

DOI:10.1145/2971322

Smartphone mapping apps routinely fail to follow centuries-old mapmaking dynamic consistency principles and practices.

BY HANAN SAMET, SARANA NUTANONG, AND BRENDAN C. FRUIN

Dynamic Presentation Consistency Issues in Smartphone Mapping Apps

THE SEPTEMBER 2012 introduction of the Apple iPhone 5 smartphone and accompanying iOS 6 software replaced a mapping app on Apple's mobile devices based on Google's map data with an app that uses Apple's map data. It also changed Apple's decisions as to what data is displayed (served to the user) in response to queries, especially implicit ones through manipulation of the viewing window. This change led to significant related changes in the user experience with applications that use and serve map data and resulted in closer scrutiny

of mapping apps on mobile devices, as we do here.

Applications on mobile devices (smartphones and tablets) are not the traditional ones where the map is used in a passive manner, as in atlases, including maps that are browsed leisurely. Maps on mobile devices are instead used in an active manner as a tool to enable such tasks as navigation and location finding, using pan and zoom. In this case, accuracy is paramount, and data quality and lack of quality-assurance policies and protocols by Apple in releasing the iOS 6 mapping app became apparent. This resulted in such errors as misplaced towns; see, for example, Dobson,³ Tumblr,³⁵ and Whitney,³⁶ and misclassified areas, as in Tumblr,³⁵ which persists (see Samet et al.²⁷) The public uproar was so great it eventually led to the dismissal of Apple's leader of its new mapping app project.² Most such errors were fixed in subsequent releases of iOS 6 and its successors iOS 7 and iOS 8.

Notwithstanding this resolution, we have occasionally found the iOS 6, iOS 7, and iOS 8 mapping apps to be lacking from the perspective of presentation consistency when deployed on mobile devices like smartphones due to limited screen "real estate." Surprisingly, criticism from such a perspective was rarely leveled before (see Paolino et al.²¹ and Samet et al.²⁶), as we do here through examples of how they also plague other mapping apps.

Our definition of presentation consistency is motivated by centuries-old classical principles and practices used by cartographers derived from the static way maps have been browsed, as well as by the evolving dynamic ways maps are browsed involving manipulation, and have much to do with the platform used to view them (such as an atlas instead of a smartphone).

Dynamic presentation consistency properties, including pan, zoom, full zoom out, and wraparound, are a result of the manipulation actions users take to browse the map and are often achieved by gesturing (but see also Es-



COLLAGE BY ANDRĹJ BORYS ASSOCIATES/SHUTTERSTOCK

perança et al.⁴ and Samet et al.²⁵) Pan and zoom consistency properties correspond to the integrity of the available gesturing actions in terms of retaining label information as an action takes place. The premise is that if a spatial entity has been labeled, then the label should persist as long as the spatial entity remains visible in its entirety. The full zoom-out property reflects the user's desire to view the Earth in its entirety rather than being compelled to apply pan operations to do so. Similarly, the wraparound property is an acknowledgment that the Earth is round and again reflects the user's desire to not have to

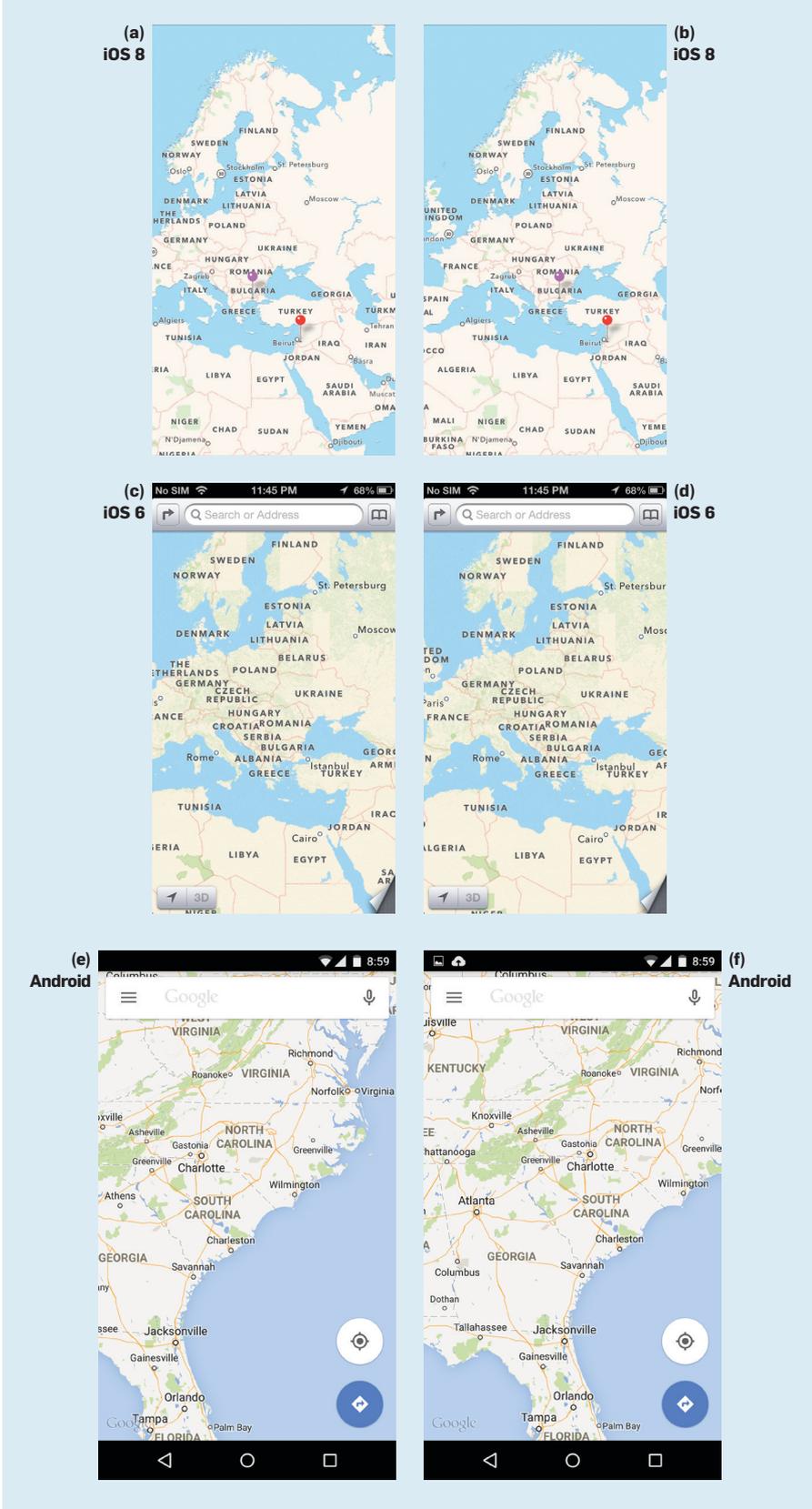
apply pan operations in the opposite direction to be able to view adjacent locations on the map. These properties are the subject of this article.

The classical static presentation consistency properties involve the undesirability of label overlap and having a reasonable label distribution that is aesthetic in nature in a way the user finds aesthetically appealing. Other static consistency properties include hierarchical consistency, which seeks a consistent way of presenting labels of containing entities by requiring they must be included whenever visible in their entirety, while sibling consistency

» key insights

- For pan and zoom-in operations, if a spatial entity is labeled, then the label should persist as long as the spatial entity remains visible in its entirety.
- Smartphone mapping apps must take into account the small form factor that limits the size of screen real estate while still enabling the whole world to be seen in one view without having to pan.
- Updates to mapping apps do not necessarily result in improvements, especially on smartphones where the small form factor limits what can be seen and manipulated when using gesturing user interfaces, unlike command-line and search-box user interfaces.

Figure 1. The result (b) of panning the map (a) to the left (achieved by a swipe to the right) in the iOS 8 mapping app on an iPhone 5. The result (d) of panning the map (c) to the left (achieved by a swipe to the right) in the iOS 6 mapping app on an iPhone 5. The result (f) of panning the map (e) to the left (achieved by a swipe to the right) in the Android mapping app on Lollipop. Note the absence of The Netherlands in (b) and (d) and Athens in (f), demonstrating object panning inconsistency.



corresponds to labeling all visible spatial entities that are at the same level of the mapping object hierarchy. The static presentation consistency properties pertain to the identity of the labels that should be present and how they are presented; these properties are discussed in greater detail in Samet et al.²⁷

The motivation for our study is to understand how to take advantage of the fact that a map provides an efficient way of accessing spatially referenced data when we cannot look at all of it at once. Our observations are based on our experience building the STEWARD,¹⁴ NewsStand,^{13,28,34} TwitterStand,^{9,11,32} PhotoStand,²³ and TweetPhoto⁷ systems and adapting them (especially NewsStand and TwitterStand) to run on smartphones.^{24,29} These systems access such documents as news and photos with a map-query interface, usually by location, and, to a lesser extent, by topic. In these applications, as well as in many related ones, the map on the smartphone helps users anchor and orient answers to queries in which they want to take advantage of spatial synonyms. In addition, we continue to be motivated by the desire to be able to place spatially referenced information on the map (such as icons for topics, image thumbnails,^{17,18,22} names of particular locations, names of people and diseases,¹² mentions of brands,¹ and any other data that lends itself to being classified using an ontology). Note the Gazetteer used to translate textual specifications to geometric specifications can also be considered an ontology. The result is analogous to a mashup, except that, in our case, the mashup is hierarchical in the sense that, as we zoom in on the map, additional spatially referenced information is displayed that was not sufficiently important to be displayed when we zoom out completely. This zoom-out capability is not available in comparable systems like HealthMap^{6,10} for disease monitoring.

Comparison

In terms of devices, we compare the iOS 6 mapping app (initially iOS version 6.1 on iPhone 5 and most recently 6.1.4), the iOS 7 mapping app (iOS version 7.0 on iPhone 4), the iOS 8 mapping app (iOS version 8.1.2 on iPhone 5), the iOS 5 mapping app (iOS version 5.1.1 on

iPod Touch), the Android mapping app (Maps version 8.0.0 on Android 4.3), Google’s iOS mapping app (version 1.0) for Google Maps data (referred to here as the iOS mapping app for Google), and the HERE Maps app on Microsoft’s Windows Phone 8 (HERE Maps version 3.5.481.8 with map data 8.0.50.116). Although each vendor’s mapping apps are similar, they do not always yield the same result. At times, we also use the qualifiers “old” and “new” to distinguish between the versions of iOS 6 that we used in our initial tests (version 6.1 in October 2012) and in our most recent tests (version 6.1.4 in April 2014 and later), respectively. This distinction is necessary because we observed that the algorithms used to implement the various mapping apps are frequently changed for a particular version of the operating system, even if the operating system is not updated. It is especially true for the label-distribution-and-placement algorithms that we often found yield different results for the same queries. This difference in results is due to all queries being transmitted to the map tile server over the Internet, and the server makes the final decision as to what labels will be placed, and where, on the resulting map tiles that are also transmitted by the server. Do not be surprised if you are unable to always repeat our observations. The important takeaway from them is that the undesirable behaviors of some mapping apps should not be taken as absolutes but as indications of what could possibly go wrong. Our comparison also included the iOS apps of Bing Maps, Nokia Maps, ESRI, MapQuest, and OpenStreetMap.^a

The table here outlines the comparison in terms of presentation consistency properties that any user would want satisfied. Such satisfaction is primarily an issue on smartphones where the screen size is small, thereby requiring

^a We include the iOS apps of Bing Maps (version 3.03), Nokia Maps (HERE Maps version 1.8) that increasingly serve as the source for Bing Maps,⁸ ESRI (ArcGIS version 2.3.2), MapQuest (version 3.3.1), and OpenStreetMap, or OSM, where open source map data is the basis of OpenSeaMap version 1.1, which is used here. OpenStreetMap could have also been used as the source map data for the MapQuest app.^{16,19,20} Note that the iOS mapping apps developed by Google, MapQuest, Nokia Maps, OSM, Bing Maps, and ESRI were all tested on iOS version 6.1.

panning and zooming for more information, while not needed on tablets where screens are larger, as described later in greater detail with examples from the various mapping apps. The apps are identified using I5 for iOS 5, I6 for iOS 6, I7 for iOS 7, I8 for iOS 8, A for Android, WP for HERE Maps on Windows Phone 8, IG for iOS of Google, IB for iOS of Bing Maps, IN for iOS of Nokia Maps, IQ for iOS of MapQuest, IO for iOS of OpenStreetMap, and IE for iOS of ESRI. The table denotes whether the property does not (×), partially (P), or holds (✓) for the apps.

Note we do not compare the mapping APIs, as they correspond to a set of features in the programming environments that are designed to make it easy for users to build mapping apps. However, just because a feature is not available in a mapping API does not mean a user cannot deploy a more complex workaround to obtain this functionality in the mapping app being built. Horizontal wraparound is an example operation available in the iOS 6 mapping app but not in the corresponding API, as discussed later. Another example is the proportion of the Earth’s surface that can be viewed at the maximum zoom-out level, which is much greater for the Android mapping app than for its corresponding API.

Note, too, the variation in the relative sizes of the screenshots in some of the figures due to the different devices we used. In particular, the screenshots for the Android and Windows Phone mapping Apps are larger due to a 2.5-inch × 4-inch screen (sometimes a 2.5-inch × 4.3-inch screen for the Android) instead

of a 2-inch × 3-inch screen on the iPhone 4 and iPod Touch devices we used. We used the iPod Touch to perform comparisons with the iOS 5 mapping app that is not available on the iPhone 5, as well as with the iOS mapping apps. However, although the iPhone 5 has a 2-inch × 3.5-inch screen, the scope of the maximum zoom-out level for both landscape and portrait modes vis-à-vis the iPod Touch is unchanged. On the other hand, the iPhone 6 Plus, as of September 2014, with a 2.7-inch × 4.8-inch screen retains the same maximum zoom-out level for the landscape mode while slightly enlarging it for the portrait mode.

Panning Consistency

When panning on the map, the objects, as well as the type of the objects (such as cities, states, countries, and continents), being displayed should be consistent and not disappear or change as long as the underlying space is visible, or “panning consistency.”³³ This property is usually satisfied when the display screen is relatively large, even though panning consistency conflicts arise at times for the smaller form factors, as in smartphones. We found panning consistency to hold for many of the iOS mapping apps, while we were able to find examples of panning inconsistency for each of the current most commonly used mapping apps.

We also encountered this problem on an iPhone 5 running iOS 6 (see Figure 1c and the result of panning it to the left in Figure 1d) but not on an iPhone 4 running iOS 7. We attribute the latter, not seeing a misapplied label on

Consistency property comparison of mobile mapping apps.

Properties	Apps											
	I6	I7	I8	I5	A	IG	IB	IN	WP	IQ	IO	IE
Hierarchical C	×	×	×	P	P	P	✓	P	×	×	✓	×
Sibling C	×	×	×	✓	×	×	×	×	×	×	✓	×
No Overlaps	✓	✓	✓	✓	✓	✓	P	✓	✓	✓	✓	×
Label Distribution	×	×	P	✓	✓	✓	×	×	×	✓	✓	✓
Panning C	×	×	×	✓	✓	✓	✓	✓	×	✓	✓	✓
Zoom C	×	×	×	✓	×	×	✓	✓	×	✓	✓	P
H Wraparound	✓	✓	✓	×	✓	✓	×	✓	✓	✓	✓	✓
V Wraparound	×	×	×	×	×	×	×	✓	×	✓	×	×
Full Zoom Out	P	P	P	✓	×	P	✓	×	✓	×	✓	✓

an iPhone 4, to the different position of the United Kingdom label on an iPhone 4 running iOS 7 from the one on an iPhone 5 running both iOS 6 and iOS 8, as the iPhone 4 screen is smaller than the iPhone 5 screen.

This particular problem did not arise in the Android mapping app, as the positions of the labels are not rigid in the sense they can change as zoom levels are changed; see, for example, the varying positions of the United Kingdom label in Figure 2. Nevertheless, we did find examples using other objects (such as cities) where panning consistency does not hold for the Android mapping app on Lollipop; for example, Athens, GA, vanishes as the map in Figure 1e is panned to the left, even though there is room for it, resulting in Figure 1f. A similar example using other cities can be constructed for the Windows Phone App (such as Rotterdam in The Netherlands, although not shown here). Note in all these examples, where the objects were represented on the map with a label corresponding to its name and a symbol, as with a hollow circle at its geographic position (such as a city at a zoom level where it can be represented as a point), map app designers would disregard edge cases where the symbol is outside the display window. Most of the time the label is not present in such a case, but we do not deem this absence after a panning operation an instance of panning inconsistency.

As an example of the panning inconsistency of the types of the objects map app designers would seek to avoid, consider the portion of the world map including Africa and Europe using the iOS 7 mapping app on an iPhone 4 in Figure 3a. Note the implicit border in the middle of the display screen so the left half consists of names of countries and the right half consists of names of cities. This implicit border is already an ominous sign, as the map should display the same type of information at all locations. Panning the map to the right (achieved by a swipe to the left) yields Figure 3b where we see the resulting map now consists primarily of names of cities, with a few names of countries on the extreme left and right. Panning the map further to the right (again achieved by a swipe to the left) yields Figure 3c, where the resulting map now consists primarily of names of countries with a few names of cities in which case the name of the containing country is usually given; that is, the map is hierarchically consistent. This behavior is not what is expected, in that the types of the objects that are displayed should be consistent as users pan. It can be said the original data was not consistent so users should not expect it to be consistent once they pan. We do not agree and were not able to repeat this example on the iOS 6 and iOS 8 mapping apps on an iPhone 5.

As an aside, note the presence of Dji-

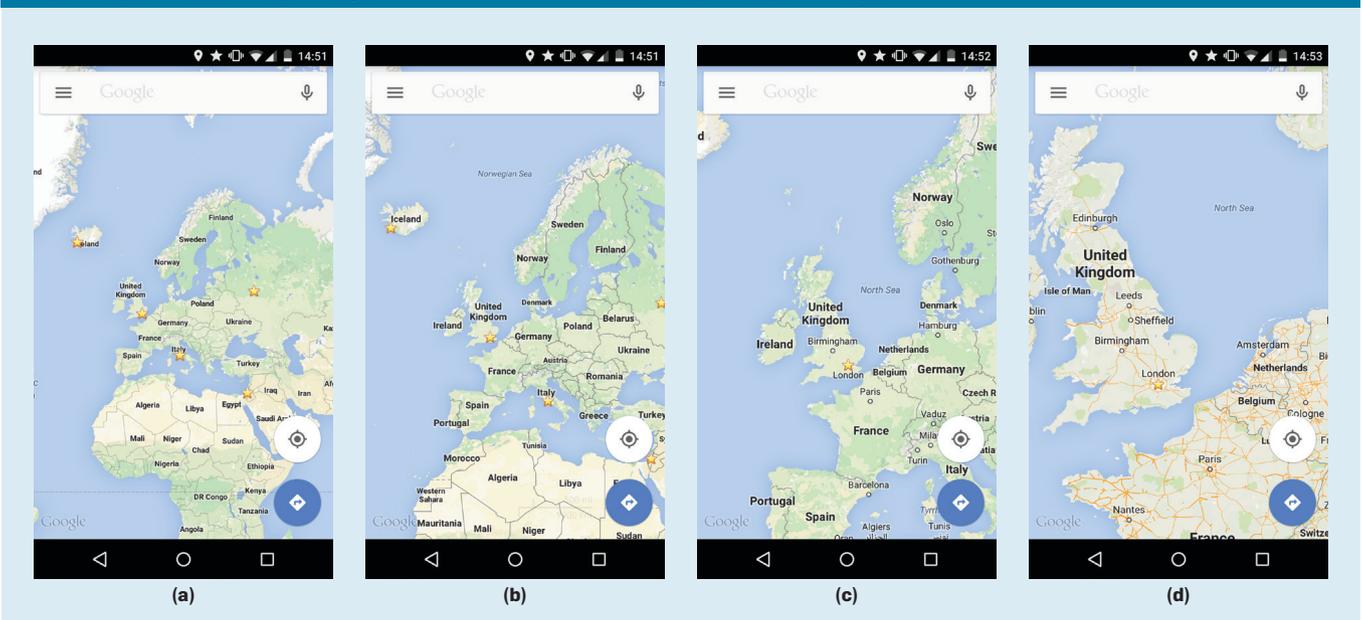
bouti in Figure 3c using a style that capitalizes the first letter, as is used for cities. However, Djibouti is the name of both a city and a country and thus should probably be presented here in upper-case letters. If we zoom in sufficiently in the iOS 7 mapping app (also in iOS 6 and iOS 8), then Djibouti is displayed twice—once in upper case corresponding to its interpretation as a country and once where only the first letter is capitalized, corresponding to its interpretation as a city.

Zoom Consistency

As the user zooms in or pans, names of places that are displayed should continue to be displayed as long as the area they span is visible in its entirety, or “zoom consistency” and “pan consistency.”³³ Names of large containers (such as “United States”) may vanish altogether as the zoom gets very deep. Note the distinction from hierarchical consistency, which deals with just one map view, while zoom consistency is concerned with multiple successive map views.

Consider Figure 4, which shows the failure of zooming consistency to hold for the Android mapping app as we zoom into Europe. The zoom starts in Figure 4a, where Croatia is labeled, while Slovenia is not, even though there is room for its label. Subsequent zooming leads to labeling Zagreb in Croatia in Figure 4b. Further zooming in Figure 4c finds that Ljubljana, the capital

Figure 2. Varying positions of the United Kingdom label in the Android mapping app on Lollipop depending on the zoom-in level, progressing from (a) corresponding to a low zoom-in level to (d) corresponding to the highest zoom-in level.



of Slovenia, is labeled, though Slovenia is not; at this point the app is hierarchically inconsistent. Further zooming in Figure 4d finds both Ljubljana and Zagreb disappear only to be replaced by Venice, while Slovenia is labeled for the first time. Another zoom in Figure 4e causes Zagreb to reappear. Another zoom in Figure 4f fills the display screen with Slovenia with more (and less prominent) cities but still no mention of Ljubljana. A final zoom in Figure 4g finds Ljubljana reappearing with additional cities, but now there is no mention of the containing country Slovenia.

It appears the Slovenia label is missing from Figure 4g in order to avoid overlap on account of the Slovenia label being in approximately the same position on the map as the Ljubljana label. The same example (not shown here) can be used to show that both hierarchical and zooming consistency also fail to hold for the iOS 7 and iOS 8 mapping apps, as well as for the Windows Phone mapping app. The difference, in, say, the iOS 7 and iOS 8 mapping apps is that Slovenia and Croatia are both initially visible, but subsequent zooming in finds Slovenia disappearing only to reappear on further zooming in. However, in the case of the iOS 8 mapping app, when a user zooms in further so Slovenia occupies most of the display screen, then at times both its label and the Ljubljana label are present, in which case the iOS 8 mapping app labels Slovenia with a watermark style font, as in Figure 4h. The figure shows the watermark style font is also used for Croatia, which is the adjacent neighboring country to Slovenia. Interestingly, Apple designers recognized the utility of a watermark style font label yet did not permit it to overlap a label for another object of a different type.

We also observed that, as a user zooms in further on Slovenia in the iOS 8 mapping app, the Slovenia label disappears and the font and style of the Ljubljana label change from the standard first-letter capitalized used for cities (see Figure 4h) to one that uses all caps with wide spacing similar to that used for countries (see Figure 4i). This is in contrast to the Android and Windows Phone mapping apps where the font and style do not vary with the level of zoom. We thus have another variant of zoom inconsistency that involves the

Figure 3. The result (b) of panning the map in (a) containing city and country names to the right (achieved by a swipe to the left) in the iOS 7 mapping app on an iPhone 4 yielding mostly names of cities followed by the result (c) of a further pan to the right yielding mostly names of countries, demonstrating type panning inconsistency.



font/style used for labeling objects of a given type as the user zooms in and out. Observe this is related in spirit to the object type consistency discussed earlier in the context of panning consistency. Also worth noting is the sparseness of the data associated with Slovenia that is common in the mapping apps provided by Apple—iOS 6, iOS 7, and iOS 8—compared with those available for Android (see Figure 4g), iOS 5 (not shown here), and Windows Phone (see Figure 4j).

Zoom inconsistency is also the case for the Windows Phone app for Croatia and Cincinnati (not shown here), as well as for Philadelphia (not shown here) in the iOS mapping app for Google. Observe the Windows Phone mapping app does not use watermark-style font labels, even though the Windows Phone mapping app is rooted in the iOS mapping app for Bing, which used it well. Note, too, that zoom consistency does hold for older mapping apps (such as the iOS 5 mapping app and the iOS mapping apps for Bing, Nokia, MapQuest, and OSM). It also holds partially for the iOS mapping app for ESRI, as long as users do not mind that labels change their location at the different zoom-in and zoom-out levels. This change in locations is fine, as such changes are usually prompted by a need to avoid label overlap, which is usually the highest priority. Oddly, zoom consistency is currently satisfied by the iOS 6 mapping app, though we found it failed consistently in earlier experiments. Finally, this change in locations was also less of a problem on an older version of the

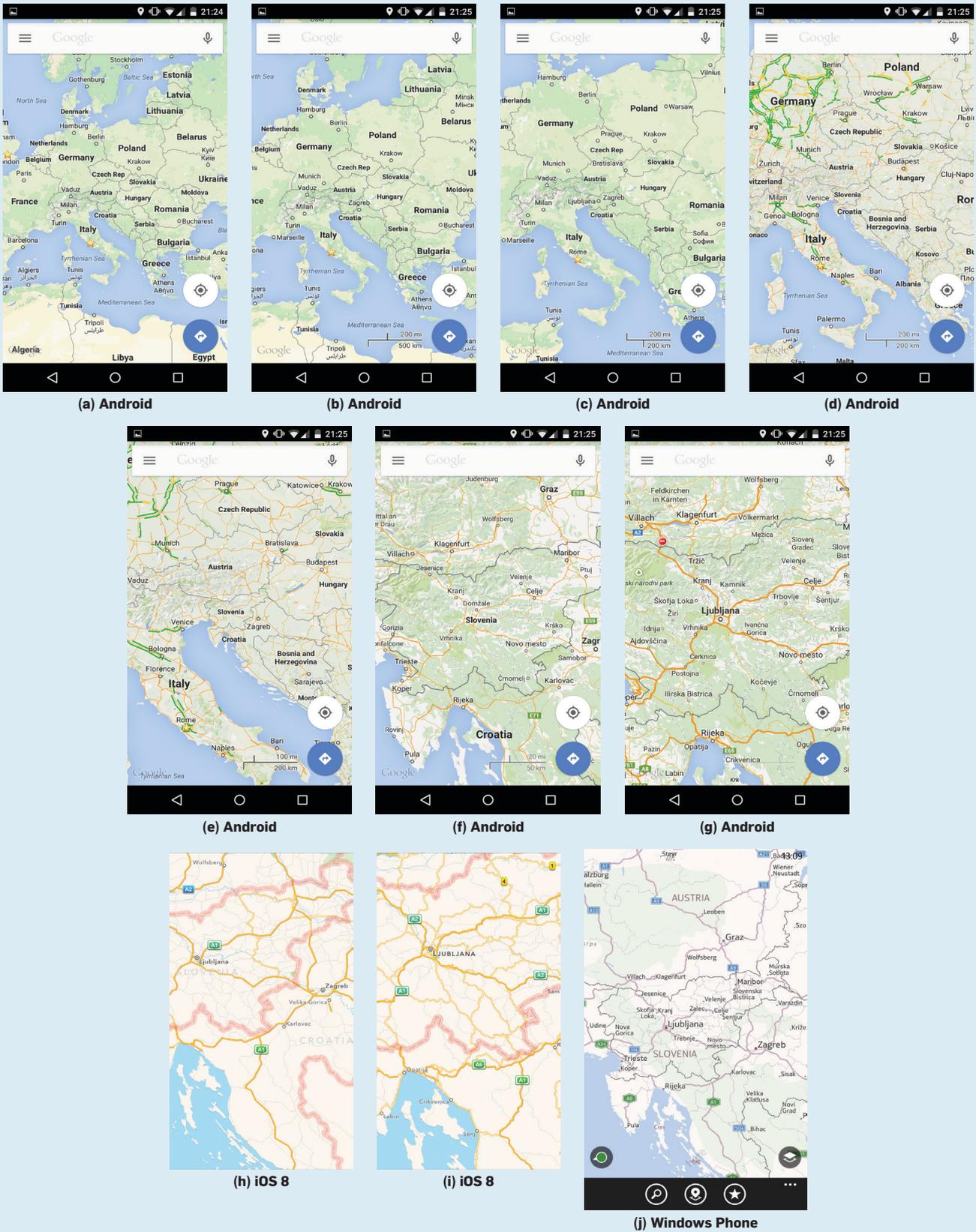
Android mapping app—Google Maps 7.4.0 on Android 4.2.1—where, once the Slovenia label appeared, it stayed put as the user continued to zoom in; the Ljubljana label never appeared until a zoom in at a very high level, at which time the Slovenia label did not appear and, likewise, neither did the Croatia label. During this process, the Zagreb label vanished at times, so this Android variant also failed to satisfy zoom consistency.

Wraparound

A user should be able to view every location both left and right or above and below one another; that is, continuous panning (wraparound) should be possible, as the Earth is a sphere. It is useful in the vicinity of, say, the Bering Strait where, without it, a user would need to do a considerable amount of panning to transition between North America and Asia.

Horizontal wraparound is possible in the iOS 7, iOS 8, Windows Phone, and Android mapping apps and APIs, as well as in the iOS mapping app and API for Google. In addition, horizontal wraparound is possible in the iOS 6 mapping app and the iOS mapping apps for MapQuest, Nokia, ESRI, and OSM. Although the iOS 5 mapping app and the iOS mapping app for Bing do not support horizontal wraparound, they do enable zooming out in landscape mode, so the entire world can be viewed in a single frame, as discussed later. This means there is at least the possibility of viewing both sides of the Bering Strait in one

Figure 4. Example showing zoom inconsistency for the Android mapping app on Lollipop while successively zooming in on Ljubljana in Slovenia and on Croatia. (a) Croatia is labeled initially while Slovenia is not. (b) Zagreb is labeled in Croatia. (c) Ljubljana is labeled while Slovenia is not. (d) Ljubljana and Zagreb vanish while Venice and Slovenia appear. (e) Zagreb reappears. (f) Slovenia fills the map. (g) Ljubljana replaces Slovenia. (h) Zoomed in on Slovenia on the iOS 8 mapping app on an iPhone 5. (i) Zoomed-in on Ljubljana on the iOS 8 mapping app on an iPhone 5. (j) Zoomed-in on Slovenia on the Windows Phone mapping app.



(a) Android

(b) Android

(c) Android

(d) Android

(e) Android

(f) Android

(g) Android

(h) iOS 8

(i) iOS 8

(j) Windows Phone

view, though they are not adjacent (see Figure 5a and Figure 6a).

Vertical wraparound is possible in only the iOS mapping apps for MapQuest and Nokia and impossible in the rest. Vertical wraparound is impossible on all of the APIs. Vertical wraparound would be useful in panning around Antarctica or the Arctic. It could also be useful in creating maps where the Southern Hemisphere is on top, in contrast to the prevalent use of the Northern Hemisphere on top. In addition, it would ease making maps like the Wizard of New Zealand’s upside-down world map centered in New Zealand.⁵

Full or Maximum Zoom Out

The ability to zoom out completely makes it possible for a user to see the entire world on the device display with one view instead of having to pan the map. Figure 5 and Figure 6 show the extent of its availability in both portrait and landscape modes, respectively, for iOS 5, iOS 8, and Android mapping apps and only for the portrait mode in the Windows Phone mapping app, as the landscape mode option is not presently available for it. At the maximum zoom-out level, the entire world can be seen in both portrait and landscape modes in the iOS mapping apps for Bing and OSM and in portrait mode for the Windows Phone mapping app.

At this level in the iOS 5 mapping app and the iOS mapping app for ESRI, the entire world can be seen in landscape mode, while 95% can be seen in portrait mode. Continuing at this level, the portrait (landscape) iOS mapping apps for Nokia and MapQuest all present a much narrower view of the world, enabling only about 25% (35%) of the world to be seen, while the portrait (landscape) iOS 6, iOS 7, iOS 8, and Android mapping apps, and the iOS mapping app for Google, being remarkably similar in coverage, are somewhere in between, enabling approximately 35% (60%) of the world to be seen. The same behavior as in the mapping app is available in the mapping APIs for iOS 5, iOS 6, iOS 7, iOS 8, Windows Phone, and the iOS mapping API for Google. However, for Android, the mapping API allows the user to see a greater part of the world than can be seen through the mapping app.

At maximum zoom-out level, most mapping apps obey hierarchical consistency, with the exception of the iOS 6, iOS 7, and iOS 8 mapping apps, which present only the names of cities (see Figure 5b and Figure 6b) and not their containing countries.

The iOS 5 and Windows Phone mapping apps display only names of continents and oceans at the maximum zoom-out level. On the other

hand, the iOS 6, iOS 7, and iOS 8 mapping apps also display the names of a few cities (but no countries), while the Android mapping app displays only the names of a few countries (but no continents or cities) at this maximum zoom-out level. The Windows Phone mapping app (see Figure 5d) is the only app able to label and display all seven continents and all five oceans at this maximum zoom-out level. This is done without requiring any panning or additional zooming as needed by the remaining mapping apps and thus is the only one that obeys sibling consistency at this maximum zoom-out level. The iOS 5 mapping app has similar properties that differ only by requiring some panning to see Antarctica and some additional zooming in to see the Arctic and Southern oceans (see Figure 5a and Figure 6a). The Android mapping app shows all oceans upon panning save for the Arctic Ocean, which requires more zooming in.

The iOS 6, iOS 7, and iOS 8 mapping apps label all continents after panning and all oceans save for the Southern Ocean, which they never label regardless of how far they zoom in. The iOS 6, iOS 7, and iOS 8 mapping apps differ primarily in their use of a boldface font for the continent names in the iOS 7 and iOS 8 mapping apps (see Figure 5b and

Figure 5. (a) World map in portrait mode demonstrating the maximum zoom-out level for (a) iOS 5 on an iPod Touch, (b) iOS 8 on an iPhone 5, (c) Android, and (d) Windows Phone mapping apps.

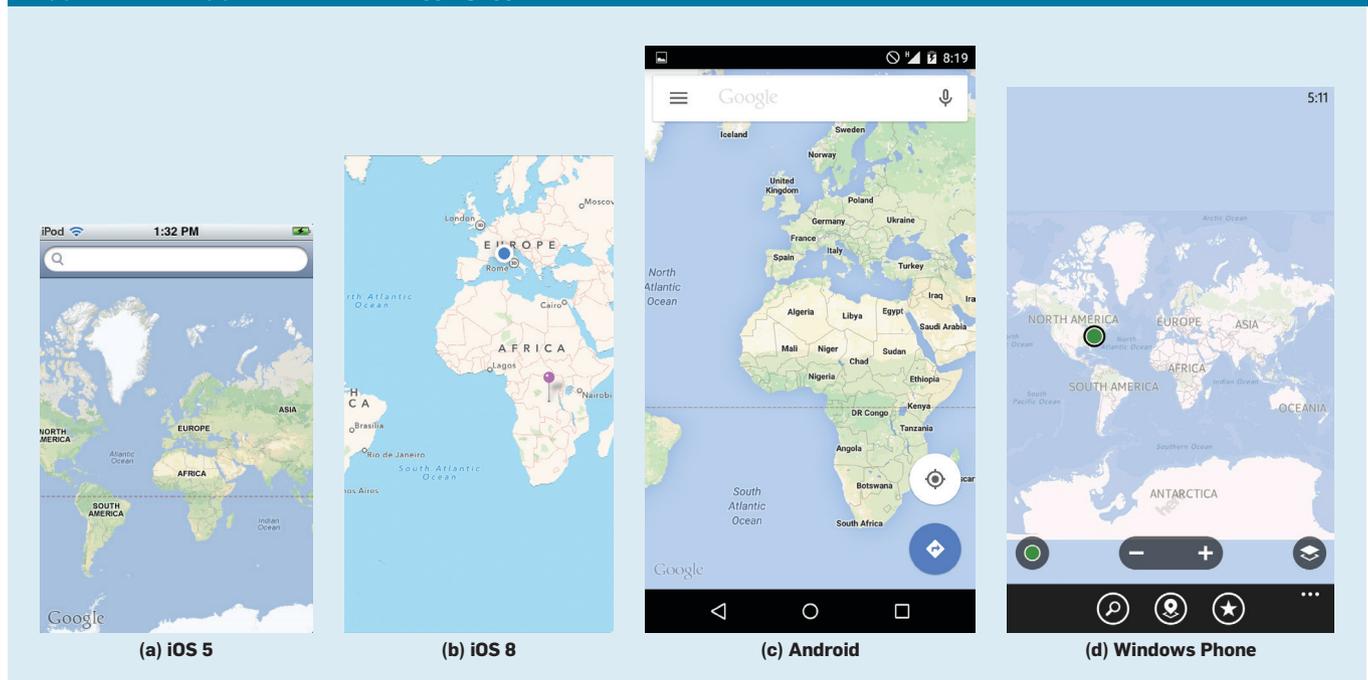


Figure 6. World map in landscape mode demonstrating maximum zoom-out level for (a) iOS 5 on an iPod Touch, (b) iOS 8 on an iPhone 5, and (c) Android mapping apps.

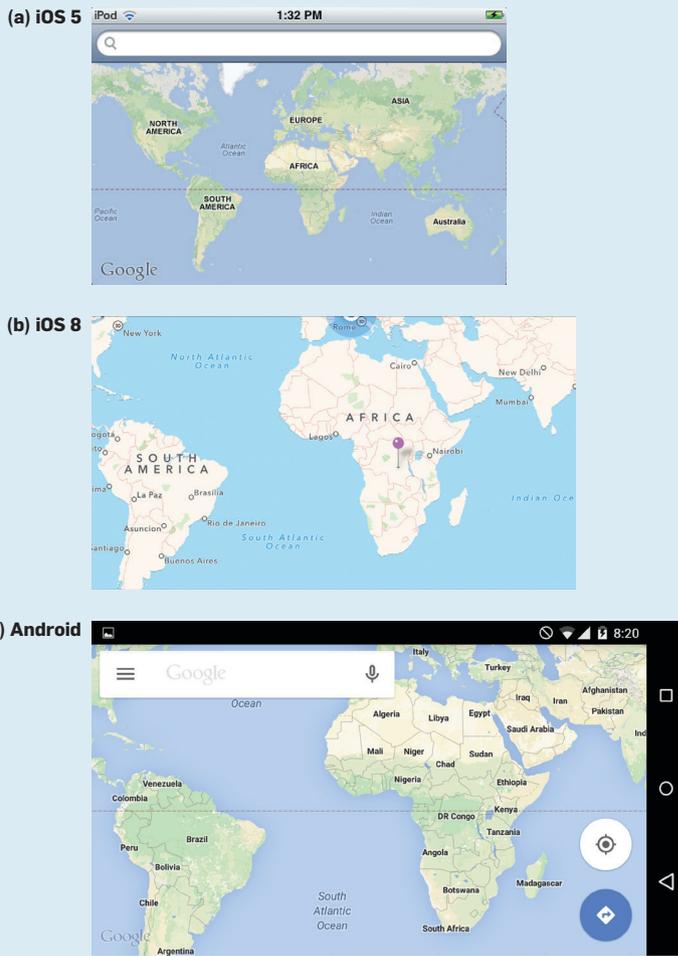


Figure 6b) and watermark style in the iOS 6 mapping apps (not shown here). As we see, this difference in the labeling of continent names occurs in both landscape and portrait modes for the iOS 6, iOS 7, and iOS 8 mapping apps.

Seeing the whole world is important if a user wants to observe a feature’s behavior over the whole world with one view rather than having to pan the map to see the full extent of the feature’s behavior. Figure 7a and Figure 7b are examples of spatially referenced news clusters from NewsStand^{13,28,34} using the iOS 6 and iOS 5 mapping apps, respectively, while Figure 7c and Figure 7d are spatially referenced mentions of diseases from the same system using the iOS 6 and iOS 5 mapping apps, respectively. A slider serves to vary the displayed clusters or disease mentions.

Conclusion

Many of the manipulation-presentation-consistency issues we have discussed here could be resolved by caching map data, as discussed by, for example, Liu et al.,¹⁵ and dynamic map labeling by, for example, Peng et al.,²² which are directions for future study. We now review some manipulation-presentation-consistency issues for the various mapping apps we found noteworthy. However, we first re-emphasize that our aim here is not to criticize Apple, Google, or Microsoft. Instead, it is to use examples motivated by Apple’s foray into the maps space where Google and Microsoft have a longer history due in part to their work on Microsoft Virtual Earth and Google Earth and Maps, to point out the difficulty of such a task, and the need to consider centuries-old principles and practices in mapmaking.

Despite the obvious similarities in Android and iOS 5 mapping apps, we identified important differences, including the way they deal with the whole world—both portrait and landscape—where only the iOS 5 mapping app provides a full view at the maximum zoom-out level.

From our limited comparison, as summarized in the table, we conclude that newer is not always better in that the iOS 5 mapping app is probably still best with respect to our four properties of presentation consistency—hierarchical, sibling, panning, and

Figure 7. Examples of spatially referenced news clusters with (a) iOS6 mapping app and (b) iOS 5 mapping app and examples of spatially referenced mentions of diseases using (c) iOS 6 mapping app and (d) iOS 5 mapping app.



zoom—making for better map-based applications on smartphone form-factor devices. The Android mapping app often performs like the iOS 5 mapping app, though it was plagued by zoom and sibling inconsistency that are also common in the iOS 6, iOS 7 iOS 8, and Windows Phone mapping apps, as well as others. The iOS mapping apps for Nokia and Bing maps were a distant third, with the main advantage for the iOS mapping app for Nokia being the ability to wrap around fully, which was sorely missing in the iOS 5 mapping app.

Note the main emphasis of this article has been to point out manipulation-presentation consistency issues in popular mapping apps. Ideally, any mapping app should satisfy all presentation consistency properties outlined earlier. In practice, however, an app developer may choose to partially satisfy or even completely ignore some of the properties due to such factors as physical space and execution time. In some cases, trade-offs must be made, and a possible future research direction is to study how to make them without compromising the user experience. Such a study would require access to different labeling algorithms and involve some form of usability testing to assess which ones deliver the best performance.

An important by-product of our study is the empirical observation that both the Google Maps API^b and the Bing Maps API^c use crow-flying distance rather than road-network distance, as discussed by Sankaranarayanan and Samet,^{30,31} in their distance-ordering queries, though both use road-network distance when providing directions. This is another example of the failure of modern mapping systems to adhere to centuries-old lessons in dealing with spatial data.

Acknowledgments

This article is based in part on an earlier article by Samet et al.²⁶ This work was supported in part by the National Science Foundation under Grants IIS-10-18475, IIS-12-19023, and IIS-13-20791. We are also grateful to Anis Abboud for his help in generating some of the screenshots. 

b <http://tinyurl.com/GoogKNN>
 c <http://tinyurl.com/BingMapsKNN>

References

- Abdelrazek, A., Hand, E., and Samet, H. Brands in NewsStand: Spatiotemporal browsing of business news. In *Proceedings of the 23rd SIGSPATIAL International Conference on Advances in Geographic Information Systems* (Seattle, WA, Nov. 4–6). ACM Press, New York, 2015. article 97.
- Biddle, S. Bloomberg: Apple fires guy responsible for crappy Apple Maps. *Gizmodo* (Nov. 27, 2012); <http://gizmodo.com/5963651/apple-firesguy-responsible-for-apple-maps>
- Dobson, M. Google Maps announces a 400 year advantage over Apple Maps. *TeleMapics* (Sept. 20, 2012); <http://blog.telemapics.com/?p=399>
- Esperança, C. and Samet, H. Experience with SAND-Tcl: A scripting tool for spatial databases. *Journal of Visual Languages and Computing* 13, 2 (Apr. 2002), 229–255.
- Flourish. *The Upsidedown Map Page*; <http://flourish.org/upsidedownmap/>
- Freifeld, C.C., Mandl, K.D., Reis, B.Y., and Brownstein, J.S. Model formulation: HealthMap: Global infectious disease monitoring through automated classification and visualization of Internet media reports. *Journal of the American Medical Informatics Association* 15, 2 (Mar.-Apr. 2008), 150–157.
- Fruin, B.C., Samet, H., and Sankaranarayanan, J. Tweetphoto: Photos from news tweets. In *Proceedings of the 20th International Conference on Advances in Geographic Information Systems* (Redondo Beach, CA, Nov. 7–9). ACM Press, New York, 2012. 582–585.
- Gabriel, C. Nokia's Bing Maps deal is a sign of the future. *Rethink TV* (May 28, 2012); <http://www.rethink-tv.com/2012/05/28/nokia-bing-maps-deal-sign-future.htm>
- Gramsky, N. and Samet, H. Seeder finder: Identifying additional needles in the Twitter haystack. In *Proceedings of the Sixth ACM SIGSPATIAL International Workshop on Location-Based Social Networks* (Orlando, FL, Nov. 5). ACM Press, New York, 2013. 44–53.
- HealthMap. *HealthMap: Global Health, Local Information* (Sept. 2012); <http://www.healthmap.org/>
- Jackoway, A., Samet, H., and Sankaranarayanan, J. Identification of live news events using Twitter. In *Proceedings of the Third ACM SIGSPATIAL International Workshop on Location-Based Social Networks* (Chicago, IL, Nov. 1). ACM Press, New York, 2011. 25–32.
- Lan, R., Lieberman, M.D., and Samet, H. The picture of health: Map-based, collaborative spatiotemporal disease tracking. In *Proceedings of the First ACM SIGSPATIAL International Workshop on Use of GIS in Public Health* (Redondo Beach, CA, Nov. 6). ACM Press, New York, 2012. 27–35.
- Lieberman, M.D. and Samet, H. Supporting rapid processing and interactive map-based exploration of streaming news. In *Proceedings of the 20th International Conference on Advances in Geographic Information Systems* (Redondo Beach, CA, Nov. 7–9). ACM Press, New York, 2012. 179–188.
- Lieberman, M.D., Samet, H., Sankaranarayanan, J., and Sperling, J. STEWARD: Architecture of a spatio-textual search engine. In *Proceedings of the 15th annual ACM International Symposium on Advances in Geographic Information Systems* (Seattle, WA, Nov. 7–9). ACM Press, New York, 2007. 186–193.
- Liu, C., Fruin, B.C., and Samet, H. SAC: Semantic adaptive caching for spatial mobile applications. In *Proceedings of the 21st ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems* (Orlando, FL, Nov. 6–8). ACM Press, New York, 2013. 184–193.
- MapQuest; <http://platform.mapquest.com/>
- Nutanong, S., Adelfio, M.D., and Samet, H. Multi-resolution select distinct queries on large geographic point sets. In *Proceedings of the 20th International Conference on Advances in Geographic Information Systems* (Redondo Beach, CA, Nov. 7–9). ACM Press, New York, 2012. 159–168.
- Nutanong, S., Adelfio, M.D., and Samet, H. An efficient layout method for a large collection of geographic data entries. In *Proceedings of the 16th International Conference on Extending Database Technology* (Genoa, Italy, Mar. 18–22). ACM Press, New York, 2013. 717–720.
- OpenSeaMap; <http://www.openseamap.org/>
- OpenStreetMap; <http://www.openstreetmap.org/>
- Paolino, L., Romano, M., Tortora, G., and Vitiello, G. Spatial data visualization on mobile interface: A usability study. In *Proceedings of the Ninth International Wireless Communications & Mobile Computing Conference* (Cagliari, Sardinia, Italy, July 1–5). IEEE Press, New York, 2013. 959–963.
- Peng, S.F., Adelfio, M.D., and Samet, H. Viewing streaming spatially referenced data at interactive rates. In *Proceedings of the 22nd ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems* (Dallas, TX, Nov. 5–7). ACM Press, New York, 2014. 409–412.
- Samet, H., Adelfio, M.D., Fruin, B.C., Lieberman, M.D., and Sankaranarayanan, J. PhotoStand: A map query interface for a database of news photos. In *Proceedings of the 39th Conference on Very Large Data Bases* (Riva del Garda, Trento, Italy, Aug. 26–30, 2013), 1350–1353.
- Samet, H., Adelfio, M.D., Fruin, B.C., Lieberman, M.D., and Teitler, B.E. Porting a Web-based mapping application to a smartphone app. In *Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems* (Chicago, IL, Nov. 2–4). ACM Press, New York, 2011. 525–528.
- Samet, H., Alborzi, H., Brabec, F., Esperança, C., Hjaltason, G.R., Morgan, F., and Tanin, E. Use of the SAND spatial browser for digital government applications. *Commun. ACM* 46, 1 (Jan. 2003), 61–64.
- Samet, H., Fruin, B.C., and Nutanong, S. DUKing it out at the smartphone mobile app mapping API corral: Apple, Google, and the competition. In *Proceedings of the Fourth ACM SIGSPATIAL International Workshop on Mobile Geographic Information Systems* (Seattle, WA, Nov. 3). ACM Press, New York, 2012. 41–48.
- Samet, H., Nutanong, S., and Fruin, B.C. Static presentation consistency issues in smartphone mapping apps. *Commun. ACM* 59, 5 (May 2016), 88–98.
- Samet, H., Sankaranarayanan, J., Lieberman, M.D., Adelfio, M.D., Fruin, B.C., Lotkowski, J.M., Panozzo, D., Sperling, J., and Teitler, B.E. Reading news with maps: The power of searching with spatial synonyms. *Commun. ACM* 57, 10 (Oct. 2014), 64–77.
- Samet, H., Teitler, B.E., Adelfio, M.D., and Lieberman, M.D. Adapting a map query interface for a gesturing touchscreen interface. In *Proceedings of the 20th International World Wide Web Conference* (Companion Volume) (Hyderabad, India, Mar. 28–Apr. 1). ACM Press, New York, 2011. 257–260.
- Sankaranarayanan, J. and Samet, H. Query processing using distance oracles for spatial networks. *IEEE Transactions on Knowledge and Data Engineering* 22, 8 (Aug. 2010), 1158–1175.
- Sankaranarayanan, J. and Samet, H. Distance oracles for spatial networks. In *Proceedings of the 25th IEEE International Conference on Data Engineering* (Shanghai, China, Mar. 29–Apr. 2). IEEE Press, New York, 2009. 652–663.
- Sankaranarayanan, J., Samet, H., Teitler, B.E., Lieberman, M.D., and Sperling, J. Twitterstand: News in tweets. In *Proceedings of the 17th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems* (Seattle, WA, Nov. 4–6). ACM Press, New York, 2009. 42–51.
- Sarma, A.D., Lee, H., Gonzalez, H., Madhavan, J., and Halevy, A.Y. Efficient spatial sampling of large geographical tables. In *Proceedings of the ACM International Conference on Management of Data* (Scottsdale, AZ, May 20–24). ACM Press, New York, 2012. 193–204.
- Teitler, B., Lieberman, M.D., Panozzo, D., Sankaranarayanan, J., Samet, H., and Sperling, J. NewsStand: A new view on news. In *Proceedings of the 16th annual ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems* (Irvine, CA, Nov. 5–7). ACM Press, New York, 2008. 144–153.
- Tumblr. The amazing iOS 6 maps; <http://theamazingios6maps.tumblr.com/>
- Whitney, L. Apple's Maps app slammed over missing cities and other mistakes. *CNET* (Sept. 20, 2012); <http://www.cnet.com/news/apples-maps-app-slammed-over-missing-cities-and-other-mistakes/>

Hanan Samet (hjs@cs.umd.edu) is a distinguished university professor in the Computer Science Department, Center for Automation Research, and Institute for Advanced Computer Studies at the University of Maryland, College Park, MD.

Sarana Nutanong (nutanong@umiacs.umd.edu) is an assistant professor in the Computer Science Department of the City University of Hong Kong, Hong Kong SAR, China; his research for this article was conducted while he was a research associate at the University of Maryland, College Park, MD.

Brendan C. Fruin (bcfruin@gmail.com) is a software engineer at Zillow, Seattle, WA; his research for this article was conducted while he was a student at the University of Maryland, College Park, MD.