

Document downloaded from:

<http://hdl.handle.net/10251/36919>

This paper must be cited as:

Palomar-Vázquez, J.; Pardo Pascual, JE.; Sebastián Tarín, L.; Recio Recio, JA. (2012). A Technical Solution to Allow Off-line Mobile Map Querying of Discrete and Continuous Geographic Attribute Data. *Cartographic Journal*. 49(2):143-152.
doi:10.1179/1743277411Y.0000000029.



The final publication is available at

<http://dx.doi.org/10.1179/1743277411Y.0000000029>

Copyright Maney Publishing

A technical solution to allow off-line mobile map querying of discrete and continuous geographic attribute data

Jesus Palomar-Vazquez^{a*}, Josep E. Pardo-Pascual^a, Laura Sebastián^b,
Jorge A. Recio^a

^a *Department of Cartographic Engineering, Geodesy and Photogrammetry*

^b *Department of Computer Systems and Computation*

Universitat Politècnica de València

Camí de Vera s/n, 46022 València, Spain

Corresponding author. Email: jpalomav@upvnet.upv.es

Phone: 34963877007 (Ext:75563); Fax: 34963877559

In this article a technique towards the generation of hybrid raster-attributes map for use in mobile devices is described. Our solution is based on coding the map attributes within an image using RGB values. The designed coding method enables the simultaneous storage of discrete thematic attributes and continuous quantitative attributes. This approach offers a wide range of possible uses. Small memory storage requirements and the simplicity of the software enable this coding method to be used efficiently in mobile devices without internet connection.

This article describes the basic fundamentals of the coding technique, as well as the operation and limitations regarding the volume of information. Two specific applications are presented: a topographic map used for recreational activities, and a visitor map of a university campus.

Keywords: raster coding, visualisation, GIS, mobile devices

Acknowledgements.

The authors appreciate the financial support provided by the Spanish *Ministerio de Ciencia e Innovación* in the framework of the Projects CGL2009-14220-C02-01 and CGL2010-19591 and by the *Universidad Politécnica de Valencia* in the framework of the *Programa de Apoyo a la Investigación y el Desarrollo* (first research projects PAID-06-08).

1. Introduction

The availability of geographical information has multiplied over the last several decades due, in large part, to the use of computing resources (computers, mobile devices, etc.) and the ease with which the internet enables digital information to be transmitted. These computing developments have greatly increased the number of users of cartography and, therefore, geographical information has gone from being used mostly by specialists to become general information used by untrained individuals. As a result of this new reality, the generalised access and use of selected geographic data can become an important incentive to activate and drive the market for determined services and, therefore, the economic development of a given area.

For this information to be used by most people, it must be easy to acquire and manage, as well as economical to use. A mobile phone can meet these needs by acquiring map information from the internet – especially with the major improvements developed in recent years.

Mapping on mobile devices presents distinct advantages and disadvantages in comparison to computers and laptops. The main disadvantages are related to the display limitations (small and non-standardised screens), as well as the storage and management of information. The advantages are related with the portability that enables users to use the mapping features in the location indicated on the map, especially if combined with a positioning system. Nivala et al. (2007) studied the needs and benefits identified by real users (hikers following topographic maps on mobile phones) and concluded that the main benefit apart from GPS information (which has been identified as the main advantage by several authors including Nivala and Sarjakoski, 2003; Letho and Kilpeläinen, 2000, 2001) is the ability to combine and display information from various databases on the map. In this way, content is partly individualised to match the needs and desires of users. This is a key advantage as it enables maps to be made specific to the context of each user.

The use of mobile mapping requires making several adaptations to achieve real applicability. Various adaptation approaches have been proposed to achieve adequate mapping capabilities on mobile devices (Reichenbacher, 2004; Nagi, 2005):

- (i) Adapting geo-information – including encoding format, amount and detail of content, classification of information, and size of geographic area.
- (ii) Adapting the user interface to facilitate access to information with features that simplify interaction: such as zooming, panning, rotating, selection, hotlinks (Brown et al., 2001, Frank et al., 2004).
- (iii) Adapting the content of map displays (Zipf, 2002): including map orientation, scaling, colour scheme, user position on the map, focus on the area of interest.

We focus on resolving some of the limitations in adapting the geo-information that is generally used in mobile devices. The volume of information that can be stored on a mobile device is always limited – as is the ability to interact with information in search processes or location attributes. This has meant that users usually access information in geo-referenced image formats and the user position is established through GPS (if available).

Users can ‘navigate’ through a territory and recognise the equivalence between the map and reality, but users cannot request information from the map. Because the map is a raster image there are no explicit geographic attributes (names, areas, distances, street names, altitudes, etc.). Groups of pixels with a specific colour help the user recognise some geographic information that may be implicit and this is achieved with the help of map symbols or the availability of an adjacent legend – features that are always difficult to add given the small size of the screen. The fact that the information is not available explicitly means that the most useful mapping solutions must be predefined for the potential user. For example, the design of a hiking map should predefine significant height data, as well as the location and name of springs, etc. However, given the limitation of the display, only part of this information would be explicit. The availability of all the information stored in a database enables the information to be used in a way that matches user preferences and needs. To achieve this, a database must be accessible together with a software tool to translate the information. There is an additional complication caused by the fact that the bulk of geographical information is stored in two different format types (vector and raster) and this complicates the task because of the limitations in memory, software, and computing power on most phones.

If there is a good internet connection these tasks can be resolved on a server and the mobile device is simply left to request information and the tasks are resolved remotely. In this case, the wireless device simply receives and displays information. However, mobile coverage is neither very broad nor universal. There are wide areas in which the transmission signal is limited or non-existent – especially in remote mountainous areas far from cities. This means certain types of activities that require maps cannot be developed using mobile phones. Examples of these activities include hiking or mountain sports, forestry protection, and mountain rescue.

The purpose of this article is to offer a solution to make available geographic information (vector and raster) so that all the communication capability of the thematic information can be used and managed from general mobile devices (PDAs, mobile phones) without the need to be connected to the internet. In this way, the user can use the map and consult attributes from a mobile device without an internet connection.

In the next section of this paper, the proposed solution based on a hybrid raster-attribute format is described, showing the suitability of the PNG format to code information. The third section explains the three levels used to organise the information. Following, the methodology to rasterize the features of a vector map and to code continuous and discrete attributes is presented. In the fifth section, technical details about the real application of this methodology with two different types of maps, a topographic map and a university campus map, are given. Finally the strengths and limitations of this methodology are analyzed.

2. Description of the proposed solution

The proposed solution is based on the idea that the information is handled inside the mobile device as images – given that images are the most standard format and can be handled by all mobile devices. Therefore, the first task to be resolved is how to link the attributes to be transmitted to the raster image. To achieve this, the image must be converted from a raster format into a ‘hybrid raster-attribute format’. The idea is not new, and there are already raster formats with attributes on the market, the most well-

known being the ESRI¹ GRID format. Although the general philosophy of this format would fulfil our requirements, it presents several disadvantages:

- The format is proprietary. Map servers use standard image formats that are open (JPEG, GIF, PNG, etc.) and easy to visualise using any application. The use of proprietary formats requires that the mobile devices have the specific software in order to read them – and this need greatly restricts their use.
- The format is complex and composed of various files (at least seven) and requires a very large application in terms of memory for management – an important limitation if it is to be used with mobile devices. In fact, not even ArcPad² (a mobile GIS application from ESRI) uses this format.
- Discrete and continuous variables can be stored, but not in a single file (for multivariate analysis, GRID files can be combined into a structure called a stack, but they continue to be separate files).

One of the most widely used formats for the exchange of raster information is TIFF (and its version for geo-referenced data, GeoTIFF (Mahammad and Ramakrishnan, 2003)). This format is capable of storing continuous attributes and although open has the disadvantage of being complex, which means it cannot be used on most mobile devices.

Therefore, due to the specific characteristics of mobile devices (low storage and processing capacities, etc.) we need formats with widespread usage (standard) and low complexity for the visualisation and processing of images. Although there are many graphic standards on the market (BMP, GIF, PNG, etc.) only some of these are suitable for mobile devices.

Most mobile devices currently accept the programming of applications in J2ME (Java 2 Micro Edition). J2ME is a specification of a subgroup of the Java platform for the development of software in devices with limited resources such as PDAs, smartphones, mobile phones, etc. (White, 2001). In the graphics section, J2ME recommended the PNG³ format in 1999 due to the following characteristics (Roelofs, 1997; Hunter and Zhan, 1999):

- Open format (not subject to patents).
- Small size.
- Accepted by many graphic editors.
- Allows transparencies.
- Format accepted by most mobile terminals.

Another important candidate is the GIF format which, while sharing many of the same advantages as the PNG format, has certain disadvantages for use in mobile devices, such as its dependence on patents (it was not freed until 2004) and a limited colour palette⁴.

Our goal is, therefore, to select a graphic format that is standard, simple, suitable for mobile devices, has associated graphic attributes, and permits visualisation and

¹ http://webhelp.esri.com/arcGISdesktop/9.2/index.cfm?TopicName=About_the_ESRI_grid_format

² <http://www.esri.com/software/arcGIS/arcpad/index.html>

³ <http://www.libpng.org/pub/png/>

⁴ <http://www.w3.org/Graphics/GIF/spec-gif89a.txt>

querying without internet access. Based on these premises, the PNG format was selected as being adequate for this research.

Another fundamental point to be resolved is how to assign geographic attributes to a graphic format. Previous works (Andrienko and Andrienko, 1999; Dang *et al.*, 2001) served as a foundation for Zhao and Shneiderman (2005) to design a methodology based on the coding of attributes within an image and associated with the pixel colour. The work of Zhao and Shneiderman is oriented towards the distribution, visualisation, consultation, and manipulation of choropleth maps using web applications. These maps offer a solution in which the geographic attributes are coded through the colour of each pixel within a GIF image. This code is used as a link to a file of standard attributes (CSV, DBF, plain text, etc.). The coded GIF file and the attribute file are downloaded from the server and stored in the client, where queries and thematic maps can be made without having to make a request to the server (the GIF file is managed in local mode using Javascript) and this approach makes response time very short.

This solution offers some basic principles that can help us fulfil the goal we have proposed. Nonetheless, for our purposes, the use of the GIF format is inadequate for several reasons. In the first place, as previously commented, this format was subject to patent until 2004, meaning it was not contemplated within J2ME. Additionally, we want to increase the capacity of the method to include any type of map and not only choropleth maps. For this reason, the maximum number of different elements that the GIF format can code (256 in 8-bits) is not enough to represent the wide variety of phenomena and variables we wish to include (for instance, features such as crops, rivers, roads, and points of interest in a topographic map). Finally, the Zhao and Shneiderman coding method contemplates the coding of discrete, but not continuous variables.

The method proposed here is also based on the coding of attributes using pixel colour, and increasing the representation capabilities using the PNG format to achieve the following objectives:

- Suitability for J2ME applications with mobile devices.
- Ability to code any type of map. RGB coding using 24-bits from the PNG format considerably increases the number of geographic features represented (about 16 million).
- The capacity to code both discrete and continuous variables.

By using the compression method of the PNG format a considerable reduction in the size of the image can be made without lessening quality and with a colour depth of 8-bits per channel. This is ideal for our work, as we need a graphic format that enables the coding of a large number of features without altering the colours. This last characteristic is very important for our interests. As mentioned earlier, the coding process basically consists of identifying and converting the value of an attribute into a specific RGB code, that is, a colour. Other types of formats also compress images, but with a loss of quality (such as the case of JPEG). During the image compression process the original colours are changed into colours that are visually similar, but numerically different. This prevents the decoding process subsequently making these types of formats inadequate for this work (Wiggins *et al.*, 2001).

3. Architecture

The system is organised on three information levels (Figure 1):

- Level 1. The visualisation level. This level is represented by the image that the user will see in the final application (a map, an orthophoto, etc.). It is a reference image.
- Level 2. The coding level. This is the coded image. It contains the values of the identifiers of each geographic element and is the image that is consulted by the user.
- Level 3. The attribute level. This level contains the attributes of the image. This level can have different formats and store the identifiers that serve as links to the image in Level 2.

Figure 1. Level 1 (a): reference map – what the user will see in the device; Level 2 (b): coded map; Level 3 (c): attributes.

According to this plan, a user that accesses a map server with the capacity to offer this type of data would download the three files corresponding to the three levels shown above. All of the attribute information is in the second and third levels, meaning that any type of consultation can be made off-line.

This proposed methodology does not have to be the only solution. For example, instead of storing several continuous layers in the same image, we could use one image for each continuous variable. In this way, users could choose what type of layer they wish to download.

4. Methodology

Given the solution proposed here, it is necessary to transfer the attributes of the features from a vector map to a raster map. This process entails two different phases: (i) the rasterizing of all the vector layers and their integration into one PNG file; and (ii) coding the attributes as part of the colour information of each pixel (RGB coding).

Each of these phases will be described thoroughly in the following sections.

4.1 Rasterizing the vector layers

Any digital vector map is formed by a group of layers. In each of these, different types of features (roads, rivers, buildings, etc.) with different geometric types (polygons, lines, and points) are represented. The rasterizing of a vector feature implies the subdivision of the space into cells (pixels) and the association of the pixels with an attribute related to that feature (Congalton, 1997). In our case, the attribute used will be the identifier (ID) of each feature in each layer of the map. Thus, the first step consists of uniquely identifying each vector feature of the map. The number of final features (NF) will be:

$$NF = L_1 \cdot Fn_1 + L_2 \cdot Fn_2 + \dots + L_n \cdot Fn_n \quad (1)$$

being L_i each of the layers and Fn_i a finite number of features within each layer.

Once the ID of each feature is obtained, the second step consists of obtaining one raster layer from all of the initial vector layers. For this, each of the vector layers is rasterized separately, so obtaining their corresponding raster layers (with the same extent and pixel size). The process ends when all of the raster layers are combined into one final layer (Figure 2). This process mosaics all the separate raster layers into a new single raster layer. To solve overlapping problems, we use the ‘minimum’ rule: the output cell value of the overlapping areas will be the minimum value of the overlapped cells (e.g. although pixel [2,5] shares IDs 2, 6 and 9, only ID 2 will survive). This rule enables an adequate rasterization order, which coincides with the final visualisation order.

Figure 2. Rasterization process: a) initial layers; b) unique identification of each feature; c) rasterization of each layer; d) combination into one raster layer.

The following recommendations must be considered for the process to be effective.

- The integration of the individual raster layers into one raster layer should be done in a specific order. In this way, raster layers with low IDs values will be promoted to the first places. Normally, this order depends on the visualisation hierarchy, but it can be changed depending on needs.
- In general, all of the linear and point features should be converted into polygons to improve the efficacy of the query process. This process will be carried out by buffer operations (Figure 3). In this case, the size of the buffer zones has been calculated according to the symbol size (point features) or symbol width (linear features).

Figure 3. Improving user sensitivity with buffer zones: (a) base map. (b) vector features. (c) buffer zones according symbol size. (d) combined raster.

At the end of this process, a raster image is produced where each pixel is associated with the ID of the vector feature that lies underneath.

4.2 RGB coding of the feature IDs

In our work, the use of colour is fundamental for coding geographic information through the transformation of attributes into RGB code.

This method is based on converting the value of the ID (an integer number) into a binary number. The 24 bits that the PNG format allows are used to represent this number, and are arranged from right to left. Each of the 8 bits corresponding to the RGB components is then taken and converted into an integer number (between 0 and 255) to obtain the final coded colour (Figure 4).

Figure 4. Coding process of an ID within the RGB colour code. An ID value of 1000 corresponds to (0,3,232) RGB value.

The opposite process of decoding consists of converting each of the RGB components to the binary system and combining them to form one 24-bit binary

number. The final ID is obtained by converting the 24-bit binary number to the decimal system.

4.3 RGB joint coding of the feature IDs and continuous attributes

The IDs of the vector features represent discrete information. It is important to be able to apply this type of coding to continuous attributes as this enables a wider range of applications. Continuous attributes are considered to be those for which information exists in all areas of the territory, such as values of potential solar radiation, distances from a defined point, slopes, altitudes, etc. All of this data will be in raster format. To carry out the tests, a digital elevation model (DEM) and the IDs of the vector features were coded.

Both types of codes should be integrated into the RGB value of each pixel of the final image. For each attribute a certain number of bits are allocated. If we call NBF the number of bits allocated for the IDs of the features, and NBRV the number of bits allocated for the raster value, then $NBRV = 24 - NBF$. In this way, the maximum ID value that can be stored is $2^{NBF} - 1$. Additionally, the maximum value of the continuous attribute that can be stored is $2^{NBRV} - 1$.

Therefore, it is important to decide the NBF and NBRV values. The first is directly conditioned by the number of vector features to be stored. The larger the NBF, the lower the NBRV, and as a result the continuous attribute information will be more degraded. Given that the available storage capacity for the continuous attributes depends on the number of vector features in the map, it is necessary to impose some limitations on the manner of storing a continuous variable so as to ensure the storage of all the represented information. These limitations will establish both the maximum amount of data to be introduced and the degree of precision that can be achieved.

If we know the NBF and NBRV, all that remains is to apply a methodology similar to that explained in Section 4.2.

- The ID and continuous attribute values are converted to the binary system.
- The resulting binary numbers are extended (with zeros to the left) until reaching the same number of digits as the NBF and NBRV.
- Both binary numbers are combined by placing the binary code of the identifier of a feature on the left; and the binary code of the raster value or continuous attribute on the right – so forming a total of 24 bits. Figure 5 shows an example where the identifier 1926 appears coded and there is an elevation value of 359. The final result in RGB code is (120,97,103), resulting in a grey tone.

Figure 5. Joint coding process of an ID and the elevation value for a specific pixel.

If the range of the continuous attribute is greater than $2^{NBRV} - 1$, attribute data should be scaled to the interval $[0..2^{NBRV} - 1]$ in order to be stored in the available bits.

Figure 6 shows the applied coding and decoding rules for the present example, using a DEM, a rounding rule has been applied to the elevation values and these have been transformed into integer numbers.

Figure 6. Rules for the coding and decoding of elevation values.

The decision regarding the level of precision that should appear in the information on the continuous attribute should be based on: the nature of the information to be introduced, the potential applications that can be derived, and the available capacity for storing the information in a 24-bit binary code.

5. *Tests and results*

Two different tests were carried out for the generation of two off-line maps that differ in their nature and application. The first is a topographic map for recreational use. The second is a large-scale visitor map for informative use.

The software and hardware used to carry out the tests were the following:

- For vector maps processing, production of visualisation levels, and coding of the attributes: ArcGIS 9.2, VBA (Visual Basic for Applications) and FreeImage⁵ (open source library for image processing).
- For treatment of the information in mobile devices, specifically in mobile phones: J2ME, NetBeans⁶ (open source platform for Java development) and Sun Java Wireless Toolkit (development of mobile devices applications with J2ME).
- Mobile device: Nokia N95 8GB.

5.1. *Topographic map*

Map sheet 7427 of the Cartographic Institute of Catalonia (the official mapping agency of the regional government of Catalonia, Spain) was used. The characteristics of this map are the following:

- Scale 1:25 000
- Multiple vector features with very diverse typology, shape, and extension (crops, communications, hydrology, altimetry, buildings, toponymy, etc.).
- Number of layers: 14.
- Number of vector features: 3805
- Maximum elevation: 1705 m.

The 14 original layers were used for the visualisation level test, although at the coding level only 6 of these were used. The remaining 8 were not considered because either their information was integrated into the DEM data – contour lines and height points – or because at the level of testing used the introduction of the layer was irrelevant (such as the case of the sheet frame and administrative borders). As the number of vector features to be coded (3805) is greater than 2048 and less than 4096, the NBF as well as the NBRV should be 12.

Figure 7 shows: (a) the image of the level of visualisation; (b) the results of the coding of the vector features; (c) and the DEM. The result of the final joint coding where both groups of information can be seen is also shown (d).

⁵ <http://freeimage.sourceforge.net>

⁶ <http://www.netbeans.org/>

Figure 7. (a) original map; (b) coding of the IDs; (c) coding of the elevation values; (d) joint coding.

The structure of the attribute file used in this case is simple, with only three fields: row ID, type, and name. The first of these fields serves as a link to the coded image and is essential. The second field enables the grouping of pixels into layers and permits the application of masks to visualise different levels of information. This can be useful when the user only wants to visualise certain types of information in order to simplify the interpretation. The last field identifies the name of the feature and enables searching operations (Figure 8).

The inclusion of the elevation value enables the calculation of derived information that could be useful (contour, aspect, topographic profiles, etc.).

Figure 8. Database file sample for topographic map test.

Finally, to avoid multi-resolution problems, the map dataset has been structured in three levels of representation (scales 1:200 000, 1:50 000 and 1:25 000). Each map has been rasterized while taking into account the initial scale, the screen size of the mobile device (240 x 320 pixels), and the spatial resolution (4.3 meters per pixel at 1:25 000 scale).

5.2. Visitor map

For the visitor map, the information map of the Universidad Politécnica de Valencia (Spain) campus has been used as an example. The characteristics of this map are the following:

- Scale 1:200.
- Small number of features, mostly buildings, roads, and icons.
- Number of layers: 6.
- Number of vector features: 108.
- Maximum elevation: the elevation was not taken into consideration (the dimensions of the campus are small and the terrain is flat, additionally, the application is centred on the identification of campus services).

All layers are used to obtain the image of the visualisation level, although only one (buildings) is coded. Given that the elevation value is not taken into consideration, the NBF is 24 (Figure 9).

Figure 9. Original visitor map (a); coded visitor map (b).

The organisation of the information in the attribute file differs in this case. In the case of the topographic map, each field of the table has one associated value. As a result, the structure of the attribute file is quite simple. However, for the case of the visitor map, each building has different tenants (departments, research institutes, schools, etc.). For this reason, a more complicated structure was selected that enables storing the information in a single file. A product such as this can offer the user

additional information (for example, field photographs of the area, contact telephone numbers, or timetables). The enlargement of the database gives the product greater applicability, but requires more memory.

The structure of the attribute file for the visitor map of the university campus is as follows (Figure 10):

- Row ID, building ID, description of each tenant inside the building

Figure 10. Database file sample for campus test.

Depending on the structure of the attribute file, the application must be adapted to read this information, while the part of the application corresponding to the image coding remains the same.

5.3. Application

Three files were obtained for each of the tests analysed above.

- A visualisation image in PNG format. This is the image the user will see in the application.
- A coded image with the values of the IDs and the coded elevations (where applicable) through the RGB colour space. This will be the image that is consulted by the user.
- An attribute file (plain text format in this case) with information for each point of the image.

For the visualisation and query of this group of files, a MIDlet has been developed. A MIDlet is a Java application that follows the MIDP (mobile information device profile) specification and can be installed in any mobile device that accepts J2ME (Peursem, 2002).

In the case of the topographic map, the purpose of the application is to simply show the performance of the hybrid format, for which only a series of basic functions have been implemented. The tools include navigation over the image (zoom and pan), and the consultation of attributes through the positioning of the cursor.

In Figure 11, the performance of the application installed in a Nokia N95 is shown.

Figure 11. Application prototype for the visualisation and query of the raster-attribute hybrid method in the Nokia N95 mobile phone. Left image shows the topographic map application test; while right image shows the visitor map application test.

In the second example, the visitor map of the university campus⁷ has been developed with more sophisticated tools such as pathfinding, database queries, GPS and QR-code location, etc. Multiple tests have been carried out with different types of mobile phones. The definitive solution was the development of a tool that includes a database with all of the previously indicated information – as well as small

⁷ <http://polimapa.blogs.upv.es/>

photographs of the façades of the registered places. This application occupies 642 kB and its correct performance has been confirmed with various models of Nokia mobile phones using their on-line emulation service (remote device access service).⁸ The following models were tested: N95, 5230, 6710, 6220 Classic, n93, 6110, E55, 5320, X6, 5700, N97 and E72.

Finally, although both prototypes have been designed to work in off-line mode, we have also implemented an on-line version for the topographic maps. To this end we have created a server from which the user can download three levels of information (reference map, coded map, and database). Because the map tiles have been pre-computed according to the mobile device specifications, the run-time rasterization process can be avoided. In addition, the user can make any number of queries without connecting to the server using the same reference map.

5.4. Discussion of the results

The procedure for creating this mapping product is quite simple following the described methodology. There are, however, three types of important limitations to take into consideration. The first has been pointed out by multiple authors (Reichenbacher, 2004; Gartner and Uhrlirz, 2001; Sakari *et al.*, 2004), and emphasises the need to consider how much information can be managed by this method, as well as what a reasonable resolution is – given that the use of mobile devices imposes memory limitations.

Most low and mid-range mobile phones have little RAM memory. This means that the image loaded into the memory when the Java application is executed should be in accordance with these restrictions. It is very important to set a strategy to limit memory requirements when composing the whole product. For this reason, reduction techniques must be applied to the original information (image resolution for visualisation, size of the area on the screen, diversity, and complexity of the adjoining database). In this regard, the rasterization process can be adapted to each mobile device in order to improve the final visualisation: we can create map tiles according to the ‘physical resolution’ (width and height of screen phone in pixels) and ‘spatial resolution’ (meters per pixel according to the base map scale). In this sense is obvious that the higher is the raster resolution, the higher is the needed memory to store the information.

Additionally, it is important to follow strategies in the specific architecture of the software oriented towards minimising the use of memory, such as optimising the application source code as much as possible.

The second limitation to be taken into consideration is that there is not one specific language that can be used in all types of mobile phones or other devices. This makes necessary the creation of specific tools for each family of mobile phones.

A third type of limitation comes from the specific nature of the product to be developed. Since different settings should be specified according to the needs of the final user (NBF and NBRV values, as well as the structure of the database, can change) a reprogramming of the solution is required, at least in part, for each type of product to be developed.

⁸ http://www.forum.nokia.com/Devices/Remote_device_access

All of these limitations – limited memory, the need for specific tools for each type of mobile phone family, the specification of the product – make it important to consider this type of mapping solution in order to offer concrete and specific products to meet the needs of the user.

As it has been shown in the last section, the proposed technique can be applied to any type of vector maps (discrete attributes). The only requirement is to choose the suitable order and number of layers according to our needs (tourist map, topographic, geomorphologic, etc.). In the case of raster maps (continuous attributes) some limitations exist because the attribute value is stored as part of the pixel value. So if only one continuous attribute is stored (e.g. terrain elevation) it is possible to represent its value into 24-bits per pixel, but when it is necessary to store more than one type of raster attribute other kind of solution can be applied (e.g. adding other level of information or reducing the scale of the data values).

6. Conclusions

A specific solution to store and query geographic information from standard images of both discrete and continuous types has been described. These images can be stored and easily managed by some mobile devices and can be used without an internet connection. This offers a substantial advantage with respect to the more common solutions that are available in the market and where users can have access the image but not to the attributes that explain what appears on the screen; or the user is required to have complex hardware and software that makes mobility difficult. This solution enables to query discrete and continuous attribute data in out of range zones, being especially useful in mountainous areas or in underdeveloped countries, etc., and for emergences services, hiking activities, tourism, field works, etc.

The proposed solution is based on transforming the geographical attribute data available in vector and raster format to a standard image format in which the colour code is directly related to the identifier of each attribute. Each point of the available map can have two types of information: discrete (for example, a road or a building); and continuous (for example, the building's height or distance from a fixed point). To achieve this, two types of identifiers are set, those with a vector origin (discrete geographic elements) and those of raster origin (continuous geographic elements). Both types of identifiers are translated into binary code and directly related to a plain format database (text file).

The selection of the PNG format as the type of standard image for storing the information is crucial and fulfils several important objectives. Firstly, it is standardised and can be universally applied. Secondly, the use of PNG enables significant information compression without altering the original numeric data. Thirdly, PNG has a great capacity to store information (up to 24 bits) and this enables storing more than 16 million different identifiers. Due to this storage capacity, it is possible to define the identifiers of discrete and continuous data for each colour stored. This storage approach requires defining up to which binary bit each of the identifiers should be read beforehand; but, without a doubt, this approach multiplies the possible uses for this type of solution.

The tests developed with two different products – a topographic map for recreational use and a visitor map of a university campus – have demonstrated a functional character with different types of Nokia phones. Development has shown

the need to set strategies to minimise the use of memory, since this is one of the main limitations when working with mobile devices.

References.

Andrienko, G., and Andrienko, N. (1999) 'Interactive maps for visual data exploration', **International Journal of Geographical Information Science**, 13 (4), pp. 355-374.

Brown, A., Emmer, N and Worn, J.V.D. (2001) 'Cartographic design and production in the Internet Era: The example of tourist web maps', **The Cartographic Journal**, 38 (1), pp. 61-72.

Congalton, R.G. (1997) 'Exploring and evaluating the consequences of vector-to-raster and raster-to-vector conversion', **Photogrammetric Engineering and Remote Sensing**, 63 (4), pp. 425-434.

Dang, G., North, C. and Shneiderman, B. (2001) 'Dynamic queries and brushing on choropleth maps', in **Proceedings of International Conference on Information Visualization**, Piscataway, NJ, USA: IEEE Press, pp. 757-764.

Frank, C., Caduff, D. and Wuersch, M. (2004) 'From GIS to LBS: an intelligent mobile GIS', **GI Days 2004**, July 1st -2nd, 2004, Muenster, Germany.

Gartner, G. and Uhlirz, S. (2001) 'Cartographic concepts for realizing a location based UMTS service: Vienna City Guide Lol@', in **Proceedings of the 20th International Cartographic Conference**, August 06-10, Beijing, China. Vol.5, pp. 3229-3238.

Hunter, J. and Zhan Z. (1999) 'An indexing and querying system for online images based on the PNG format and embedded metadata', in **Proceedings of ARLIS/ANZ Conference**, September 1999, Brisbane, Australia.

Lehto, L., and Kilpeläinen, T. (2000) 'Real-Time Generalization of Geodata in the Web', **International Archives of Photogrammetry and Remote Sensing**, Vol. XXXIII, Part B4, Amsterdam, pp. 559-566.

Lehto, L. and Kilpeläinen, T. (2001) 'Generalizing XML-encoded Spatial Data on the Web' in **Proceedings of 20th International Cartographic Conference**, August 6–10, 2001, Beijing, China, 4: pp. 2390-2396.

Mahammad, S. and Ramakrishnan, R. (2003) 'GeoTIFF – a standard image file format for GIS applications' in **Map India 2003**.

Nagi, R.S. (2004): **Cartographic visualization for mobile applications**. Thesis at the International Institute for Geo-Information Science and Earth Observation, Enschede, The Netherlands.

NIVALA, A.M., SARJAKOSKI, L.T. and T. SARJAKOSKI (2007) 'Usability Methods Familiarity among Map Application Developers' **International Journal of Human Computer Studies**, Elsevier, 65 (9): pp. 784-795.

NIVALA, A.M. and L.T. SARJAKOSKI, (2003) 'Need for ContextAware Topographic Maps in Mobile Devices' in **ScanGIS'2003** – Proc. of the 9th Scandinavian Research Conference on Geographical Information Science, pp. 15-29, Virrantaus, K. and H. Tveite (eds.), Espoo, Finland.

Peursem, J.V., (2002) 'JSR 118: Mobile Information Device Profile 2.0' in: **Java Community Process**, [online]. Available from: <http://www.jcp.org/jsr/detail/118.prt> [Accessed 16 April 2010].

Reichenbacher, T., (2004) '**Mobile Cartography –Adaptive Visualisation of Geographic Information on Mobile Devices**'. Thesis at Institute of Photogrammetry and Cartography, Munich, Germany. ISBN: 3-89963-048-3. Available from: <http://129.187.175.5/lfkwebsite/typo3/sysex/cms/tslib/media/fileicons/pdf.gif> [Accessed 16 April 2010].

Roelofs, G., (1997) 'History of the portable networks graphics (PNG) format' [online]. Available from: <http://www.libpng.org/pub/png/pnghist.html> [Accessed 16 April 2010].

Sakari, T., Antti, O., Kalle, T. and Anu, K., (2004) 'Understanding mobile contexts' **Personal and Ubiquitous Computing**, 8, pp. 135-143.

White, J., (2001) 'An introduction to Java 2 Micro Edition (J2ME); Java in small things' in **Proceedings of the 23rd International Conference on Software Engineering**, 12-19 May, Toronto, Ontario, Canada: IEEE Computer Society, Washington, DC, pp. 724-725.

Wiggins, R., Davidson, H., Harnsberger, H., Lauman, J. and Goede, P., (2001), 'Image file formats: past, present, and future' **RadioGraphics**, 21 (3), pp. 789-798.

Zhao, H. and Shneiderman, B., (2005) 'Colour-coded pixel-based highly interactive web mapping for georeferenced data exploration', **International Journal of Geographical Information Science**, 19 (4), pp. 413-428.

Zipf, A. (2002), 'User-adaptative maps for location-based services (LBS) for tourism', in **ENTER 2002**, Proceedings of the 9th International Conference for Information and Communications Technologies in Tourism, Innsbruck, Austria, pp. 329-338.

List of figures:

Figure 1. Level 1 (a): reference map showing what the user will see in the device; Level 2 (b): coded map; Level 3 (c): attributes.

Figure 2. Rasterization process. (a) initial layers; (b) unique identification of each feature; (c) rasterization of each layer; (d) combination into one raster layer.

Figure 3. Improving user sensitivity with buffer zones. (a) base map. (b) vector features. (c) buffer zones according symbols size. (d) combined raster.

Figure 4. Coding process of an ID within the RGB colour code. An ID value of 1000 corresponds to (0,3,232) RGB value.

Figure 5. Joint coding process of an ID and the elevation value for a specific pixel.

Figure 6. Rules for the coding and decoding of elevation values.

Figure 7. (a) Original map; (b) coding of the IDs; (c) coding of the elevation values; (d) joint coding.

Figure 8. Database file sample for topographic maps test.

Figure 9. Original visitor map (a); coded visitor map (b).

Figure 10. Database file sample for campus test.

Figure 11. Application prototype for the visualisation and query of the raster-attribute hybrid method in the Nokia N95 mobile phone. Left image shows the topographic map application test; while right image shows the visitor map application test.