Research on web vector representation of thematic map using HTML5

Beiqi SHI, Neng CHEN, Lin SHEN, China

Key Words: Thematic Map, Vector, HTML5, Canvas API

SUMMARY

HTML5 is being developed as the next major revision of HTML (Hypertext Markup Language), the core markup language of the World Wide Web. It aims at reducing the need for proprietary, plug-in-based rich Internet application (RIA) technologies such as Adobe Flash. The canvas element is part of HTML5 and is used to draw graphics using scripting (e.g., JavaScript).Firstly, this paper introduces the feature and new function of HTML5. After that, Web Representation of Vector Based on Canvas API of HTML5 is described in detail. It is an open-source, vector-based, client-side framework for rendering plug-in-free, offline-capable, interactive maps in native HTML5 on a wide range of Web browsers and mobile phones. Potential applications of the technology as an enabler for participatory online mapping include mapping real-time air pollution, citizen reporting, and disaster response, among many other possibilities. In experiment, the data of the thematic map of public cultural facilities in Shanghai was stored in XML format. Via JavaScript and the Canvas API of HTML5, the thematic map can be achieved without implementing any plug-ins. At last, we draw the conclusions.

Research on web vector representation of thematic map using HTML5

Beiqi SHI, Neng CHEN, Lin SHEN, China

1. Introduction

Thematic map is a map that focuses on a specific theme or subject area (William, 2007). It is referred to as diagram that portrays spatial variations and interrelationships of geographical distributions. The majority of the thematic maps are used for geographic scientific research results. With the rise of Web2.0, instead of desktop applications, web applications on thematic map are more and more popularly used in many fields (Corcoran et al., 2011). Despite this progress, web-mapping applications still significantly linger behind their corresponding desktop counterparts in terms of both interface and functionality (Shi et al., 2009). Much of the discrepancy can be attributed to a number of limitations of the current version of HTML (Corcoran et al., 2011). Most web applications on thematic map use raster tiles not vector graphics, like ArcGIS API, OpenLayers API, Google map API et al. Usually, the server transmits data to the raster tile, and then renders the result on Google map or their own Web Map Server (WMS). Sometimes the content of thematic map is too intensive to show data clearly. Each render generates an Http request and sends to the server. This may increase network latency and server stress rapidly. Another kind of method of web publication thematic map is using the Scalable Vector Graphics (SVG). SVG is developed by the World Wide Web Consortium (W3C) SVG Working Group starting in 1999. SVG is a family of specifications of an XML-based file format for two-dimensional vector graphics, both static and dynamic. However, SVG is not convenient for users. Users just can browse the data in vector format via Adobe SVG Viewer, which is not convenient to common users. Due to the new major revision of HTML, those issues can be overcome. The reminder of the paper is contributed as follows. In section 2, we introduce features and new functions of HTML5. Section 3 describes web representation of vector based on Canvas API of HTML5. Section 4 provides an experiment for representation of thematic map. Finally, we drew the conclusion about vector thematic map of web representation using HTML5.

2. Features of HTML5

HTML5 is being developed as the next major revision of HTML, the core markup language of the World Wide Web (Hickson, 2010). It aims at reducing the need for proprietary, plug-in-based RIA (rich Internet application) technologies such as Adobe Flash, Microsoft Silverlight, and Oracle-Sun JavaFX (Boulos et al., 2010).

The canvas element is part of HTML5 and is used to draw graphics by JavaScript (Hickson, 2010). Canvas API is an immediate mode bitmapped area of the screen that can be manipulated with JavaScript. Immediate mode refers to the way the canvas renders pixels on the screen. HTML5 Canvas completely redraws the bitmapped screen on every frame using Canvas API. This makes HTML5 Canvas very different from Flash, Silverlight, or SVG, which operate in retained mode. In this mode, a display list of objects is kept by the graphics renderer, and objects are displayed on the screen according to attributes set in code (i.e., the x position, y position, and alpha transparency of an object). The feature of Canvas API keeps the programmer away from low-level operations, but gives the less control over the final rendering of the bitmapped screen (Fulton, Fulton, 2011).

Before the arrival of HTML5, it is difficult that a website with RIA is compatible with all web browsers, especially in safari on APPLE MAC OS or iPhone. Similarly, Apple has dropped Flash support on its iPad device in favor of HTML5 (Grothaus, Sadun, 2011). As instead of using computers, nowadays users prefer surfing the Internet via a mobile phone or tablet PC. Thus, it is necessary that a website should be browsed on not only PC but also mobile phone and tablet PC. Canvas API makes a higher compatibility, faster rendering speed, lower network latency and stress on the server, and it works as excellent as RIA. Most websites begin to pay attention to using HTML5 instead of RIA. For example, YouTube is planning to use HTML5 version of its service instead of Adobe Flash.

3. Web Representation of Vector Based on Canvas API of HTML5

The basic HTML5 Canvas API includes a 2D context in which various shapes, render text, and display images can directly drew onto a defined area of the web browser window (Fulton, Fulton, 2011). It is available to draw a thematic map using Canvas API, since Canvas API is a way of rendering graphics on a specific region.

3.1 Vector graphic using Canvas API

To draw a vector graphic using Canvas API, two elements are necessary. Firstly HTML5 should be defined. In HTML5, there is only one declaration. A sample looks like this:

```
<!DOCTYPE html>
```

Another is the declaration of canvas element. It looks like this:

```
<canvas id="myCanvas"></canvas>
```

It is better to add the ID for call the canvas by JavaScript. A canvas also can be determined the size using its attributes, height and width. A sample looks like this:

```
<canvas id="myCanvas" height=”400px” width=”300px”></canvas>
```

Canvas is ready, the rest of the brush, JavaScript. To begin with, the canvas element should be found with the Document.getElementById(), and then the Canvas API is initialized. A simple looks like this:

```
<!DOCTYPE html>
<html>
<body>
<canvas id="myCanvas"></canvas>
<script type="text/javascript">
var canvas=document.getElementById('myCanvas');
var ctx=canvas.getContext('2d');//define a 2D context
ctx.fillStyle='#FF0000';
ctx.fillRect(0,0,80,100);
</script>
</body>
</html>
```

And Figure 1 shows the result in FireFox browser.

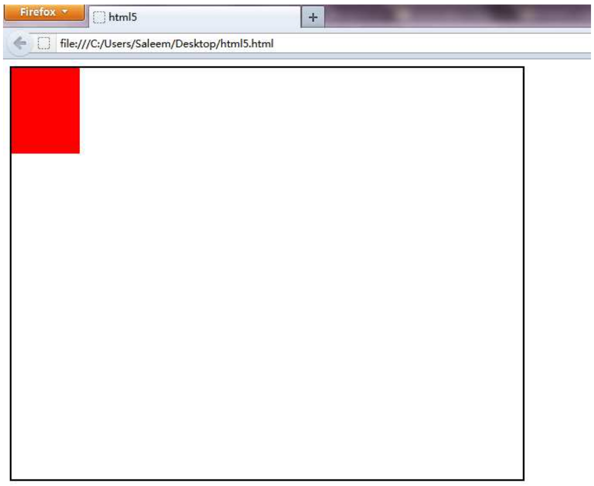


Figure 1: Result of drawing vector graphic using Canvas API

FillRect(x, y, w, h) is the function to draw a filled rectangle, and it needs four parameters, which express the position and size of the rectangle. FillStyle is the attribute of the rectangle’s color.

In Canvas API, there are three types of graphic format, which includes shape, text, and image, which is shown in Figure 2.

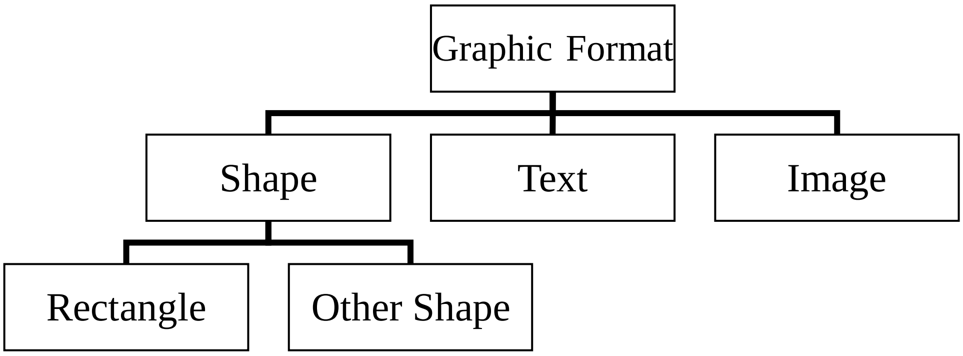


Figure 2: Three types of graphic format

Rectangle is the only basic graphic and other shape can be drawn with the function Canvas Path. beginPath() means the beginning of the path and closePath() means the end. A simple of circle and Triangle looks like this:

```
ctx.beginPath();
ctx.strokeStyle = "black";
ctx.lineWidth = 5;
ctx.moveTo(20,20);
ctx.lineTo(20,100);
ctx.lineTo(70,100);
ctx.lineTo(20,20);
ctx.stroke();//draw a triangle
ctx.moveTo(400,200);
ctx.arc(200, 200, 200, (Math.PI/180)*0, (Math.PI/180)*360, false);
ctx.stroke();//draw a circle
ctx.closePath();
```

Figure 3 is the result of circle and triangle using Canvas API .

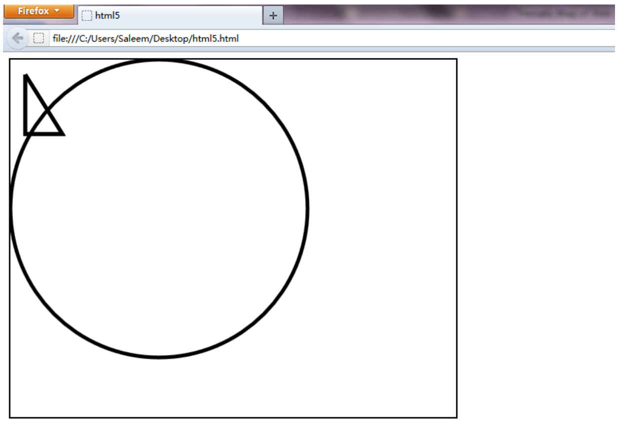


Figure 3: Result of circle and triangle using Canvas API

3.2 Thematic Map of Vector Representation via Canvas API

The first step of representing vector thematic map is initialization. In Canvas API, the function createImageData(width,height) is used to create a blank imagedata object. A sample shows as follow:

```
ctx.createImageData(map_hScreen, map_vScreen);
```

Due to the difference between geographic coordinate and screen coordinates, It is necessary to transmit geographic coordinate to screen coordinate, with the function like this:

```
map_Rect.Rx = cpt.x + map_hScreen/map_Scale/2;
map_Rect.Ly = cpt.y + map_vScreen/map_Scale/2;
```

map_Scale is the parameter of map scale. Position of the map in the canvas can be set by the center coordinates of map and the function createImageData(width,height).

After the map initialization, it is essential to load data. The data of thematic map is stored in XML format, which contains coordinate and its style. Canvas API uses the appropriate style according to the graphic type. The coordinates are retrieved from XML file and stored in JSON temporarily. The function lineTo(x,y) connects each coordinate in JSON one after another, and then the thematic map can be drawn.

4. Example of Vector Thematic Map of Web Representation Using HTML5

Followed Section 3, an example of vector thematic map of web representation using HTML5 was designed. The data stored in the server, and users can browse the vector thematic map in the client via HTML5.The browser showed the vector thematic map, the basic elements of a thematic map and the map controller bar, with Figure 4 Showed the result.

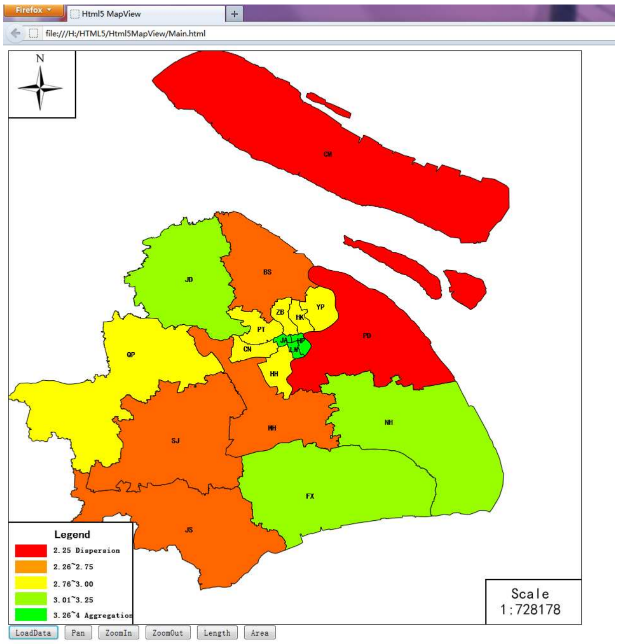


Figure 4: example of Vector Thematic Map of Web Representation Using HTML5

There is the vector thematic map in the centre. It reflects the characteristics and spatial pattern of the number of public cultural facilities in Shanghai. The vector thematic map can be zoomed in, zoomed out and paned. In the client, user can load data and view the thematic map by the map control bar, which is controlled with JavaScript (Figure 5). Due to processing all operations on the client, it avoids to send HTTP request to the server.

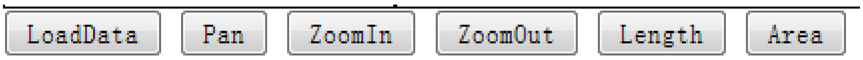
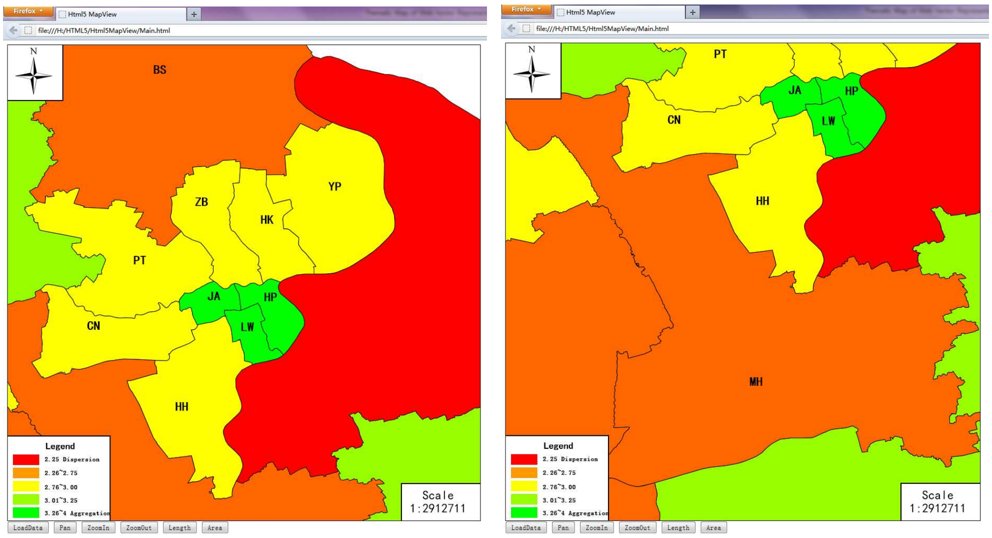


Figure 5: the map control bar

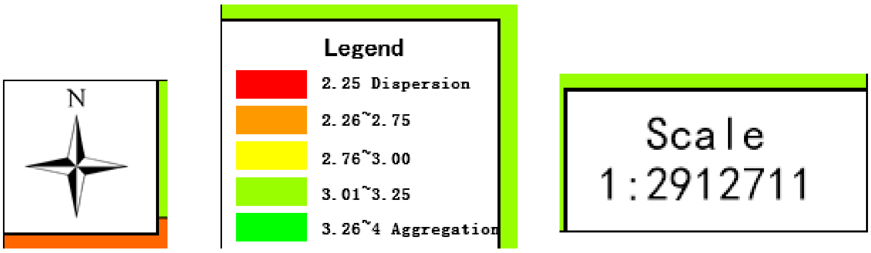
Users can add the data by pressing the button LoadData. The thematic map can be viewed with the button Pan, ZoomIn, ZoomOut . The geographic parameters such as distances and areas can also be measured with the button Length or Area. Figure 6(a) shows the result of zoom-in, and Figure 6(b) shows the result of panning.



(a) (b)

Figure 6: Result of zoom-in and panning

Northarrow, legend and scale as the basic element of thematic map should be represented in the browser. These were realized by the function ctx.filltext() and ctx.drawImage() in Canvas API. Figure 7 shows the result of thematic map elements.



(a) (b) (c)

Figure 7: Elements of thematic map

5. Conclusions

Due to the limitations of previous HTML revision, the web browser cannot represent the vector data. To overcome the issue, some vector representations are presented, but most of them need plug-ins, which possibly contains the risk to users. As a new element in the HTML5, Canvas API has a good effect in the rendering of vector data. In this paper we demonstrate that HTML5 represents a non-plug-in platform for vector thematic map. This platform gives users a less rendering speed, lower network latency. It also gives programmers another choice in WebGIS field. Compared to the traditional thematic map, it is more interactive for user and easier for information sharing in the government and education fields.

Acknowledgement:

This work is supported by Innovation Program of Shanghai Municipal Education Commission (Project ID: 11YZ290). It is also supported by Leading Academic Discipline Project of Shanghai Municipal Education Commission (Geography and City Environment)(Project ID: J50402) and Leading Academic Discipline Project of Shanghai Normal University (Project ID: DZL801)

Reference

Boulos, M.K., Warren, J., Gong,J., Yue, P., 2010, Web GIS in practice VIII: HTML5 and the canvas element for interactive online mapping, International Journal of Health Geographics 9(1), 14-15.

Corcoran, P., Mooney, P., Winstanley, A., Bertolotto, 2011.,Vector Data Transmission and Visualization Using HTML5. Proceedings GISRUK, Portsmouth.

Fulton, S., Fulton, J., 2011. HTML5 Canvas. O'Reilly Media, Inc., California.

Grothaus, M., Sadun, E., 2011, Taking Your IPod Touch to the Max, IOS 5 Edition, third ed. Apress, New York.

Hickson, I., 2010, A vocabulary and associated APIs for HTML and XHTML, http://www.w3.org/TR/html5/.

William, N.J.T., 2007. Maps & Civilization: Cartography in Culture and Society. University Of Chicago Press, Chicago.

Shi, W., Kwan, K., Shea, G. and Cao, J.2009.A dynamic data model for mobile gis, Computers and Geosciences 35 (11), 2210 – 2221.

CONTACTS

Research on web vector representation of thematic map using HTML5
First author: Beiqi SHI
Shanghai Institution of Tourism
No.500 HaiSi Rd., Shanghai, China, 201418
Corresponding author: Neng CHEN
Urban Information Research Center, Shanghai Normal University
No.100 Guilin Rd., Shanghai, China, 200234
Email: ch5082@shnu.edu.cn
Co-author: Lin SHEN
Urban Information Research Center, Shanghai Normal University
No.100 Guilin Rd., Shanghai, China, 200234