LOW-COST REAL-TIME SAR SIMULATION FOR APPLICATIONS IN MISSION PLANNING, EDUCATION AND INFORMATION EXTRACTION

Timo Balz

Institute for Photogrammetry (ifp), Universität Stuttgart, Germany
Geschwister-Scholl-Strasse 24D, D-70174 Stuttgart
timo.balz@ifp.uni-stuttgart.de

Commission VII, WG 2 & 7

KEY WORDS: SAR, Radar, Real-Time, Simulation

ABSTRACT

SAR simulators are important for a huge variety of applications. Realistic SAR simulations need realistic 3D models, which are often not available. Less realistic models can be used in the less accurate real-time simulation approach. Using modern graphic cards for SAR simulation even complex environments can be simulated in real-time. This is realised by implementing of SAR geometry and radiometry within standard graphics hardware, which offers 3D hardware acceleration and programmable graphics processing units (GPU). The real-time simulation capabilities are useful for mission planning, education and information extraction applications. For these applications less realistic simulation results based on less detailed models are feasible.

1. INTRODUCTION

SAR simulators are important tools for developing new SAR systems as well as for analyzing acquired SAR data. For many applications the time needed to calculate the simulation is not crucial, but for some others the simulation should be in real-time. Real-time SAR simulations are useful for interactive applications, like assisting the analysis of SAR data and creating simulated data for pattern recognition applications. Furthermore, applications like mission planning, education and training require fast calculation and interactivity. However, these applications accept less accurate simulation results.

Acquiring reliable ground truth data is time consuming. In most areas no 3D data is available at all, let alone high resolution 3D models containing surface properties. Very realistic SAR simulations depend on realistic and detailed models. If no such models are available, less accurate but fast simulators to prefer.

Using the more and more powerful and flexible graphics processing units (GPU), it is possible to visualise SAR effects in real time. For mission planning as an example, real-time visualization is crucial. With faster visualisations, many different sensor properties and flight paths can be tested to evaluate the optimal parameters, before flights actually take place. This reduces unnecessary and expensive errors during data acquisition. In training and education users should be able to interactively change simulation parameters and see the results in real time, improving the learning results.

Although GPUs can be used for visualizing radar images, some adjustments are necessary. For instance, the geometry of SAR images is different, because in range directions the geometry is based on the running time of the signal. In azimuth the geometry is based on the Doppler Effect. The radiometry is also different due to the different wavelengths of the signal. Different approaches to computer graphics used in a variety of computer games are adapted to realise visualisation of SAR data in real time. These approaches, however, must be modified in order to use them for SAR geometry and radiometry. By further modifying these approaches, complex scenes, bi-static missions, or effects caused by object movements can be visualised.

Acquiring good ground-truth data for a simulation is difficult, especially in less developed countries. This will be discussed in section 2. Low-cost hardware and software solutions as well as general-purpose computation using graphics hardware will be explained briefly in section 3. Examples of real-time simulation applications are presented in section 4. In section 5 the real-time SAR simulator SARViz (Balz, 2006) will be described.

2. ACQUIRING GROUND TRUTH DATA FOR SAR SIMULATIONS

For every simulation of real world objects, these objects have to be modelled. SAR simulating a city requires a 3D city model or a DSM of the city area. A realistic simulation needs detailed models and further information about the material properties. For most areas this information is not available. Although many 3D city models exist, only a few fully textured high-resolution models are available. Normally no city model includes material information, especially information about the material reflection properties in the microwave spectrum. Beside some test areas, there is no ground truth data for realistic SAR simulation available. Because of this, realistic simulations of real cities are often impossible.

Assuming the availability of a very sophisticated simulator, detailed information about the simulated model and its material properties are needed to achieve very realistic simulation results. This information is normally not available. Simulating a model with only limited information about the reflection properties of its surface, results in less accurate simulation results. Even a very realistic simulator is not generating realistic results if the models and their material properties are not sufficient. Realistic simulations depend on realistic models, which are not widely available.

The situation in lesser developed and/or fast developing areas is even worse. 3D city models are often not available or outdated. In fast growing urban areas, maps are outdated as soon as they are printed. Reliable 3D information is normally not available at all.

The realistic simulation of high resolution SAR images requires very detailed 3D models. These high resolution 3D models can for example be acquired using terrestrial 3D laser scanning. Triangulating 3D point clouds and closing existing gaps are time consuming tasks. After the triangulation, the model does still not contain any surface information. Texturing the model is laborious. For SAR simulation the material reflection properties are important. Creating even one high-resolution model for a realistic SAR simulation is very time consuming.

The building models inside the 3D city model of Stuttgart (see Figure 1), provided by the City Surveying Office of Stuttgart, were photogrammetrically reconstructed in a semi-automatic process. The wireframe city model contains the geometry of 36,000 buildings covering an area of 25 km², meaning that almost every building of the city and its suburbs is included. The overall complexity of all the building models amounts to 1.5 million triangles. Around 1,000 of these buildings have been textured in about 30 man-months. Still this dataset is not sufficient for realistic high resolution SAR simulations, because the material properties are still missing. To include those, the approximately 8,000 ground based close-up photographs of the building façades would have to be classified.

Figure 1. 3D real-time visualisation of the Stuttgart city model showing over 36,000 buildings (Kada et al, 2003)

Even if all of the data is available, a ray-tracing simulation containing millions of triangles, thousands of textures and multiple reflections would be very time consuming. A realistic high-resolution SAR simulation of complete city models is therefore not accomplishable.

Less accurate but fast simulators are more useable in real-world applications. These simulators should deliver fast results, preferable in real-time. The hardware requirements should be low and affordable, especially if the simulator is to be used in lesser developed countries.

3. LOW-COST HARDWARE AND SOFTWARE SOLUTIONS FOR SAR SIMULATION

GPUs of the latest generation have more computational power as standard CPUs. Nowadays they provide almost supercomputing power at low costs. Due to their comparably cheap prices, programmable graphics cards are especially lucrative for scientific applications. Massive parallel graphic processing units are particularly suitable for the calculation of intense applications and are useable for a variety of applications, due to their flexible programmability.

The computational power of GPUs can be used for a variety of general computation purposes, like for example linear algebra, signal, image and audio processing as well as data mining (Owens et al, 2007). A GPU design differs from a CPU design. As depicted in Figure 3, a big part of the CPU is used for branching, whereas most transistors on the GPU are used for calculations (Owens, 2005). A GPU is a data-parallel streaming processor working in a single-instruction, multiple data (SIMD) fashion and allows massively parallel computing.

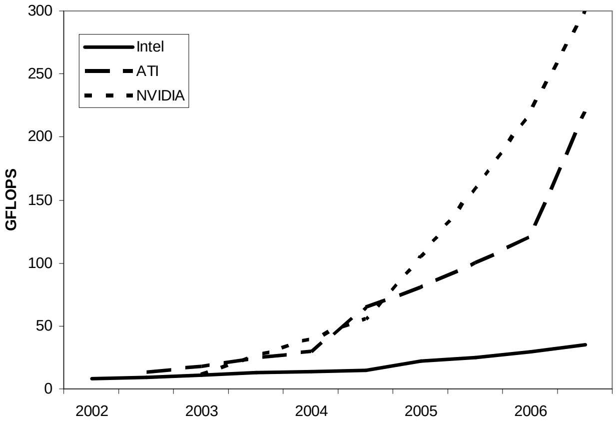

Figure 2. Comparison between CPU and GPU (see Buck, 2004; Owens et al, 2007)

GPUs are specialised for visualisation applications. SAR simulation is a visualisation application. Therefore GPUs are well suited to calculate SAR simulations. But radar images differ in many ways from images acquired by passive sensor systems. The imaging geometry of a SAR system is different in azimuth and range directions. In range direction the placement of objects depends on the slant-range distance between objects and sensor. In azimuth, the geometry is based on the Doppler Effect.

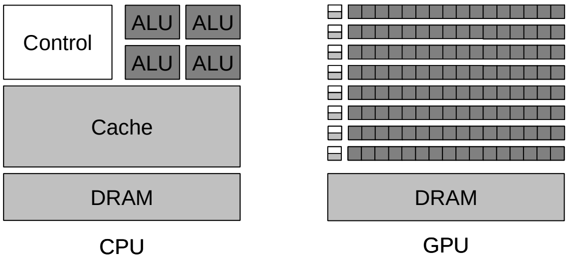

Figure 3. A GPU uses comparably more transistors as arithmetic logical unit (NVIDIA, 2007)

GPU based simulators adapt the geometry in the vertex stage of the graphics card using so called vertex shaders. The dynamically adapted geometry is piped into the fragment stage where pixel shaders are used to calculate the radiometry. The radiometric ambiguities, as well as speckling, have to be considered. Because of the programmability of these shaders, different models for backscattering electromagnetic waves can be implemented (see Figure 4).

Using GPGPU applications on game devices is a rather new approach. This is only practicable due to the powerful processors and the flexible programmability of the newest generation of gaming devices. For example Sony's Playstation 3 can be used to assist the solution of scientific problems. Stanford University is using distributed computing for calculating protein folding processes. 2,000,000 CPUs are combined for this approach. The distributed module for the PS3 is installed on thousands of PS3 achieving 270 TFLOP/s of the overall 570 TFLOP/s of the project (Stanford University, 2007). This distributed "super computer" is build using gaming devices.

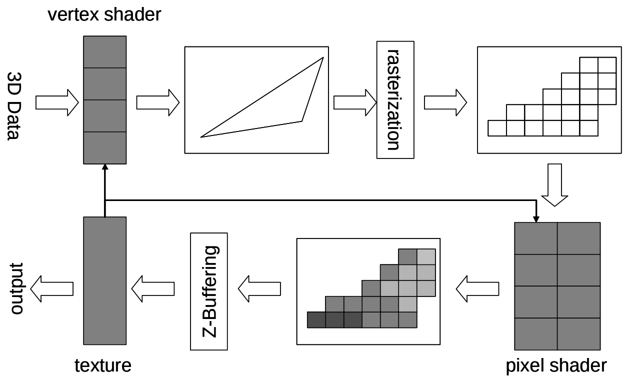

Figure 4. Programmable graphics pipeline of modern computer graphics cards

Microsoft's Xbox 360 is nowadays programmable by everybody. Using the XNA framework (Microsoft, 2006), the Xbox 360 can be used as scientific visualisation device. The XNA framework is available for free and can be used to develop even complex 3D applications. At the moment the use of the Xbox 360 with XNA is still limited for scientific applications, due to licensing issues and the missing network support. The Xbox could be used as a SAR simulator, especially for training and education purposes. It is a standard device and it is easy accessible and programmable using the XNA framework. The cheap hardware cost and the straightforward usability of the PS3 or the Xbox 360 are ideal for applications in education and training.

4. REAL-TIME SAR SIMULATION APPLICATIONS

Although most scientific SAR applications do not need real-time simulation capabilities, some application depend on them. SAR mission planning as well as education and SAR image interpretation benefit or even depend on fast simulation results.

4.1 SAR mission planning

Mission planning is of the utmost importance for successful remote sensing applications. Analyzing occlusions and ambiguities affecting an area of interest is especially important in urban areas. Because the percentage of humans living in cities is already above 50% (UN-HABITAT, 2006), remote sensing in urban areas is very important. Many of the fastest growing agglomerations are in areas endangered by flooding due to global change or disasters.

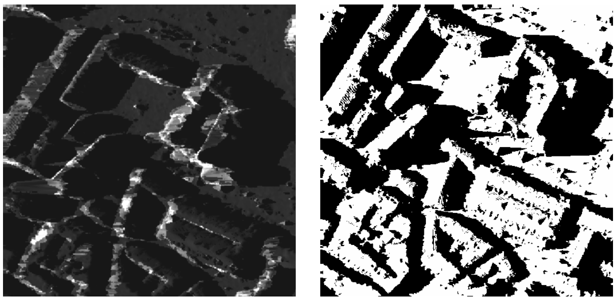

Figure 5. Simulation of a DSM (left) and analysis of occlusions (right)

Disaster management in urban areas needs fast and reliable sources of information. Remote sensing is very important. SAR can be optimally used in disaster management, because of its all-weather capability. This is especially important in disaster monitoring, because many disasters occur due to bad weather conditions. Furthermore most of the fast growing agglomerations are in cloud-prone or rainy areas, hindering time-critical data acquisition using VHR sensor systems.

In urban areas a simulation based occlusion analysis can be done using a DSM (see Figure 5). Using a real-time simulation, every possible combination of azimuth and off-nadir angle can be analysed in less than 15 minutes using just brute-force techniques.

In many areas of the world the required DSM or 3D city models are not available. A high-resolution simulation based occlusion analysis is not possible in these areas. Still the simulation can assist the mission planning using the world wide available SRTM data. The example in Figure 6 demonstrates the possibility of this data to determine occlusion affected areas for flood monitoring.

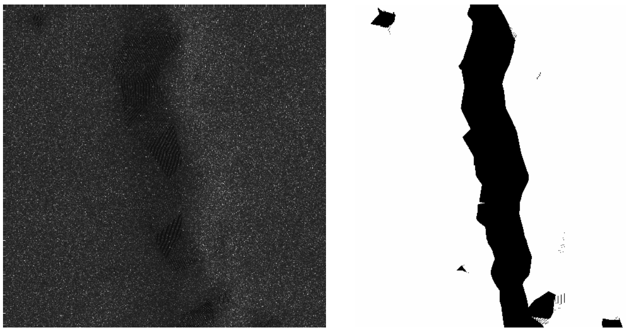

Figure 6. SAR simulated SRTM data showing a valley in Germany (left) and the respective shadow analysis (right)

4.2 SAR training and education

The increasing amount of available SAR data requires education and training of new users. Analysing SAR images is difficult and requires a deep understanding of the SAR theory. SAR simulators are an important tool in training and education. They can provide a variety of simulated images showing defined scenes.

Real-time simulation is even more advantageous. Users can directly interact with the simulator. The expected learning curve is steeper and the results are more lasting.

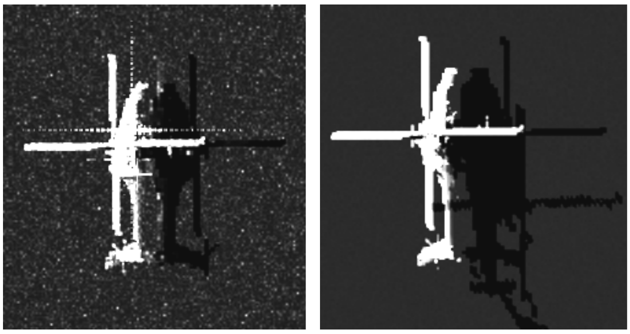

Figure 7. SAR Visualisation of a helicopter model using double-bounce effects (left), single-bounce bi-static but without side-lobes and speckling (right)

As illustrated in Figure 7, the SAR effects can be visualised separately or combined. This guarantees a deeper understanding of each single SAR effect. Because of the usability and interactivity, real-time SAR simulation is the optimal choice for SAR training and education.

4.3 Simulation assisted SAR image interpretation

Analysing SAR data requires knowledge about the application the analysis is used for as well as SAR knowledge. For disaster monitoring, knowledge about the affected area and the disaster characteristics are important. The SAR image interpretation can be assisted by SAR simulations of the area. Some SAR effects can be visualised to support the less experienced SAR user.

Not every user of SAR images has s deep understanding of the underlying SAR principles. SAR simulators can assist the education of these users, but it can also assist the interpretation of SAR images. To assist the analysis, a real-time SAR simulation is most useful. If the simulation takes a long time to process, it does not support the user. The user should be able to interactively change the simulated environment, therefore fast and interactive tools are necessary.

5. SARVIZ: A REAL-TIME SAR SIMULATOR

The real-time SAR effect visualisation tool SARViz has been presented the first time in 2006 (Balz, 2006). SARViz is using methods developed by computer graphics to simulate SAR images. The GPU is processing triangles using local illumination. Each triangle is visualised independent from any other triangle. Each triangle point is processed by the so-called vertex shader, which treats the geometry. After the rasterization, the radiometry of each pixel is calculated by the so-called pixel or fragment shader.

Vertex shaders are highly specialized parts of a graphics card and are optimized for matrix calculations. Each point is transformed from the model coordinate system to world coordinates and then subsequently to screen or image coordinates. The distance geometry of a SAR image needs to be implemented in the vertex shader. The range position of each object in a SAR image depends on the distance between the object and the sensor, thus higher points, i.e. points with larger z-values, are closer to the sensor and are therefore mapped closer to near-range. This results in a shift in range direction Δx depending on the height above the ground level z and the incidence angle σ :

$$\Delta x = z \cdot \tan(\sigma)$$

Afterwards the pixel shader is processing these pixels to compute the corresponding radiometry. Calculating the reflection intensity of SAR images is complex. Modern pixel shaders are powerful and flexible enough even for complex tasks like SAR simulation.

For each pixel the corresponding face normal is determined using the 3D model. Taking material properties, like the dielectric constant, and sensor properties into account, reflection strengths can be calculated in the pixel shader. SARViz offers three different methods of backscattering computation. The statistical method based on measurements of Ulaby & Dobson (1989), a direct calculation based on the roughness and dielectric constant of the material developed by Zribi (2006) and an adaptation of computer graphics methods. Most commonly used is the adaptation of the computer graphics methods, due to its computing time efficiency.

According to the Phong reflection model (Phong, 1975), three illumination elements (diffuse, specular and ambient) are combined for the resulting reflection. Ambient lightning is negligible for SAR simulation. The diffuse illumination element is equivalent to the Lambertian reflection element. Combining the diffuse and specular elements, the SAR reflection intensity can be calculated. The reflection is calculated locally, therefore multi-reflections are not supported. The speckle is visualised as additive noise.

5.1 Speckling and multi-look

Speckling is produced by mutual interference of coherent waves that are subject to phase differences. For simplicity it can be visualised as additive noise. The speckling can be described as Rayleigh distributed.

The reflection value of a multi-look image is a combination of M single SAR images. The resulting speckle value in a multi-look image s_M is calculated using:

$$s_M = \frac{1}{M}\sum_{i=1}^{M} s_i$$

Visualising multi-look images by changing the speckle value is a simple approach, which is acceptable for most applications. The separate simulation of each look is more realistic. A multi-look image is the combination of sub-aperture images. Each sub-aperture image has a different squint angle. Although the differences between the squint angles are small, edges, layovers and shadows appear blurred in the combined multi-look image (see Figure 8).

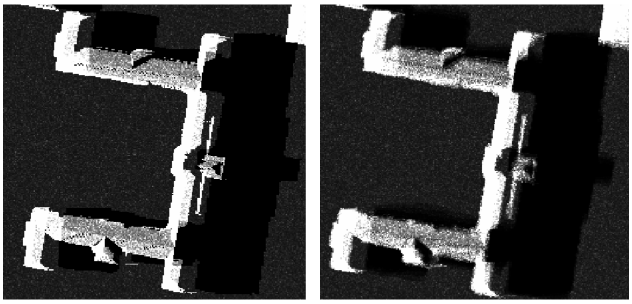

Figure 8. Simple (left) and separately simulated (right) multi-look visualisation of the “Neues Schloss” in Stuttgart

5.2 Visualisation of moving objects

The visualisation of moving objects in SAR images depends on the direction of the movement. Objects moving in range direction will be shifted in azimuth direction, because the azimuth position of an object in a SAR image is determined by the Doppler Effect. Moving objects disturb the Doppler. Therefore the object position in azimuth is erroneous. Objects movements in azimuth direction lead to smearing in the SAR image. If an object is moving in azimuth and range direction, the effects are combined (see Figure 9).

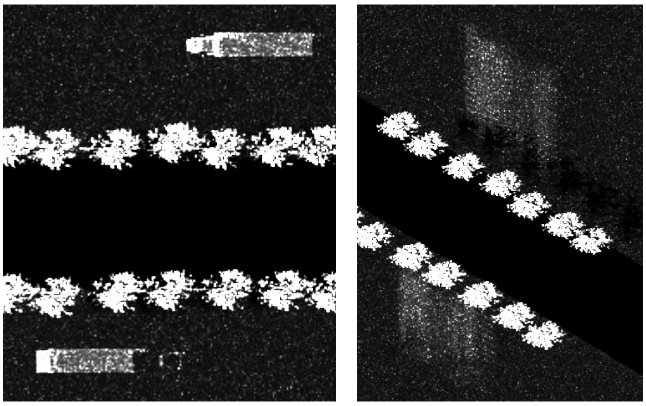

Figure 9. Objects moving in range direction (left) and objects moving in azimuth and range direction (right)

5.3 Limited multi-reflection support

Real-time visualisation used on graphics cards does have some limits. Current GPUs are using rasterization. Unlike in ray tracing, the paths of the rays are unknown. Therefore, multiple reflections cannot be visualised. However, these limitations are not very important in most applications. For instance, mission planning does not require visualization of multi-bouncing.

In computer graphics the so called environmental mapping is used for the real-time visualisation of specular reflections. Most common is the cubic environment mapping (Greene, 1986). In this method a cube texture is created around the reflecting object (see Figure 10). The cube contains the scene around the object for visualising the reflections. To generate the cube texture, the scene is rendered six times without the reflecting object, but from the position of the object and looking to different directions. The six sides of the cubes are filled with textures containing the results from this rendering step. Based on the environment stored in the cube texture, reflections from the surrounding area can be visualised.

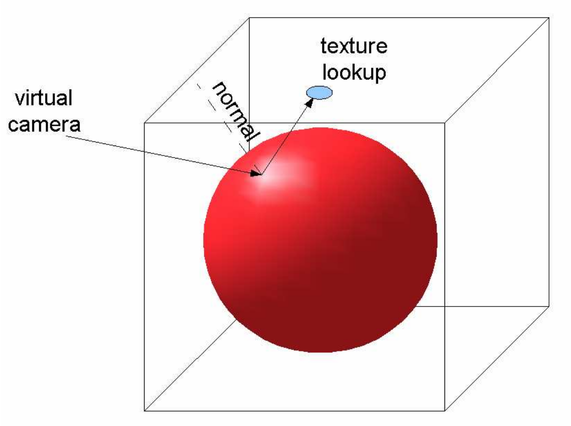

Figure 10. Cube mapping

For SAR simulation, this technique is also applicable, but only double-bounce reflections can be visualised that way. Even worse, only double-bounce reflections between different objects can be visualised. Double-bounce reflections from the ground, for example caused by asphalt surfaces, can be simulated using this technique. For this purpose, not even a real cube has to be simulated, because just one cube side is needed for the environment map, since only the asphalt surface is assumed to reflect.

6. CONCLUSION

SAR simulations are important tools for radar remote sensing. Simulators can assist mission planning as well as the analysis of acquired radar data. A meaningful simulation requires 3D models. These models have to represent the true geometry as well as realistic material properties. Acquiring such models is time-consuming and expensive. Therefore, in most areas of the world no such models are available. Less detailed models still can be used for simulations, but the results will not be that realistic. Using such models, a less accurate but fast simulator can be used, because the achievable level of realism is limited by the models not the simulator.

Simulating SAR images in real-time offers a huge variety of new possibilities to the users. Interactivity improves mission planning tools, as well as analysis of SAR data. Moreover, fast and interactive visualisations offers many training and education benefits. Compared with the possibilities from the real-time visualisation capability, the absence of multi-bouncing is only a minor drawback. SAR simulations can benefit from the tremendous development in computer graphics. Using low-cost hardware and software solutions, SAR simulations in real-time become affordable for many users. Due to ongoing hardware developments, using even less expensive hardware, even more realistic future results are expected.

REFERENCE

Balz, T., 2006. Real Time SAR-Simulation on Graphics Processing Units. In: *Proceedings of the 6th European Conference on Synthetic Aperture Radar (EUSAR 2006)*, Dresden.

Buck, I., 2004. GPU computation strategies and tips. In: *Proceedings of the conference on SIGGRAPH 2004 course notes GRAPH '04*, Los Angeles.

Greene, N., 1986. Environment Mapping and Other Applications of World Projections. In: *IEEE Computer Graphics and Applications*, 6 (11), pp. 21-29.

Kada, M., Roettger, S., Weiss, K., Ertl, T., Fritsch, D., 2003. Real-Time Visualisation of Urban Landscapes Using Open-Source Software. In: *Proceedings of ACRS 2003 ISRS*, Busan, Korea.

Microsoft, 2006. XNA. In: *http://msdn.microsoft.com/xna*. As at 10/4/2007

NVIDIA, 2007. CUDA Programming Guide Version 0.8. In: *http://developer.nvidia.com*. As at 2/16/2007

Owens, J.D., 2005. Streaming Architectures and Technology Trends. In: Pharr, M.: *GPU Gems 2. Programming Techniques for High-Performance Graphics and General-Purpose Computation*. Addison-Wesley, Boston, pp. 457-470.

Owens, J.D., Luebke, D., Govindaraju, N., Harris, M., Krüger, J., Lefohn, A.E., Purcell, T.J., 2007. A Survey of General-Purpose Computation on Graphics Hardware. In: *Computer Graphics Forum*, 26 (1), pp. 80-113.

Phong, B.T., 1975. Illumination for computer generated pictures. In: *Communications of the ACM*, 18 (6), pp. 311-317.

Stanford University, 2007. Folding@Home Distributed Computing. In: *http://folding.stanford.edu/*. As at 5/4/2007

Ulaby, F.T., Dobson, M.C., 1989. *Handbook of Radar Scattering Statistics for Terrain*. Artech House, Norwood, Massachusetts.

UN-HABITAT, 2006. *State of the Worlds Cities 2006/07: The Millenium Goals and Urban Sustainability*. UN-HABITAT, Nairobi, Kenya.

Zribi, M., Baghdadi, N., Guérin, C., 2006. A new Semi-empirical model for the analysis of surface roughness heterogenity. In: *Proceedings of the 2006 IEEE International Geoscience and Remote Sensing Symposium and 27th Canadian Symposium on Remote Sensing (IGARSS 2006)*, Denver, Colorado.