling.

Unfortunately, this rules out some evaluation procedures. For example, to investigate natural user behavior, a chat study where two participants exchange messages (such as in [7]) would be a more fitting choice. This could also be designed as a longitudinal study that monitors users' chatting behavior (such as in [17]). However, this approach risks that only a small number of emoji are actually typed and does not generate a lot of data as much of the time is spend not entering emoji. Larger amounts of data can be generated by deploying prototypes to an app store (such as in [1]). However, this still would not give control over which emoji are used. Most commonly text entry methods are tested with a task where participants have to copy text verbatim. That is also the approach we chose for our investigation, as this allows for control over which emoji are to be typed.

We hence built a test application (see Figure 6) that presents participants with an emoji and asks them to enter it. Before participants started a session, they were given time to try out the respective keyboard. After entering 10 emoji we consider a participant to be sufficiently familiar with the interface and move on to the main study. If participants took more than one minute to find an emoji in the testing phase we aborted the trial. This was done after we noticed in pilots that some emoji can take a very long time to find, which frustrated participants.

Design

The study was a within-subjects design with keyboard as the only factor. We counterbalanced keyboard order between participants. For each participant, we draw a random set of 50 emoji to enter by sampling with replacement from the set of all emoji. This allows for paired comparisons for each participant as they enter each emoji twice: once with each keyboard. Each participant completed 100 trials, which took approximately 45 minutes.

Results

Overall, we saw that there is no difference between the two keyboards with respect to failure rate (trial was aborted due to taking too long). For both keyboards, about 4 % of the trials were aborted because the participant could not find and enter the required emoji in one minute. We also count a trial as a failure if the wrong emoji is entered. A paired-samples t-test shows no significant difference in failure rate between the Google keyboard (M=4.2 %, SD=3.3) and EmojiZoom (M=4.4 %, SD=2.7); $t(17) = -0.24$, $p > 0.8$. However, EmojiZoom was faster than the Google keyboard (see Figure 7). A paired-samples t-test shows a significant difference in retrieval time between the Google keyboard (M=15.6 s, SD=2.9) and EmojiZoom (M=12.7 s, SD=1.9); $t(17) = 3.49$, $p < 0.01$. This is an 18 % increase in emoji entry speed even though participants had no familiarity with methods like EmojiType. On the other hand, many had a lot of experience entering emoji with a standard category-based keyboard.

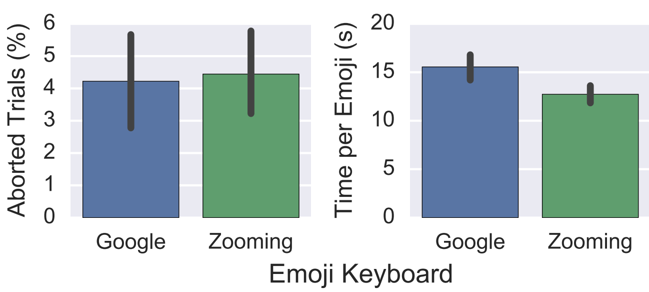

Figure 7. The Google keyboard and EmojiZoom do not differ with respect to the number of failed trials (4.2 % for the Google keyboard and 4.4 % for EmojiZoom). On average it took participants 15.6 s (10.7 s median) to enter an emoji with the Google keyboard. With EmojiZoom, participants only needed 12.7 s (8.1 s). This is significantly faster and about an 18 % increase in speed. Error bars show 95 % CI.

When we look at how people used EmojiZoom, we can see some patterns emerge. An interesting question, e.g., is how users choose between zoom levels: do they only zoom in as much as possible and skim that view, or do they make use of multiple zoom levels? As shown in Figure 8, the maximum zoom level is dominating the statistics. Note that this view excludes the initial zoomed-out view, thus any occurrence of that zoom level in the histogram is due to users zooming out again. This could, e.g., be due to selecting the wrong initial region and backtracking to the start in order to jump elsewhere in the emoji grid. We can also see, though, that medium zoom levels were chosen quite regularly as well. This shows that users take up the chance to zoom in slightly, explore a region of emoji, and only then zoom in more to make the final selection.

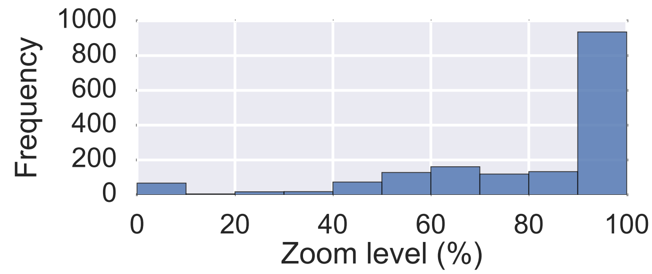

Figure 8. This histogram of zoom levels shows participants zooming behavior, excluding the initial zoomed-out view. Most of the time they chose to zoom in all the way, yet sometimes that chose a medium zoom level to aid in exploration of a smaller region.

We also asked participants to rate six statements on a Likert scale after completion of the study (see Figure 9). Asked to rate the quality of EmojiZoom, all participants gave favorable ratings to the ordering. The majority of participants also preferred EmojiZoom to the Google keyboard. As most participants did not have much more experience with the Google keyboard than with EmojiZoom, this shows that EmojiZoom makes the better impression. Most participants could imagine using a zooming keyboard for emoji entry in the future.

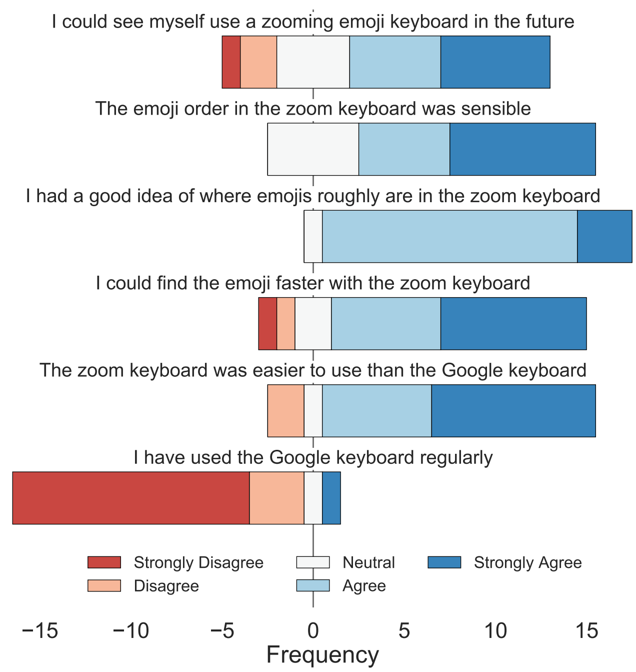

Figure 9. Participants rated six statements on a 5-point Likert scale in our exit interview. Results show preference for EmojiZoom.

Interestingly, several participants stated that the ordering of emoji in EmojiZoom was better than the one in the Google keyboard. Instead of just using the order as specified in the Unicode standard, the Google keyboard moves some emoji around. For example, 🔇, 🔈, 🔉, 🔊, 📢, and 📯 follow after one another in the Unicode ordering, yet are split over two different categories in the Google keyboard. Of course, any rigid ordering for emoji is flawed, as some emoji can have several interpretations (e.g., 🐎 could be put with other animal emoji or with other activity emoji, where 🏇 can be found).

Discussion

The results show that emoji entry with a zooming keyboard is a viable approach. In fact, EmojiZoom outperformed the Google keyboard by 18 %. Users also preferred using EmojiZoom and stated that the ordering was better than in the Google keyboard. By having all emoji visible at once, some of the problems of emoji categories are avoided. Where a search for an emoji in the Google keyboard carries a heavy penalty if a wrong category is picked, moving on to a different region is easier and faster in EmojiZoom. A limitation of our study is that the participants were all comparably young. We would expect performance to be lower for older users who might not be able to see the small emoji that well.

LONGITUDINAL EVALUATION

Our study showed that zooming emoji entry outperforms emoji entry with the Google keyboard. However, participants in that study only got to use our keyboard for a short time. To investigate whether performance could improve further, we ran a limited additional longitudinal evaluation.

We recruited three additional participant for this evaluation (all male, age 25, 28, and 42 years). One participant installed EmojiZoom on his own phone (Samsung Galaxy S6), while the other two used one of our Nexus 5 phones. The Samsung phone had a slightly higher resolution screen than the Nexus 5 at 2560×1440 px with 577 ppi and also ran Android 5.1.1. As this allows for slightly clearer rendering of emoji in the zoomed out view, its performance numbers (bottom plot in Figure 10) are thus not directly comparable. They each ran the evaluation several times over the following weeks, yielding 12, 19, and 20 runs respectively. This time, we only included our keyboard and did not gather longitudinal data for the Google keyboard. Each run, a random emoji test set was used to prevent repetition of a small subset of emoji over the course of the study.

Results

As shown in Figure 10, we observed steady improvement over the course of the study for all three participants. For example, while it took participant three 14.5 s per emoji in the first run, performance improved to 5.6 s in the last run. This is a speedup of almost 60 %. We can run a statistical test to check whether performance in the end is significantly better than performance in the beginning. An independent samples t-test shows a significant difference in retrieval time between the first two runs and the last two runs for all three participants with $p \leq 0.001$. Linear regression per participant also is significant with $p \leq 0.01$ for all three.

Discussion

While we have only tested the impact of prolonged use with three participants, we saw clear improvements in a short time period. It seems like EmojiZoom indeed builds spatial memory and enables users to zoom into the rough vicinity of their desired emoji faster. However, additional studies are necessary to confirm this improvement, due to the small sample size of this study. We are working on making EmojiZoom available to a wider range of users and hope to gather in-situ data over a prolonged period of time.

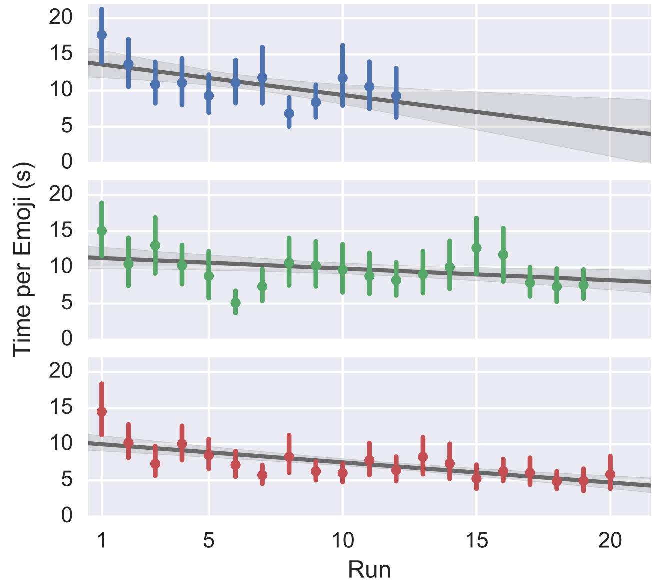

Figure 10. Three additional participants used EmojiZoom over the course of up to twelve days, completing the study 12/19/20 times (each time with a different emoji test set). Performance improved significantly (e.g., by ~60 % for the third participant). Error bars show 95 % CI.

EMOJI-LEVEL ANALYSIS

So far we have only looked at overall selection time. But we were also curious whether there were any patterns in the data showing that EmojiZoom worked better for some emoji than for others. For this analysis we pool together all collected data for EmojiZoom: from the lab study and all trials from the longitudinal evaluation. This gives us a total of 3423 trials, of which 3235 were successful (95 %).

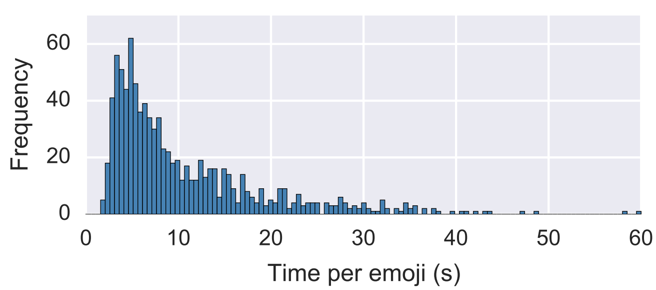

Figure 11. The distribution of selection times for emoji has a long tail. About 70 % of selections are done within 10 seconds, but some selections take much longer.

We first take a look at selection times. As the distribution in Figure 11 shows, most selections are fast, but a long tail is visible as well. While 50 % of trials completed within 5.8 s, it took 34.8 s until almost all (95 %) selections finished. Note that we stopped trials after 60 seconds. Had we allowed participants to continue, the tail would be longer. But as this data only makes up 5 % of the trials overall, the impact on the distribution would not be strong. However, we observe that this distribution is not equal for all emoji, but varies for different ones.

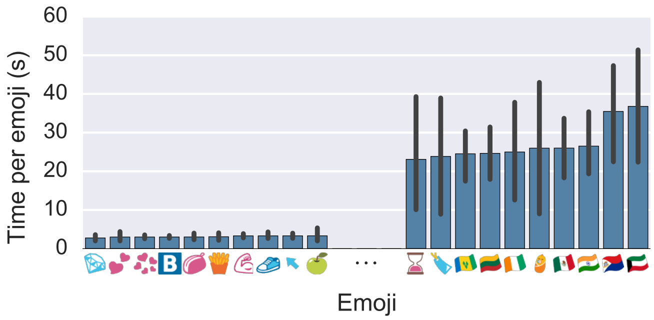

Figure 12. Best and worst performing emoji with respect to selection time. We only consider emoji for which we have collected at least 5 trials. Error bars show 95 % CI.

As shown in Figure 12, some emoji were always selected quickly, while others tend to lead to slow selection times. For this comparison we only consider emoji with at least 5 recorded trials. Among the ten worst performing emoji we find 7 that are flag emoji. Checking back with the whole dataset, we find that while it took 22.1 s on average to enter a flag, it only took 9.6 s to enter other emoji. An independent samples t-test shows a significant difference between selection times of flags and other emoji; $t(3419) = 20.9$, $p < 0.0001$.

The likely reason for this strong difference is the visual similarity between all the flags. As can be seen in Figure 3, the flags form a large block on the right side of the grid. While some flags, such as 🇲🇰, stand out a bit, many flags are rather similar. An extreme example are flags such as 🇦🇨, 🇦🇺, 🇨🇰, 🇲🇸, 🇳🇿, 🇵🇳, 🇸🇭, 🇹🇦, 🇹🇨, and 🇻🇬. The ordering of flags is also not readily apparent as they are sorted by their corresponding region code. Hence, while they are ordered alphabetically, Germany (region code `DE`) is not found near other countries starting with `G`, but next to other countries with region codes starting with a `D`. However, countries' region codes are likely not familiar to users. Even if they were, region codes are not shown in the emoji and are thus not readily accessible.

Emoji	Trials	Success	Emoji	Trials	Success
▪	2	0.0 %	🇬🇳	3	33.3 %
🍳	4	25.0 %	🇰🇵	6	33.3 %
🌗	11	27.2 %	🇨🇫	3	33.3 %
🔻	3	33.3 %	🇸🇨	3	33.3 %
🇳🇱	3	33.3 %	🎰	5	40.0 %

Table 1. Emoji with the highest share of failed trials overall. We only include emoji for which we collected data from at least two trials.

The problem with flags also shows when we look at failed trials (those aborted after 60 s). As shown in Table 1, flags again make up a large share of those emoji. However, with a low share of failed trials overall and a large number of emoji to enter, there is only little data on failures for individual emoji. A larger scale deployment and analysis could uncover clearer patterns in which emoji are more likely to take a very long time and would thus benefit most from keyboard improvements.

Emoji	Trials	Success	Emoji	Trials	Success
🍔	15	100 %	🇹🇻	12	100 %
🍡	12	100 %	😉	11	100 %
➕	12	100 %	♦️	11	100 %
🌾	12	100 %	👾	11	100 %
🕕	12	100 %	🌄	11	100 %

Table 2. Emoji with the lowest share of failed trials overall ranked by the number of trials recorded.

While some trials failed, it is important to note that the majority of emoji were entered successfully each time. For 789 emoji (85 % of our tested emoji), we recorded no failed trial at all. In Table 2, we show the emoji with the largest number of successful trials. This does not take into account selection time, but only shows whether they were selected within the 60 s. Interestingly, this set also contains a flag, the one of Tuvalu. While participants always found that flag, this still took them 18.1 s on average. While this is fast for flags, it is slow compared to other emoji.

CONCLUSION

We have introduced EmojiZoom, an input method for emoji that outperforms existing emoji keyboards built around selection from long lists. Our method shows a clear performance advantage with the potential for further improvement with additional training. Participants also preferred EmojiZoom and found the ordering of emoji superior to the one in the Google keyboard. However, the use of EmojiZoom necessitates a high-resolution screen. But as such screens have become more prevalent in current generation smartphones, methods like EmojiZoom have become viable.

Compared to list-based entry methods, EmojiZoom is better prepared for future growth in the number of emoji. Adding an additional 110 emoji to the evaluated Google keyboard would require adding six pages of emoji. The same emoji could be added to the EmojiZoom grid with two more rows and one more column, which would not make individual emoji much smaller. Effectively, EmojiZoom scales with the square root of the number of added emoji.

Yet there are some possible improvements to EmojiZoom. In the current implementation flags take up a large share of the space. However, most users will only ever need a small subset of the available flags and, as shown, flags are hard to distinguish. Instead of showing all flags, flag selection could be relegated to a second level. Users would select a flag stand-in (which could be larger than regular emoji to make selection easier) which would switch the emoji selection to a dedicated flag mode. Here flags could either be displayed in a second grid or in any other arrangement. For example, it might be suitable to pick flags from a world map or from a list of country names. A similar approach might be possible for the clock or moon phase emoji. This would introduce a hierarchy to EmojiZoom, further strengthening its ability to cope with the growing number of emoji.

REFERENCES

1. Matthias Böhmer, Christian Lander, Sven Gehring, Duncan P. Brumby, and Antonio Krüger. 2014. Interrupted by a Phone Call: Exploring Designs for Lowering the Impact of Call Notifications for Smartphone Users. In *Proceedings of the 32nd annual ACM conference on Human factors in computing systems - CHI '14*. ACM Press, New York, New York, USA, 3045–3054. DOI:http://dx.doi.org/10.1145/2556288.2557066

2. Daniel Buschek, Alexander De Luca, and Florian Alt. 2015. There is more to Typing than Speed: Expressive Mobile Touch Keyboards via Dynamic Font Personalisation. In *Proceedings of the 17th International Conference on Human-Computer Interaction with Mobile Devices and Services - MobileHCI '15*. ACM Press, New York, New York, USA, 125–130. DOI: http://dx.doi.org/10.1145/2785830.2785844

3. Xiang 'Anthony' Chen, Tovi Grossman, and George Fitzmaurice. 2014. Swipeboard: A Text Entry Technique for Ultra-Small Interfaces that Supports Novice to Expert Transitions. In *Proceedings of the 27th annual ACM symposium on User interface software and technology - UIST '14*. ACM Press, New York, New York, USA, 615–620. DOI: http://dx.doi.org/10.1145/2642918.2647354

4. Heeryon Cho and Toru Ishida. 2011. Exploring Cultural Differences in Pictogram Interpretations. In *The Language Grid*, Toru Ishida (Ed.). Springer Berlin Heidelberg, 133–148. DOI: http://dx.doi.org/10.1007/978-3-642-21178-2{_}9

5. Heeryon Cho, Toru Ishida, Satoshi Oyama, Rieko Inaba, and Toshiyuki Takasaki. 2008a. Assisting Pictogram Delection with Categorized Semantics. *IEICE Transactions on Information and Systems* E91-D, 11 (2008), 2638–2646. DOI: http://dx.doi.org/10.1093/ietisy/e91-d.11.2638

6. Heeryon Cho, Toru Ishida, Toshiyuki Takasaki, and Satoshi Oyama. 2008b. Assisting Pictogram Selection with Semantic Interpretation. In *Proceedings of the 5th European Semantic Web Conference - ESWC '08*. 65–79. DOI:http://dx.doi.org/10.1007/978-3-540-68234-9_8

7. Jeffrey T. Hancock, Christopher Landrigan, and Courtney Silver. 2007. Expressing Emotion in Text-Based Communication. In *Proceedings of the SIGCHI conference on Human factors in computing systems - CHI '07*. ACM Press, New York, New York, USA, 929–932. DOI:http://dx.doi.org/10.1145/1240624.1240764

8. Joris H. Janssen, Wijnand a. Ijsselsteijn, and Joyce H D M Westerink. 2014. How Affective Technologies can Influence Intimate Interactions and Improve Social Connectedness. *International Journal of Human Computer Studies* 72, 1 (2014), 33–43. DOI: http://dx.doi.org/10.1016/j.ijhcs.2013.09.007

9. Per Ola Kristensson and Keith Vertanen. 2014. The Inviscid Text Entry Rate and its Application as a Grand Goal for Mobile Text Entry. In *Proceedings of the 16th international conference on Human-computer interaction with mobile devices & services - MobileHCI '14*. ACM Press, New York, New York, USA, 335–338. DOI: http://dx.doi.org/10.1145/2628363.2628405

10. Joonhwan Lee, Soojin Jun, Jodi Forlizzi, and Scott E. Hudson. 2006. Using Kinetic Typography to Convey Emotion in Text-Based Interpersonal Communication. In *Proceedings of the 6th ACM conference on Designing Interactive systems - DIS '06*. ACM Press, New York, New York, USA, 41. DOI: http://dx.doi.org/10.1145/1142405.1142414

11. Johnny C. Lee, Jodi Forlizzi, and Scott E. Hudson. 2002. The Kinetic Typography Engine: An Extensible System for Animating Expressive Text. In *Proceedings of the 15th annual ACM symposium on User interface software and technology - UIST '02*. ACM Press, New York, New York, USA, 81–90. DOI: http://dx.doi.org/10.1145/571985.571997

12. Luis A. Leiva, Alireza Sahami, Alejandro Catala, Niels Henze, and Albrecht Schmidt. 2015. Text Entry on Tiny QWERTY Soft Keyboards. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems - CHI '15*. ACM Press, New York, New York, USA, 669–678. DOI: http://dx.doi.org/10.1145/2702123.2702388

13. Mitsuru Minakuchi and Katsumi Tanaka. 2005. Automatic Kinetic Typography Composer. In *Proceedings of the 2005 ACM SIGCHI International Conference on Advances in computer entertainment technology - ACE '05*. ACM Press, New York, New York, USA, 221–224. DOI: http://dx.doi.org/10.1145/1178477.1178512

14. Stephen Oney, Chris Harrison, Amy Ogan, and Jason Wiese. 2013. ZoomBoard: A Diminutive QWERTY Soft Keyboard Using Iterative Zooming for Ultra-Small Devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems - CHI '13*. ACM Press, New York, New York, USA, 2799. DOI: http://dx.doi.org/10.1145/2470654.2481387

15. Krisela Rivera, Nancy J. Cooke, and Jeff a. Bauhs. 1996. The Effects of Emotional Icons on Remote Communication. In *Conference companion on Human factors in computing systems common ground - CHI '96*. 99–100. DOI:http://dx.doi.org/10.1145/257089.257180

16. Wei Song, Andrew Finch, Kumiko Tanaka-Ishii, Keiji Yasuda, and Eiichiro Sumita. 2013. picoTrans: An Intelligent Icon-Driven Interface for Cross-Lingual Communication. *ACM Transactions on Interactive Intelligent Systems* 3, 1 (2013), 1–31. DOI: http://dx.doi.org/10.1145/2448116.2448121

17. Chad C. Tossell, Philip Kortum, Clayton Shepard, Laura H. Barg-Walkow, Ahmad Rahmati, and Lin Zhong. 2012. A Longitudinal Study of Emoticon Use in Text Messaging from Smartphones. *Computers in Human Behavior* 28, 2 (2012), 659–663. DOI: http://dx.doi.org/10.1016/j.chb.2011.11.012