Computer aided generation of stylized maps

Neeharika Adabala and Manik Varma and Kentaro Toyama

Microsoft Research India

Scientia, 196/36 2nd Main

Sadashiv Nagar, Bangalore

India, 560 080

email: {neeha,manik,kentoy}@microsoft.com

Abstract

Geographic maps have existed from early stages of human civilization. Various styles of visualizing the geographic information have evolved depending on the nature of information and the technology available for visualization. This has led to innumerable map styles. In this work we develop a technique to create maps by combining two-dimensional and three-dimensions information such that the resulting maps are both functional and aesthetically appealing. Our technique requires geographical information in vector form and aerial images as inputs. We use computer vision based approaches and user defined inputs to augment the vector data with information that is required to create stylized maps. We define procedural graphics methods to generate a range of geographic elements that can be composed together into a stylized map. We demonstrate our technique by generating example maps of a region in Las Vegas.

Keywords: cartography, geographic maps, vector data, computer vision, stylized rendering

1 Introduction

Maps have been a means of representing spatial information from the dawn of human civilization. Numerous types of maps have existed depending on the information that is to be visualized and the artistic inclination of map-makers. Maps especially those of cities can exhibit large variation in style based on the technique employed by a map-maker, this is validated by the existence of a wide variety of tourist maps. These maps have the distinction of enabling maximum style variations by combining two-dimensional data in the form of street information, three-dimensional data in the form of buildings, and other elements like trees that enhance the artistic value. They are an example of maps that achieve a unique balance between functionality and aesthetic appeal. These maps form the focus of this work, and our goal is to develop a technique that enables the creation of such maps with the aid of a computer.

The main issues that one has to address to enable the creation of stylized maps are:

- Augment vector data: a typical geographic database representing city information contains two-dimensional data in of point features, networks of streets, or boundaries of regions example administrative boundaries, etc. It rarely contains information on shapes of buildings, locations of trees and pools. Techniques that enhance a typical vector-data set with these elements are required.

- Rendering of two-dimensional data: Vector data like streets, and boundaries of regions have to be rasterized to create a map. This is the most well explored aspect in map-making.

- Rendering of three-dimensional data: Traditionally the most commonly occurring three dimensional data in the context of maps has been the elevation model for the underlying surface of the earth. But here we are interested in including details of three dimensional elements like buildings and trees. The challenge is to develop techniques that create stylized two-dimensional representations, for three-dimensional shapes, that can be composed together to form a map.

We develop a solution that address each of these issues, and this results in a system that enables us to generate stylized maps. It has application in the context of recent online GIS systems like [1] and [2] that include three dimensional representations of cities, and also in computer games that contain urban scenes.

The rest of the paper is organized as follows. The next section briefly mentions some related work, we described methods that enable the augmentation of vector data with the help of computer vision based techniques and user inputs in section 3. The section 4 describes rendering techniques, we focus on the rendering for three-dimensional elements for the purposed of creating stylized maps. We summarize the algorithm in section 5. Result are presented and discussed in section 6 and we conclude by giving directions for future work in the final section.

2 Related Work

Previous work in this area include the vast literature on thematic cartography, which is rich with qualitative descriptions of what is required to design maps [3, 4, 5], contain the fundamental concepts required for map making.

Other related work that investigates creating maps using computer graphics approaches, is the work of Agrawala's on LineDrive [6]. In this work, route maps are abstracted dramatically through compressed distances and simplified details, for the sake of capturing *only* the essential features of driving directions from one point to another [6]. The output is a clear, 2D rendition of a route-map as might be drawn by hand. LineDrive thus focuses on legibility as the key attribute, and while there was some experimentation with a hand-drawn look, there was no intent to output maps that incorporated creative artistic effects.

Stylized maps are most similar in spirit to certain kinds of rendering in the area of computer graphics called non-photorealistic rendering (NPR). In NPR, the goal is not a to generated scene that appears realistic like a photograph, but to create an artistic version thereof; some fidelity to an underlying photograph or vector representation is expected, but realism is traded for a more creative look [7]. DeCarlo and Santella [8] describe a scheme to stylize photographs through homogenization of texture and emphasis of critical edge lines by considering what viewers pay attention to in the original. Sidiropoulos and Vasilakos [9] present a technique to create visualizations of urban regions the focus on comparing realistic and symbolic visualizations. Rakkolainen and Vainio [10] describe a technique to augment maps with three-dimensional information of buildings using VRML for mobile users.

The problem of generating stylized maps that look like ”tourist maps” or ”town plans” of the 19th century with the aid of a computer has not been directly addressed so far. We describe how this can be done by presenting details in the context of the issues identified in the previous section. The first problem is to create the additional data that is required for creating such stylized map, solutions to this problem are discussed in the following section.

3 Augmenting of data

Our approaches to generate the required data for stylized maps employ:

- computer vision based methods applied to satellite images, and
- user input with guidance from satellite images.

Data collected in this manner may possess unnecessary details that clutter information, therefore a technique that refines the data obtained from these approaches is described. The following three sub-sections, of this section, briefly outline the above mentioned techniques.

3.1 Computer vision based augmenting of vector information

This approach mainly relies on texture classification in aerial images to generate additional data. Classifying image regions on the basis of their textural appearance is a well studied problem, both in computer vision and in remote sensing [11, 12, 13, 14, 15, 16, 17, 18]. The fundamental idea is to characterize image regions as statistically textured areas and to differentiate between them as such. Remotely sensed images are particularly amenable to this kind of modeling and we apply the method of [19] to a database of aerial images collected from Windows Live Local [1].

Given the context of generating a stylistic map, the objective is to automatically detect vegetation, water and car parks in aerial images. Each class is modelled as a probability distribution over a *texton* library [20]. Following [21], textons are defined to be image patch exemplars, and are learnt from the training set by clustering all image patches within a class. In our implementation, 5×5 colour patches are considered and 5 textons are learnt per class using the *K-Means* algorithm. Next, a texture model corresponding to a particular training region is generated by vector quantizing all its image patches into textons and then learning the texton frequency distribution. Multiple models per class (one for each image region in the training set) are generated to account for statistical variability. More details can be found in [19, 21].

Classification of a novel image is carried out by partitioning it into small non-overlapping regions. A nearest-neighbour classifier employing the χ^2 statistic as the similarity measure is then used to classify each region individually. Background regions are rejected if their distance to any class is above a learnt threshold. Figure 1 shows typical classification results for a novel image.

Figure 1: Texture classification of a novel image: On the left is the image submitted to the classifier and on the right are the detections.

As can be seen the classifier accurately picked out the trees and swimming pools in the image. The classification threshold is set conservatively so as to minimize the number of false positives.

3.2 User input for building information

While it is possible to identify the textures for areas in the map and create additional data for regions in a map, it is relatively difficult to find the boundaries of building foot-prints automatically through edge detection methods. Therefore, we have developed a tool that enables users to load the satellite image of a region and mark out the building foot prints by using a mouse. The user interface for this purpose is illustrated in the figure 2.

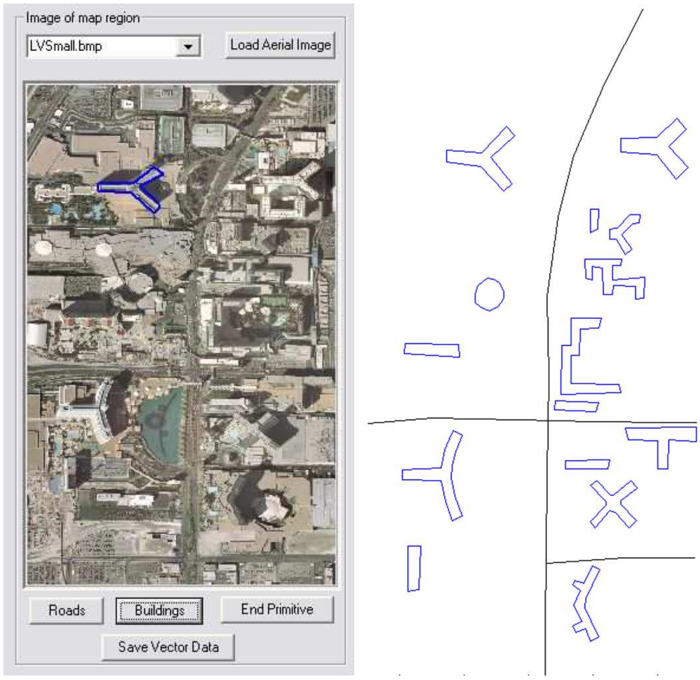

Figure 2: User interface for marking the foot prints of buildings and creating other vector data that is required for map making. The resulting vector data is shown to the right.

3.3 Feature-straightening algorithm

Our goal is to strike a balance between accuracy on the one hand and clarity and aesthetics on the other. This is an alternative approach to the well established Douglas-Peucker algorithm [22] for the reduction of the number of points required to represent a digital line or its caricature. We employ model selection to generalize *straight* roads to be piecewise linear and eliminate kinks. In doing so, we sacrifice some fidelity to ground truth but generate *stylistic* roads with greatly enhanced clarity and aesthetics. The model selection algorithm proceeds as follows.

Given a vector representation of a straight road, we model it by a series of contiguous line segments. The optimal number of line segments is detected automatically (see figure 1) and this determines the degree of the model. Given the optimal number of line segments, the algorithm places each segment according to the following rules: (a) the terminal points of each segment must be straight road vector points themselves; (b) the acute angle between two adjacent line segments must either be $45°$ or $90°$; (c) the placement of the line segments must minimize the overall difference in shape between the straight and stylistic roads.

Note that there is no universally accepted metric for determining shape similarity. The Hausdorff distance or other measures, such as those defined in [23, 24, 25], apply to only a limited number of cases. However, within our framework, a good asymmetric similarity measure is given by $\epsilon = \sum_i \|\vec{r}_i - \vec{s}_i\|$ where $\vec{r}_i$ is a straight road vector point and $\vec{s}_i$ is the closest point to it on the stylistic road.

It is clear that the higher the degree of the model, the lower ϵ will be. However, the number of kinks and other artifacts will increase correspondingly. We therefore need to select the number of line segments appropriately. This is done by choosing the model that minimizes

$$N^* = \mathit{argmin}_N \;\; N \Theta_N \epsilon_N \tag{1}$$

where N is the number of line segments in the model and $\Theta = \sum_i \theta_i$ is the sum of the angles θ_i between adjacent line segments of the stylistic road. Figure 3 highlights the effects

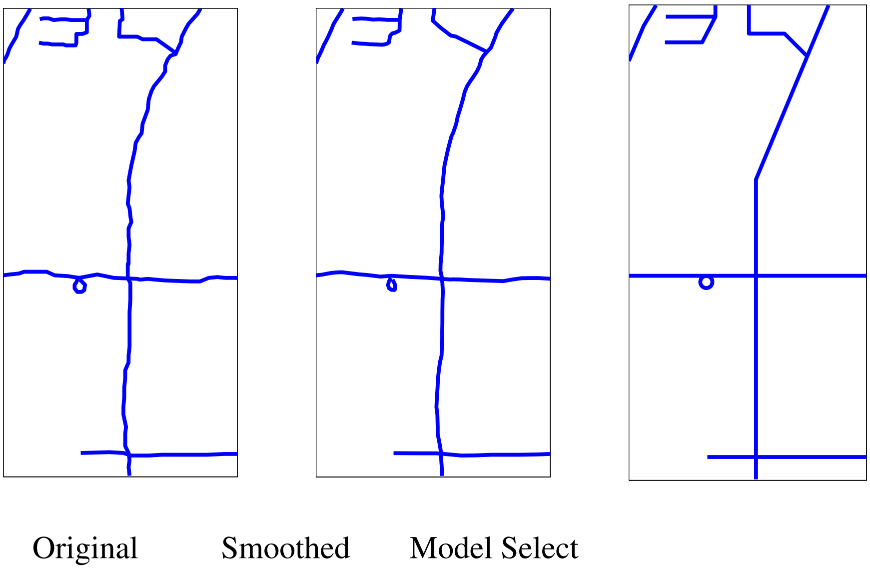

Figure 3: The original straight road is stylized using model selection. This has dramatically straightened the roads and reduced the number of kinks even as compared to smoothing.

of model selection and compares its advantages to smoothing (where each point on the road is adjusted so as to be a linear combination of its neighbours).

4 Rendering techniques for three-dimensional elements

In this section we present the rendering techniques developed by us that use the vector data created by the approaches presented in the previous section and generate stylized maps. The rendering techniques are crucial as they decide the appearance of the map. We present details in the context of rendering of trees and buildings in the following two subsections.

4.1 Rendering stylized trees

The input to the procedure that generates trees is the region of tree cover as obtained from the vision algorithm and the density of trees. A raster mask of possible regions where trees are located in generated. The locations of trees within this region is generated by random sampling of the region. The number of tree in a region is dictated by the density term.

The details of how a tree is rendered is defined by the kind of tree. We have considered creating models for coniferous, deciduous and palm trees. The model are simplistic as they are iconographic representations in the map. There are two main aspects that define the model:

- geometric primitives, and
- appearance defining parameters.

The appearance of the map is dependent on the selection of these parameters.

One of approach to create tree representations is as follows:

- Geometric Primitives: the primitives include a distribution of lines that radiate from the central line (trunk of tree) towards the periphery representing the envelop of the tree.
- Appearance defining parameters: in the current model defined as above these parameter include the number of primitives (lines) used per tree, the width of lines, stipple pattern and color.

Alternative styles are created by either varying the kind of primitives used, or the the parameters that dictate the appearance - like color, stipple pattern, etc. Example images of the resulting trees are presented in the figure 4. The top row presents tree models that are utilized in stylized maps of the 19th century, they are created with fine lines that are stippled and have the appearance of being drawn with a fine pen. The middle row presents a modernistic appearance created by using broader lines and brighter colors for the geometric primitives. The final row represents the stylized versions of trees often used in architectural plan drawing while defining landscaping around buildings. There the geometric primitives that define a tree have been changes. The trees can also be represented by using intersecting sprites with textures, which is a popular technique adopted in current day games.

Figure 4: Example tree styles created by varying the geometric primitives, and appearance defining parameters.

4.2 Rendering of buildings

The building models are created by extruding the building footprint that is created by tracing the outline in an aerial image. The appearance of the building is mainly varied by changing color and textures applied to the facades. Example facade textures used for buildings are given in figure 5. The rendering style adopted for buildings emphasizes the edges of the polygons that define the building to create the hand-drawn effect.

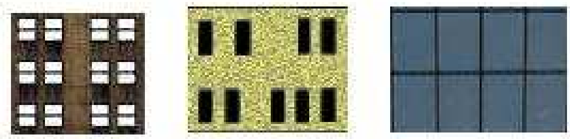

Figure 5: Example building facade textures

We have thus far presented solutions to individual issues that are to be addressed to enable creation of stylized maps. In the following section we briefly present how these techniques are applied together to generate stylized maps.

5 Summary of proposed technique

Figure 6 presents an overview block diagram of the data and processes that enable computer aided generation of stylized maps. The input data to the system is in the form of satellite images and vector data of streets if available. If the vector data is not available then it is created through image processing of aerial photographs, and applying the feature straightening algorithm. If the vector information includes elevation information for the ground plane then this information is displayed in the map by applying exaggerated shading techniques similar to the ones described in Rusinkiewicz et al. [26]. The vector data is augmented with features that are typically not found in geographic databases of urban areas (such as trees) and buildings. This is done by employing computer vision based techniques to recognize trees in the region and enabling the user to mark footprints of buildings and input heights. Algorithm then abstracts and simplifies map features are applied to the feature before they are used. Finally, non-photorealistic rendering techniques that have parameters to generate various styles are applied to the data to result in stylized maps. One of the limitations of the

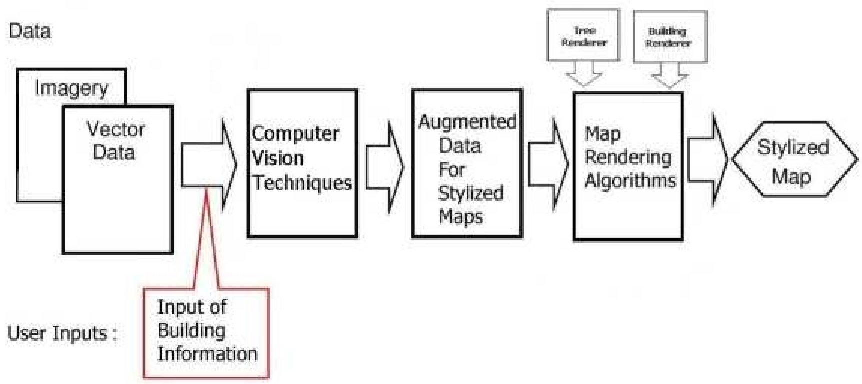

Figure 6: Overview of stylized map generation procedure.

approach is the need to delineate the buildings. In some cases when the user is interested in depicting only a few key buildings in the region this approach is acceptable. However, if all the buildings in a region are to be represented, then the current manual technique of marking footprints is laborious and technique to automate this step are to be developed.

6 Results and discussion

To illustrate the versatility of our approach, we presents maps rendered in two styles, one that is in the artistic style of hand drawn maps of the nineteenth century and the other has a modernistic appearance of current day tourist maps. The map is of an area in downtown Las Vegas that is seen in the figure 2.

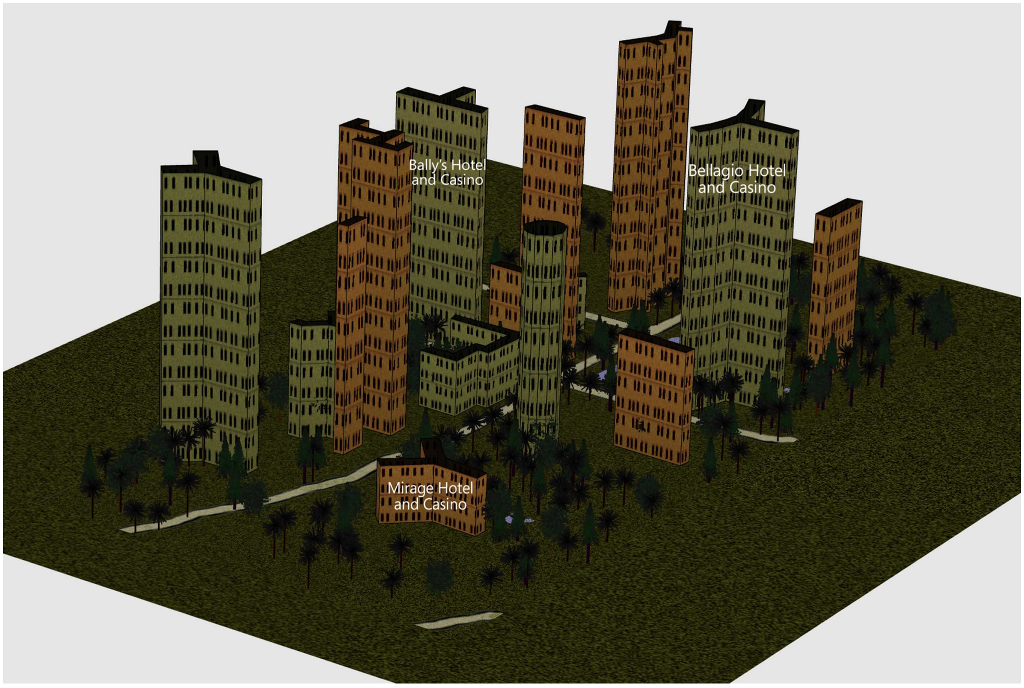

Figure 7: An semi-automatically generated map a historic style panoramic.

First, we generate an image in the style of oblique historic style panoramic maps created by artists of the eighteenth and nineteenth century. These kind of maps require extensive artist effort and as the urban areas grew in size the art of making such maps has disappeared. The original hand-drawn panoramic maps were meticulous effort of artists who spent several months to create each map [27]. Though somewhat staid in nature, these type of panoramic maps have a simple, fresh look with the following features: a near orthographic projection; dark outlines of planar surfaces of buildings; light, granulated shading for the ground and for building surfaces (with different colors indicating different building materials); small black rectangles for windows; and a rich representation of trees.

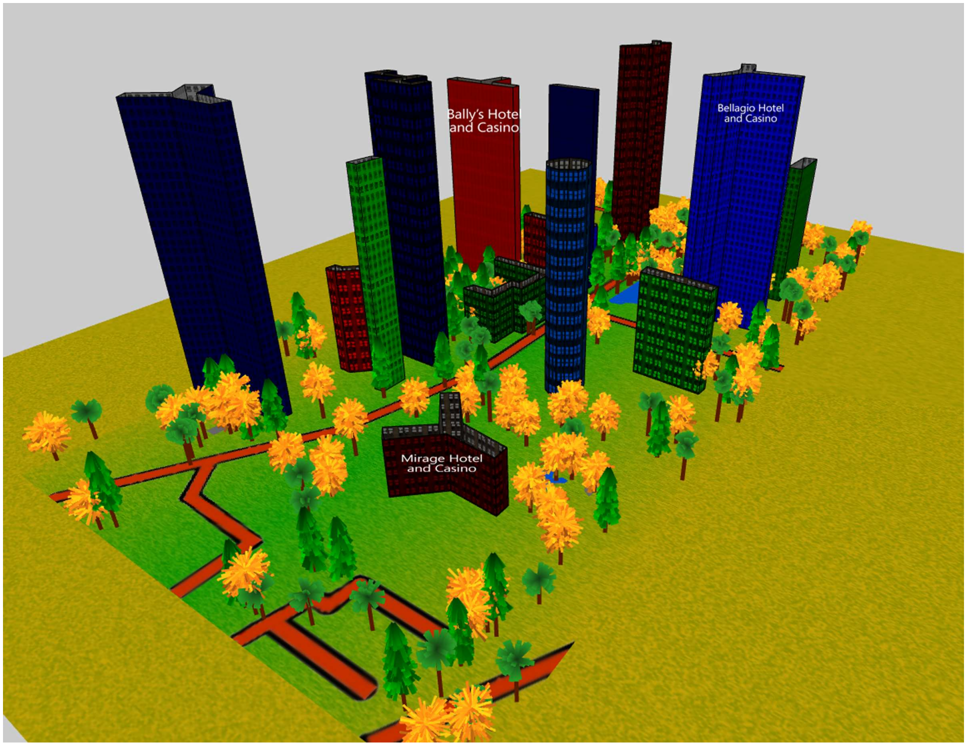

Figure 8: A semi-automatically generated stylized map in a colorful map style.

In figure 7, we show our output, which captures these elements by choosing a near orthographic projection, outlined polygons, procedurally generated texture maps which include Perlin noise, a texture map of small, black windows, and a number of types of procedurally generated trees.

By altering a few parameters to the rendering procedures in the program we generate a map with a colorful appearance more in the spirit of cartoon tourist maps.

In particular, we use saturated colors, employ perspective projection, a apply a varied collection of textures to building surfaces, and then render the trees with different appearance parameters. The resulting map is shown in figure 8. Together this map and the historic style panoramic map illustrate the range of stylized maps that can be generated with just a few minor changes in parameters and texture.

7 Conclusions and future work

We have described a set of algorithms that allows a user to easily generate stylized maps, based on available map data and aerial imagery combined with a few indications from the user specifying the desired artistic effects. This work creates a framework for future efforts to create stylized maps. A prototype system that can output maps that display a range of different styles was developed to demonstrate the possibilities.

One of the main direction of future work that we are considering is handling of occlusion that results from overlapping multi-storied buildings in oblique view. We will be exploring possibilities for moving buildings in their place by small extents to allow visibility as illustrated in the figure 9.

We expect that such manipulations will enable us maximize the information that we can visualize in a map by avoiding loss of information due to overlapping of buildings.

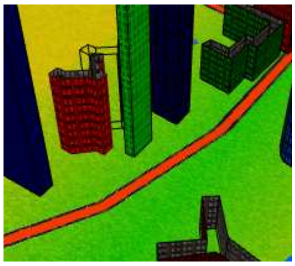

Figure 9: Visibility adjustment to avoid occlusions (work in progress).

This problem is especially relevant in the context of modern downtown urban areas that are cluttered with multi-storied buildings.

References

[1] Microsoft. Virtual earth 3d - online map service. http://maps.live.com/, 2007.

[2] Google. Google earth - broadband, 3d application. http://earth.google.com/, 2007.

[3] A. M. MacEachren. *How Maps Work*. The Guilford Press, 1995.

[4] M. Monmonier. *Mapping It Out*. University of Chicago Press, 1995.

[5] Terry A. Slocum, Robert B McMaster, Fritz C. Kessler, and Hugh H. Howard. *Thematic Cartography and Geographic Visualization, Second Edition*. Prentice Hall, 2003.

[6] Maneesh Agrawala and Chris Stolte. Rendering effective route maps: Improving usability through generalization. In *Proceedings of SIGGRAPH 2001*, pages 241–250. ACM Press / ACM SIGGRAPH, 2001.

[7] B. Gooch and A. Gooch. *Non-Photorealistic Rendering*. A. K. Peters, 2001.

[8] Doug DeCarlo and Anthony Santella. Stylization and abstraction of photographs. In *Proceedings of SIGGRAPH 2002*. ACM Press / ACM SIGGRAPH, 2002.

[9] George Sidiropoulos and Athanasios Vasilakos. Ultra-real or symbolic visualization? the case of the city through time. *Computers & Graphics*, 30(2):299–310, April 2006.

[10] Ismo Rakkolainen and Teija Vainio. A 3d city info for mobile users. *Computers & Graphics*, 25(4):619–625, August 2001.

[11] S. Konishi and A. L. Yuille. Statistical cues for domain specific image segmentation with performance analysis. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 125–132, Hilton Head, South Carolina, June 2000.

[12] S. Lazebnik, C. Schmid, and J. Ponce. Affine-invariant local descriptors and neighborhood statistics for texture recognition. In *Proceedings of the International Conference on Computer Vision*, volume 1, pages 649–655, Nice, France, October 2003.

[13] A. Lorette, X. Descombes, and J. Zerubia. Texture analysis through a markovian modelling and fuzzy classification: Application to urban area extraction from satellite images. *International Journal of Computer Vision*, 36(3):221–236, February 2000.

[14] G. Rellier, X. Descombes, F. Falzon, and J. Zerubia. Texture feature analysis using a gauss-markov model in hyperspectral image classification. *IEEE Transactions on Geoscience and Remote Sensing*, 42(7):1543–1551, 2004.

[15] C. Schmid. Weakly supervised learning of visual models and its application to content-based retrieval. *International Journal of Computer Vision*, 56(1):7–16, 2004.

[16] M. Simard, S. S. Saatchi, and G. de Grandi. The use of decision tree and multiscale texture for classification of JERS-1 SAR data over tropical forest. *IEEE Transactions on Geoscience and Remote Sensing*, 38(5):2310–2321, September 2000.

[17] Norbert Haala and Claus Brenner. Extraction of buildings and trees in urban environments. *ISPRS Journal of Photogrammetry & Remote Sensing*, 54(2-3):130–137, 1999.

[18] Mena J. B. State of the art on automatic road extraction for gis update: a novel classification. *Pattern Recognition Letters*, 24(16):3037–3058, December 2003.

[19] M. Varma and A. Zisserman. Texture classification: Are filter banks necessary? In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 2, pages 691–698, Madison, Wisconsin, June 2003.

[20] T. Leung and J. Malik. Representing and recognizing the visual appearance of materials using three-dimensional textons. *International Journal of Computer Vision*, 43(1):29–44, June 2001.

[21] M. Varma. *Statistical Approaches To Texture Classification*. PhD thesis, University of Oxford, October 2004.

[22] D. Douglas and T. Peucker. Algorithms for the reduction of the number of points required to represent a digital line or its caricature. *Canadian Cartographer*, 10(2):112–122, 1973.

[23] S. Belongie, J. Malik, and J. Puzicha. Shape matching and object recognition using shape contexts. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(4):509–522, April 2002.

[24] A. C. Berg, T. L. Berg, and J. Malik. Shape matching and object recognition using low distortion correspondence. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 26–33, San Diego, California, June 2005.

[25] E. Sharon and D. Mumford. 2D-shape analysis using conformal mapping. *International Journal of Computer Vision*, 2006. To appear.

[26] Szymon Rusinkiewicz, Michael Burns, and Doug DeCarlo. Exaggerated shading for depicting shape and detail. *ACM Transactions on Graphics (Proc. SIGGRAPH)*, 25(3), July 2006.

[27] Geography and MapsDivision. *The Library of Congress: Panoramic maps collection*. 2007. http://memory.loc.gov/ammem/pmhtml/panhome.html.