

Fractal Cityscape



Baihua Li¹ and James Miller²

¹ School of Computing, Mathematics & Digital Technology, Manchester Metropolitan University, M1 5GD, UK

² Department of Computing and Mathematics, University of Hull, HU6 7RX, UK

Abstract—There is an increasing demand for the simulation of large-scale "free-flow" urban environments in games or other virtual reality worlds. Currently most 3D cityscapes are generated by time consuming manually pre-scripted "room-to-room" approaches. Although the effectiveness of fractal modelling of rural landscape or architecture has been studied extensively, there is a relative dearth of research on fractal modelling of cityscapes.

Inspired by modern fractal city theory, we intend to develop an automatic fractal-based method to address this problem. In particular, we will investigate how fractal theory can be used to mathematically simulate the multi-level hierarchical urban structure corresponding to various land use property probabilities of city regions. Tentative findings and results are presented in this paper.

Index Terms—Cityscape modelling, fractal city, fractal geometry, virtual reality world.

I. INTRODUCTION

It seems the world is chaotic, random, discontinuous, irregular and complex in its superficial physical form. Scientifically, efficiently generating and visualizing realistic natural phenomena has been a difficult problem. However, beneath this first impression of the natural world, there is inherent regularity, self-similarity, hierarchy and organization in nature [1]. With this insight, fractal theory may provide not only the theoretical foundation but also a practical solution for mathematically modelling highly realistic graphical scenes of the natural world [2].

A fractal is generally a rough fragmented geometry that can be split into parts at infinite scales, each of which is at least approximately a reduced-size copy of the whole [2]. Fractals are specifically defined by a set of rules that govern their form and properties. They can be dynamically generated with mathematic equations and logical rules that undergo recursive iterations, designed as computer algorithms [3]. Therefore they are typically self-similar, identical and repeating at different iteration levels. Some stochastic procedures are usually involved to add variation and irregularity, avoiding absolute similarity and symmetry. Compared to Euclidean geometry, such a procedural based modelling method dramatically reduces storage requirements, and the resulting fractal can be realistic

for natural phenomena by using a relatively high number of iterations. The irregularity produced by fractals can be highly costly if generated by Euclidean geometry.

Fractal geometry has been widely used in many areas of art and science, such as in decorative imagery, architectural layout [4], theoretical construction of the universe in astronomical physics and organism simulation in biology (e.g. human nervous and blood supply systems) [1], [2]. All these complex natural phenomena can be truly presented with fractal mathematics and modelled through fractal computation.

Most existing urban theories prove that cities are fundamentally fractal in their developing form [5]–[8]. The architect's physical determinism concerning complex spatial phenomena in a city can be presented elaborately with fractals. A city planner's concern for the economic and social factors of location is entirely consistent with fractal means. Fractal principle provides an optimal method to simulate the physical form of natural growth of cities, as well as reflecting social and economic development.

Research has shown that rural environments, especially the landscape, can be generated effectively with fractals. It is the geometrical self-similarity and regular variation of rolling hills, mountain ranges, coastlines, slightly varying flat landscapes and water regions that allows them to be generated mathematically. However, there is a relative dearth of research on fractal modelling of cityscape. Cityscapes are often presented as roads, buildings, open spaces and block regions which are largely associated with various social factors, such as population, wealth and commercial and industrial development. The complexity of these factors is more notable for large-scale modern cities. It remains an open problem as to how the cityscape and social factors could be accomplished into one engineering framework reflecting the natural growth of a city.

To tackle this problem, we report our studies on automatic construction of 3D cityscape environment using fractals. The main aim of the project is to generate cityscape layout and land use property data of region blocks that can be interpreted by environment modellers to build 3D scenes.

We are particularly interested in the fractal simulation of multi-level hierarchical urban structure with regard to land use probability and the Central Place Theory, rather than fractal modelling of individual architectures or geometric landscape. Initial findings and results are presented, demonstrating how fractal methods can be best applied to cityscape modelling.

II. FRACTAL CITY: CENTRAL PLACE THEORY AND CENTRAL BUSINESS DISTRICT (CBD)

Central Place Theory [9], [10] describes how organisms group together and begin to form some kind of centre, around which sub-centres and sub-groups become orientated. This provides a theoretical concept and foundation of fractal city. Fractal effect is most obvious in urban development. Many towns and cities start from small collections of houses that grow through cycles of demands for resources and labour. A major concern of all gatherings of organisms is the need for some form of centre, in cities these are known as Central Business Districts (CBDs) [9]. It is from these centres that the surroundings are oriented and given structure. When a single CBD loses the ability to support its surrounding communities, sub-centres will begin to appear. This sub-centralization mechanism allows the surrounding communities greater access to resources and gives divisible structure to a larger body. Fractals can be used as constructor of sub-centres.

Despite the overall regularity of CBD-orientated city growth, irregularity brings natural features to a settlement. Probably similar to ant nests, since organisms made the first effort to settle down, there has been irregular distribution of localized life. Excluding particular man-made settlements that are planned from the outset, most settlements have irregular layouts. Naturally developed environments may grow progressively in unanticipated and undesirable ways, and perhaps by as little as a road or house at a time.

III. FRACTAL SUB-CENTRALIZATION

The hierarchical sub-centralization of city structure is typically in fractal form [5]. To simulate the organic growth of a city, the algorithm recursively creates the multi-scale sub-centres around the main CBD. In most modern cities, there are usually between 5 and 9 sub-centres around a parent CBD. Therefore the number N of sub-centres generated from its parent CBD is specified by

$$N = \text{round}\left(e^{-\frac{(\mu-\theta)^2}{2\sigma^2}}\right)$$

where the Gaussian mean $\mu=7$, variance $\sigma^2=2$, and the value is rounded to an integer. The distance of a sub-centre at l th level from its parent is calculated by

$$d_{l,n} = R/2^l \pm \text{random}(0, R/2^{l+1}), n = 1 \dots N$$

where R stands for the radius of the city. The directions of the sub-centres are equidistantly distributed in 360° , and altered perturbed randomly by

$$\theta_n = 360^\circ / N \pm \text{random}(0, 90^\circ / N).$$

The randomness and scalability are fractal properties of sub-centre generation, thus defining an irregular sub-centralisation orientated structure. The result of sub-centre generation is shown in Fig. 1. The points represent the location of each sub-centre in a 2D cityscape. The size of a sphere indicates its scale at different hierarchical levels, smaller CBDs are generated at higher iteration levels.

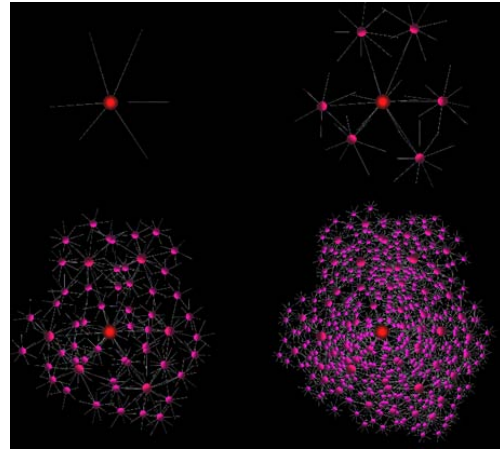


Fig. 1. Multi-scale hierarchical CBD generation $l=0,1,2,3$.

IV. ROAD GENERATION

Urban layout is usually heavily shaped by transport systems. After the sub-centralisation process, roads are built to give the physical structure of the city. We take into account various properties of modern road structures, such as ring roads, outbound roads and supplement roads needed to connect CBDs. The global appearance of the road network is determined by ring roads and outbound roads.

The road generation module is relatively complex and is the largest module in the system. It contains three object types: vertex, road and block. Instances of each type are stored in arrays and accessed through references to indices from other instances. Every vertex has a list of the connected roads; every road contains the vertices at each end; every block stores the indices of the roads and vertices of which the block is composed.

The first stage is to create a series of regular ring roads and outbound roads, as shown in Fig.2(a). Ring roads are generated around the centre of the city. The number of ring roads and their distances from the centre are defined based on the size of the city. A ring road is essentially a circle or ellipse, made of 64 vertices. Outbound roads head into the city centre from the outside. The number and orientation of such roads is determined according to the scale and population density of the city. The distribution of surrounding satellite cities should be also taken into account when the information is available.

The second stage is to add irregularity to the regular roads by randomly displacing each vertex. A 2D mid-point displacement map is generated using the method introduced in Section 6. Each vertex on the ring roads is perturbed according to the displacement map. Straight connecting lines of outbound roads are recursively subdivided and perturbed by mid-point displacement until reasonable small size bending roads are obtained. The result is illustrated in Fig.2(b).

At the last stage, supplement roads should be generated which give the layout of movement and links from one region to another. We consider the method based on Branching Principle: a close leads to a street leads to a main road leads to a dual

carriage way leads to ... and so on. The recursive Koch or L-system [4] algorithm will be employed, where branches spawn its child branches at each iteration and it is algorithmically straightforward. The variety and irregularity is obviously due to the difference in context and purpose of a region. The design and interface of this unit is already present. Full implementation will be completed to add street details.

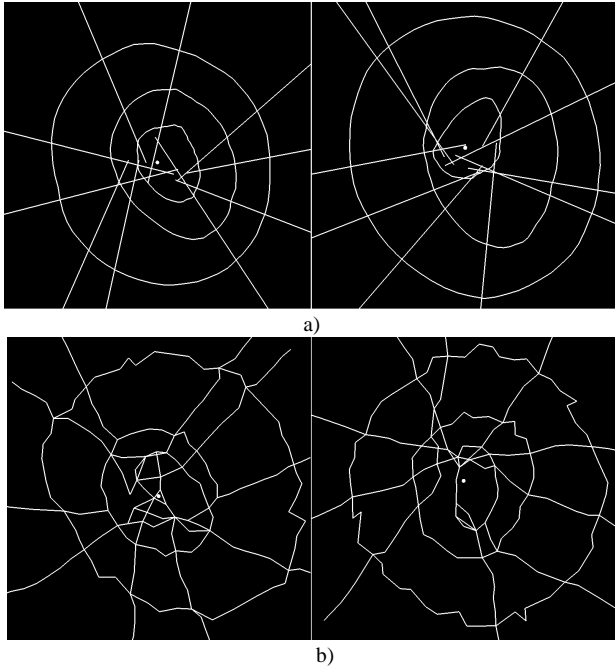


Fig. 2. Road generation: a) Regular ring roads and outbound roads, b) Mid-point perturbation.

V. CITY BLOCKS

The essential task in determining city blocks is to define polygons made from ring roads and outbound roads. Firstly, the program needs to calculate all intersections between outbound roads and ring roads, and then add new vertices to the vertex class. Subsequently, a recursive tracing function, as shown in Fig.3, is used to track vertices and edges to define polygons.

At the current tracked vertex:

- For every connecting edge of the vertex
 - 1) calculate the angle between this edge and the reference (white line in Fig. 3)
 - 2) store the edge reference in a list in ascending order of its angle
- Find the vertex at the other end of the edge with minimum angle
- If the vertex already visited, update list of vertices and edge indices, exit function; Otherwise trace into this vertex

a)

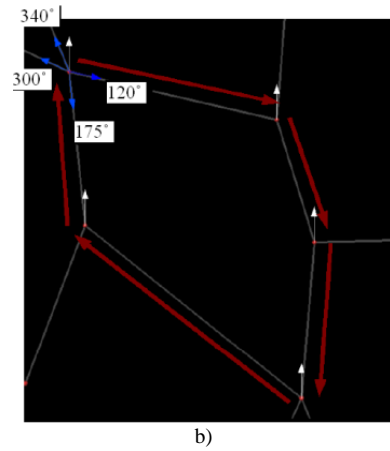
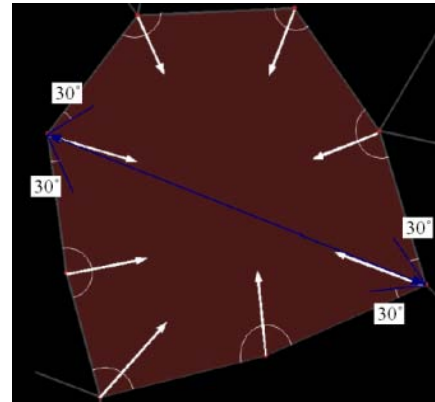


Fig. 3. Polygon tracing: a) tracing algorithm, b) tracing illustration.

Some polygons generated might be too big to give environmental features, such as polygons of residential areas. Since we intend to simulate cities on a similar scale to London (roughly 35 miles across), the desired size of a polygon should be similar to that of a school. Therefore large polygons must be broken down into smaller ones. Where splitting lines are created, they must reside within internal range of the polygon with a minimum 30° margin on each side, as shown in Fig.4(a). The middle point of a splitting edge is then perturbed and added to the vertex class.



a)



b)

Fig. 4. City blocks: a) polygon splitting, b) sub-divided polygon blocks

This process of polygon sub-division is recursively applied to all polygons until either a threshold of polygon size is achieved, or there are no more vertices to be split. A result with approximately around 600 vertices, 900 edges and 300 polygons is shown in Fig.4(b).

VI. LAND USE PROPERTY PROBABILITY

Buildings in different regions vary according to the land use properties of that particular region. For example, skyscrapers appear at centre of commercial area with high probability; whereas houses and schools are present mainly in residential regions. To create a diverse cityscape, land use properties of city blocks must be generated. In this project, we simulate several reprehensive land use properties: residential, commercial, industrial, governance, entertainment and open areas (e.g. parks and lakes).

To allow variation and smooth transfer, we propose that each property is modelled separately. In the simulation system, a probability map, scalable to match the city size in 2D, is used to represent the density distribution. It is generated by circular Gaussian distribution perturbed by a modified mid-point displacement procedure. The probability map can also be built using real-world data or by drawing images where available.

The recursive algorithm to generate the 2D mid-point displacement is shown in Fig.5. The calculation starts from a unit square, known as root element. An example result of mid-point displacement map is shown in Fig.6.

```

Set value of root element to 0
Set maximum variation  $\delta$ 
Set number of iteration  $I$ 
At  $i$ th iteration:
  For each parent element:
    1. Spit the element into  $2 \times 2$  array by mid-points
    2. Perturb the vertices of each child element with the scale of  $\delta/2^i$ 
    3. Reset value of each vertex using the average of its adjacent vertices
    4. If  $i=I$ 
       output results;
       otherwise go to iteration  $i+1$ 
    
```

Fig. 5. Algorithm of 2D mid-point displacement map.

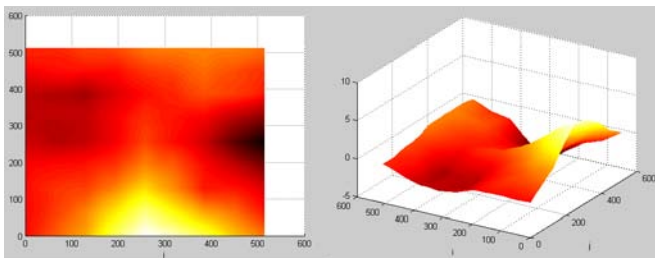


Fig. 6. Mid-point displacement map: height map (left), 3D view (right).

Since the city is presented in the form of CBD hierarchy, this suggests that probability of various land use properties could be modelled circularly according to CBD structure. For example, commercial probability was weighted towards the CBD centre; residential probability was weighted away from the centre; industrial land use or open spaces were generated toward the outskirts with a high probability value. These distributions are calculated at each sub-centre with decreasing distance and weight of influence. An example of population probability based on Gaussian is illustrated in Fig. 7. All probability distributions were perturbed through 9 iterations of mid-point displacement for a smooth variation.

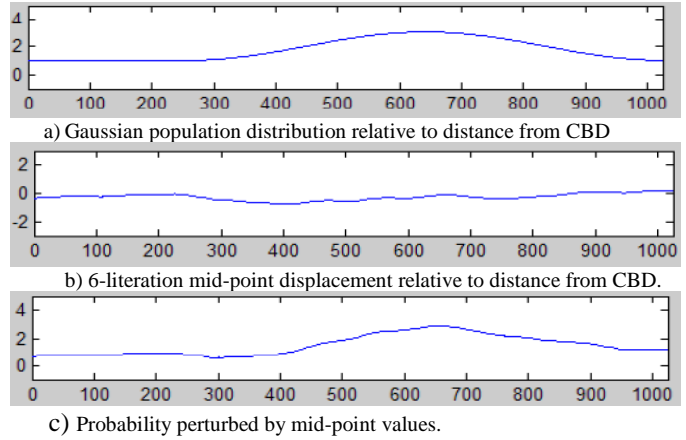


Fig. 7. Probability modelling.

Each probability map is normalized and scaled to match the city size. Taking probability values of all the maps, city blocks can be assigned with various economical and social functions.

VII. SYSTEM INTEGRATION

The system was implemented in C++ and OpenGL was used for graphic presentation. Individual components of the fractal city were integrated for free-flow dynamic modelling of the cityscape, as shown in Fig. 8. The sub-centralisation structure was generated first followed by the probability maps. Then the road system was built accordingly, and city block polygons were defined from ring roads and outbound roads. Probability map creation can be an offline module and then imported into the system where real-world data is used.

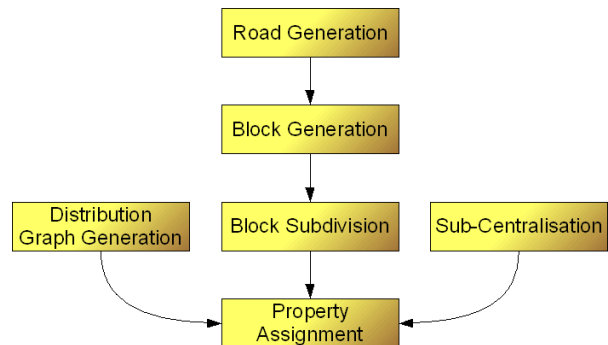
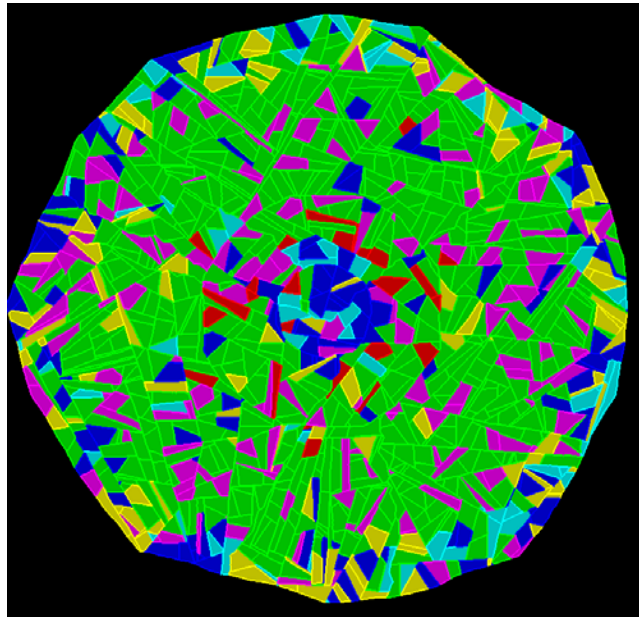
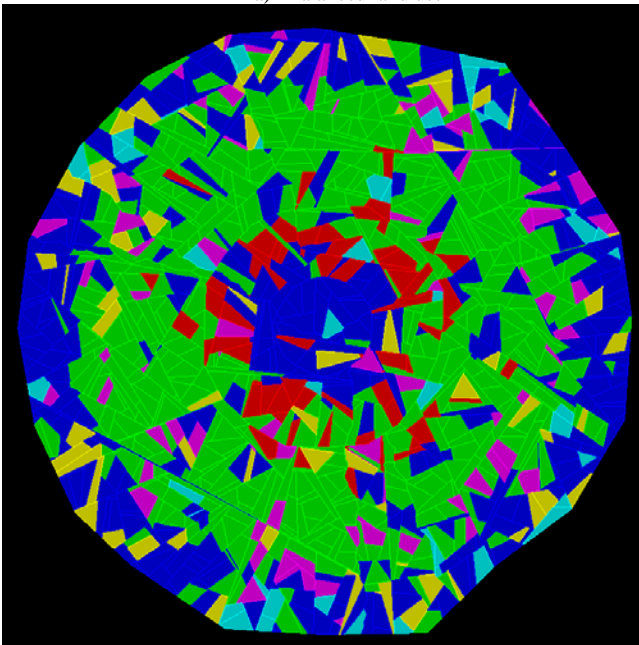


Fig. 8. Module Dependency Diagram.

Integrating the fractal city with various probability densities, city blocks were re-shaped. Land use properties and social functions of each city block were assigned based on the average probability in its polygon region by combining the probability maps with the sub-centralisation structure, thereby defining attributes of buildings and structures involved.



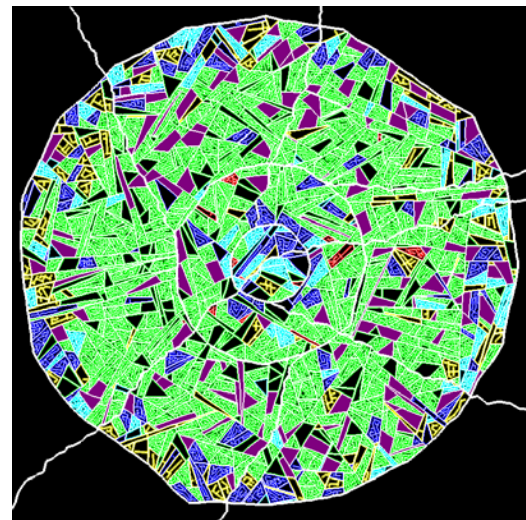
a) Balanced land use



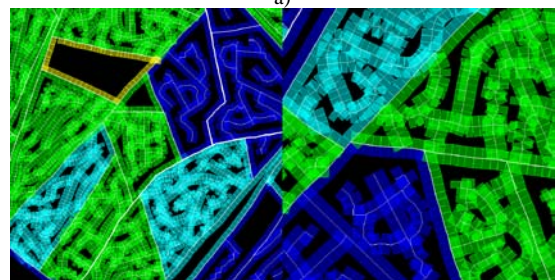
b) Land use with high commerce & governance

Fig. 9. Land use allocation. (Color Plate 7)

The visualised land use examples are shown in Fig. 9. Colours are used to present different land use properties of city blocks, e.g. green for residence, blue for commerce, yellow for industry, red for governance, magenta for open space and cyan for retail and entertainment.



a)



b)

Fig. 10. Fractal cityscape: a) functional city blocks, b) land plots. (Color Plate 8)

Land plots were created by generating polygons from the road structure and assigning their land use according to the block in which they are placed. Fig. 10 shows examples of land plots in the overall urban environment.

An interface of the software is shown in Fig. 11. It allows a friendly user environment where a number of parameters can be specified, such as road curvature, block size, land use probability, density, height and wealth features of buildings. Multi-threading was also incorporated to increase run-time efficiency.

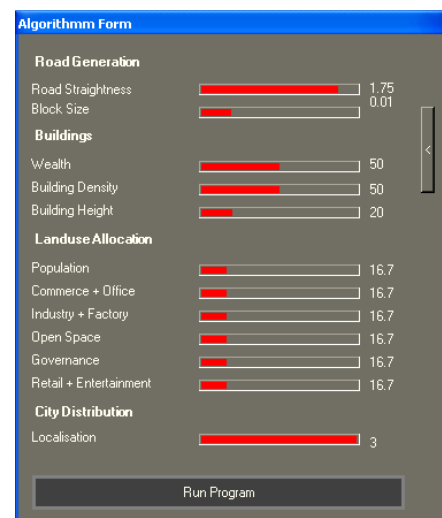


Fig. 11. Software interface.

VIII. CONCLUSIONS

The proposed system is able to automatically generate multi-level fractal urban scenery on a large scale. The Central Place Theory provides a theoretical foundation, while fractals provide the computational substrate of the simulation system. CBD orientated hierarchy is developed to simulate the natural growth of a city. Different types of roads are designed and weighted correspondingly. The probabilities of various important land uses are simulated and accomplished into the fractal framework. According to this transcription, 3D buildings with different attributes could be placed automatically into the environment.

This tentative study provides an automatic solution to the problem of cityscape construction. Implementation time has been largely reduced. Although probabilities were modelled theoretically based on hypothetical concepts, the proposed probability map allows use of real-world data or drawing data where available.

Some aspects have not been thoroughly addressed in the current design. City blocks are only assigned with overall land use properties. Detailed features, e.g. building density and aesthetic characteristics have not been fully defined. Interpretation of probability models is conducted at the final stage, which may sometimes produce some illogical results. Fusion methods of probability modelling should be studied in our future work. We will also work on standardization and optimization of the system so that it can be intergraded with game engines or modellers.

REFERENCES

- [1] M. Barnsley. Fractals everywhere, Morgan Kaufmann Pub., 1993.
- [2] B. Mandelbrot. The fractal geometry of nature, W. H. Freeman and Co., New York, 1982.
- [3] F. Kenneth. Fractal Geometry: mathematical foundations and applications, John Wiley & Sons, Ltd., 2003.
- [4] J. Marvie, J. Perret and K. Bouatouch, The FL-system: a functional L-system for procedural geometric modelling, *The Visual Computer*, 21(5), pp. 329-339, 2005.
- [5] M. Batty and P. Longley. Fractal cities: a geometry of form and function, Academic Press, London, 1994. www.fractalcities.org.
- [6] A. Lagarias, Fractal analysis of the urbanization at the outskirts of the city: models, measurement and explanation, *Cybergeo*, 2007.
- [7] C. Tannier, D. Pumain, Fractals in urban geography: a theoretical outline and an empirical example, *Cybergeo, Systems, Modeling, Geostatistics*, 2005.
- [8] A. Anas, R. Arnott and KA. Small. Urban spatial structure, *Journal of Economic Literature*, 36(3), pp. 1426-1464. 1998.
- [9] M. Berliant. Central place theory, Urban/Regional 0505001, EconWPA, 2005.
- [10] P. Haggett, AD Cliff and A. Frey. Locational analysis in human geography, Martin's Press, 1966.

character animation, 3D modelling, multimedia and virtual reality technology. Dr. Li is a member of BMVA. She has published over 30 research papers in a number of international journals and conferences, such as IEEE Trans. SMC, Pattern Recognition, Image & Vision computing, BMVC, ICPR, ICIP, CGI, F&G. <http://www2.docm.mmu.ac.uk/STAFF/B.Li>.



James Miller received a BSc in Computer Games Technology from Manchester Metropolitan University and has just completed the accredited Games Programming MSc from the University of Hull. His research interests include computer games and advanced computer graphics.



Baihua Li received the B.S. and M.S. degrees in Electronic Engineering from Tianjin University, Tianjin, China. She obtained the Ph.D. degree in Computer Science from the University of Wales, Aberystwyth, UK in 2003. She is a Senior Lecturer in School of Computing, Mathematics & Digital Technology, Manchester Metropolitan University, UK. Her research interests include video-based human motion analysis and behaviour understanding, human motion capture and