Synthetic aperture imaging radar signal simulation of urban environments

Thomas Stenson Moffat
11501

MEng interim individual project report

Academic Year 2020-2021
Dr Robert Watson
Wordcount: 3127

Part 1

Project Plan

1.1 Scope and Aims

This project is to create a simulator that can show the transformation between a 3 dimensional model and a 2 dimensional image using synthetic aperture imaging radar mostly for use as a demonstration tool. This is a method of radar that generally uses the movement of the platform that the radar transceiver is mounted on to build up an image based on the range that the radar signal travels and the reflection characteristics of the surface it is reflected off of. This form of radar imaging has a higher resolution than a conventional radar transceiver as the aperture size is based on how far the platform travels during the time it is imaging a target, and so the aperture is far larger than could be physically constructed.

The main aim of this project is to create a very basic simulator that can take a cube as an input and output an accurate SAR image. This will require a few steps to achieve. The first will be to create a scene in the chosen programming platform, which as of writing this document is MATLAB and Simulink, created by MathWorks. The radar pulse signal will need to be simulated as well as effects of multipath reflection, and finally the response received by the simulated radar will need to be transformed to an image by use of backscattering or an Omega-K algorithm. A secondary aim is to simulate the inclusion of speckle within the image, which is a SAR image artefact created by the sum of contributions from a large number of scatterers, thereby increasing the realism of the output image.

More advanced aims of this project are to be able to simulate the more advanced scenes such as actual urban environments and to simulate the multipath reflection effects of the radar signal in this area. This will end up requiring a more efficient implementation so the final major aim is to implement either a ray tracing or rasterisation approach, which are common methods of transforming a 3 dimensional space to a 2 dimensional image, most commonly used in animation and video games. While in other SAR simulation implementations efficiency is something that is considered, this is not necessarily something that is within the scope of this project. Improvements in efficiency are desirable up to a point however accuracy and functionality are more important. This is especially true as some of the literature (as covered later) focuses on a real-time implementation which is desirable for some applications, especially in practical applications such as determining SAR platform flight paths. It is not so applicable here in a demonstration application.

1.2 Project Plan

1.2.1 Workplan

A series of steps have been defined to help meet the overall aims as defined previously. The exit criteria for each of these packages are major milestones and will be defined appropriately.

Step 1 - Basic 3D Modelling

The purpose of this step is to create a 3 dimensional model of a cube using a modelling package. This step will have more time built in than is required as it also includes some research on the best technology to use, above the research that has already been performed. This step will also require some time to learn the tool chosen. The exit criteria for this step is being able to display a 3 dimensional model on screen and view it from different directions using the tools provided by the software package.

Step 2 - Antenna Beam Simulating

This step will overlap somewhat with step 1 as they can be performed in two separate environments. A basic definition of the platform that the radar is mounted on will be created as well as a basic definition of the characteristics of the radar's beam with the characteristics of the propagation medium. The exit criteria for this step of the project is to have a model of the environmental parameters and the radar parameters.

Step 3 - First Order Contributions

Having created the 3 dimensional model, this step will have the purpose of defining the reflection characteristics of the environment. This step is only to simulate the first reflections off of the target structure or the ground, and not the results from the radar signal reflecting off of the wall and then the ground or vice versa. This only really requires the completion of step 1, as all reflections can be tested using an emitter at a single point and does not need to have accurate responses, only to prove that the radar signals will reflect off of the model in a coherent and predictable way. The exit criteria for this is to have a reflection based on the dielectric constant of the model building and a different dielectric of the model ground, to be defined and hard coded into the program for different materials.

Step 4 - Create Output Image

This is the first step that will actually produce a visible result, using a matched filtering algorithm yet to be determined but most likely time-domain Backprojection, due to its flexibility. The technique uses the difference between what the radar is expecting and what it actually receives in order to construct an image. The exit criteria for this step is to be able to produce an image based on the raw SAR data that is generated from the simulated platform.

Step 5 - Simulate Second Order Contributions

This is the final necessary step for completion of this project, and is an extension to step 3. The purpose of this step is to add in the effects of the antenna beam reflecting off both the ground and the wall, in either permutation, before returning to the receiver. This will complicate the final image somewhat but also more accurately simulate its effects.

The exit criteria for this step is to include the effects of second order contributions in the output image.

Step 6 - Complicate Scene and Increase Order Contributions ***(Optional)***

This step will involve increasing the number of structures in the simulated scene as well as simulating third order contributions and above. This is when the signal reflects off of objects three or more times before returning to the receiver. The contributions from this are exceedingly small however it is still worth implementing as it will contribute something. The scene complications will also involve a more accurate dielectric modelling, with an attempt at windows and other non-stone aspects of a building. The final aspect of this is to be able to extract a scene from a mapping tool such as OpenStreetMap or Google Earth and import it into the program before using that to generate an image. The exit criteria for this step is to produce an image of a more complicated scene and be able to import a section of a Google Earth or OpenStreetMaps 3D model as the complexity of this will likely require the next step to image correctly.

Step 7 - Implement Hybrid Method ***(Optional)***

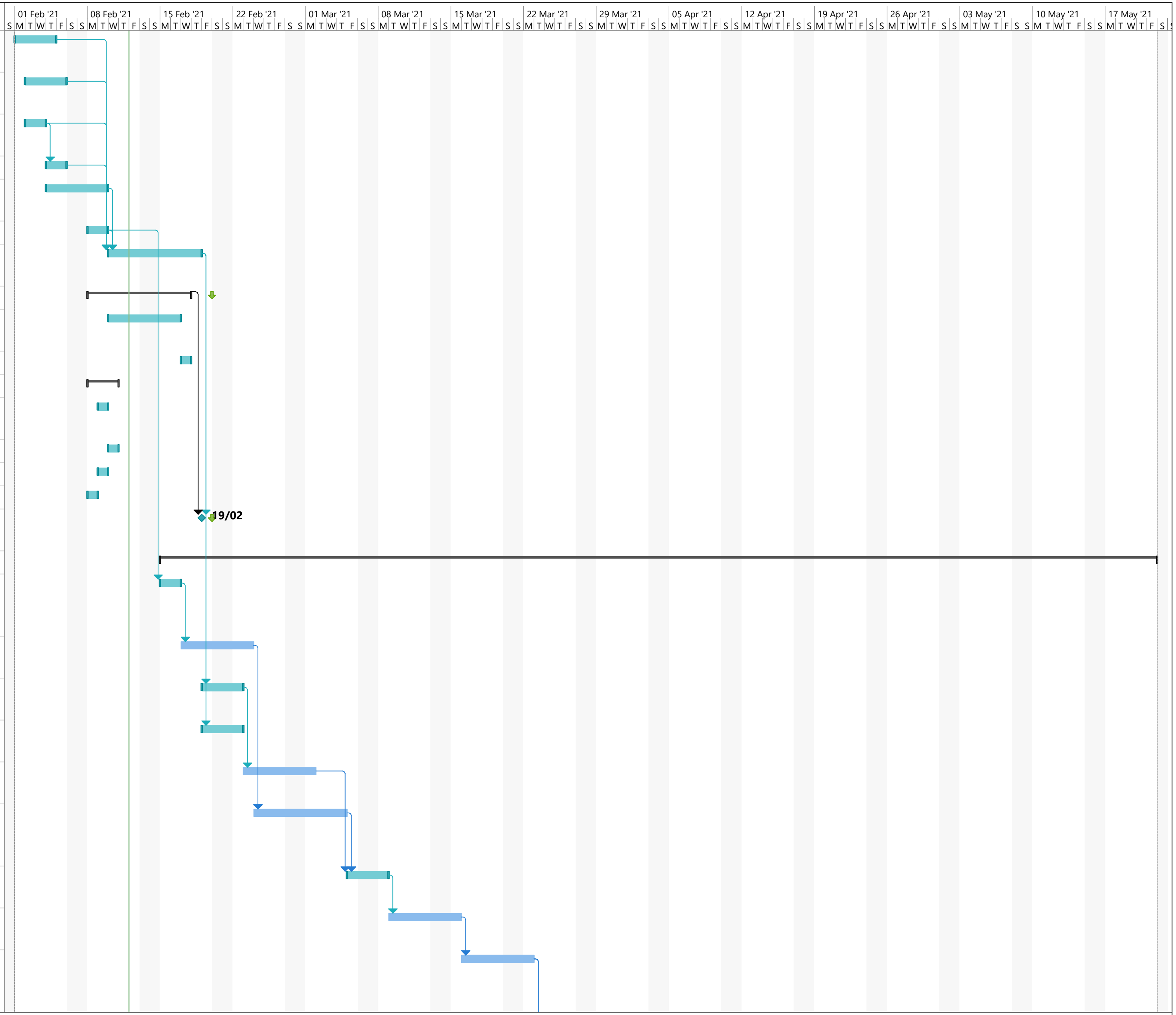
This step is to use a secondary method to reduce the computational complexity of creating the raw SAR image. There are a few approaches to this that have been investigated in the literature and are explored in more depth in part B. The actual method used is currently undecided as is the approach taken towards its implementation however the goal is to improve the efficiency of the simulator while maintaining a high level of accuracy in the output SAR image. The exit criteria for this milestone is to have an implementation of a hybrid method that improves the calculation efficiency.

Step 8 - Extra Time

This step is a catch-all step if other steps take longer than was previously expected or to implement other features that are decided upon during the implementation of previous steps. This is to avoid feature creep during the process as unless it is absolutely necessary it can be deferred to this step. This step has no exit criteria, however will result in the program being finished and after this stage no more features will be added.

1.3 Gantt Chart

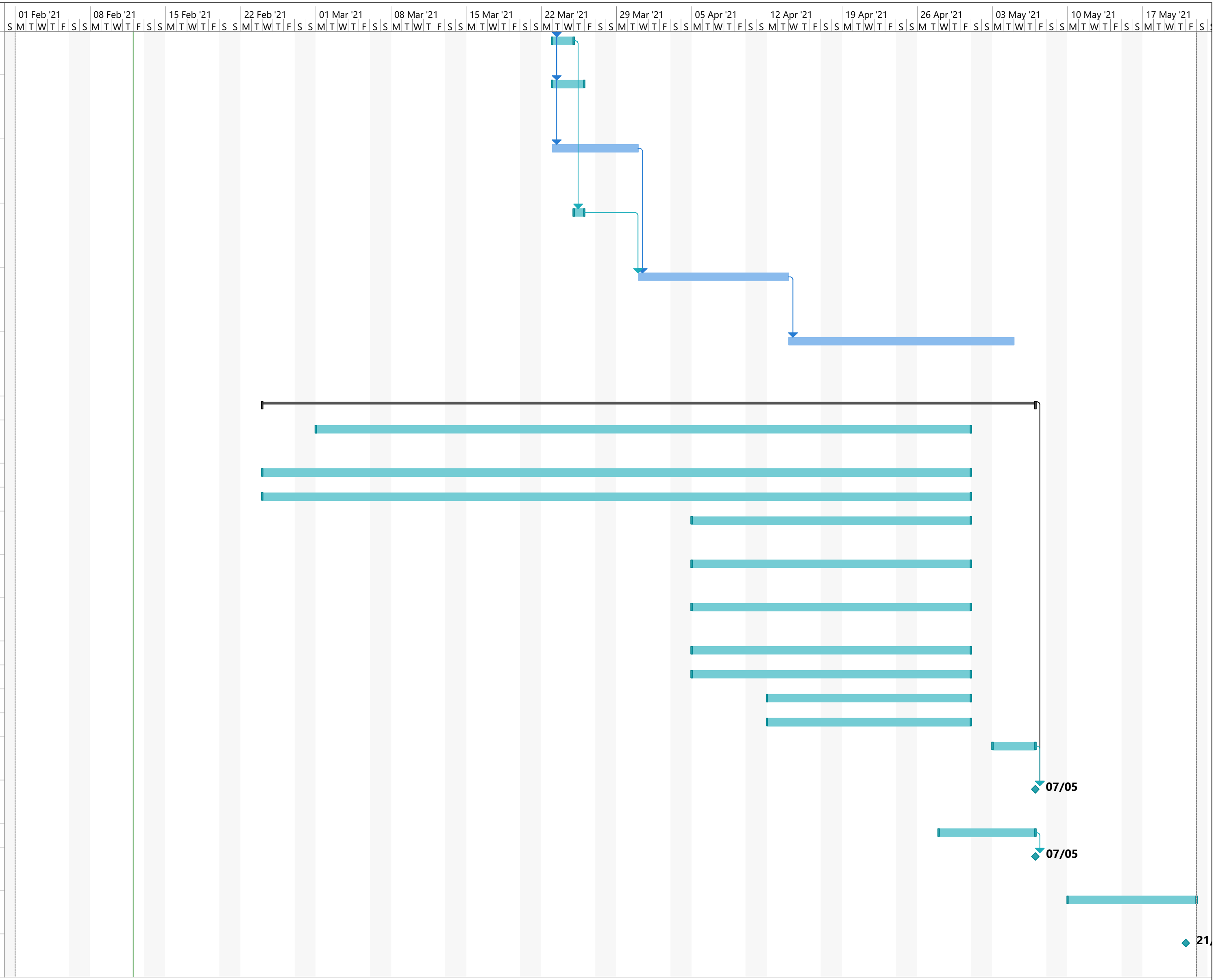
ID	Task Name	Duration	Start	Finish	Predecessors
1	Research Previous Appproaches	4 days	Mon 01/02/21	Thu 04/02/21	
2	Research Possible Solutions	4 days	Tue 02/02/21	Fri 05/02/21	
3	Decide on tasks and steps	2 days	Tue 02/02/21	Wed 03/02/21	
4	Create Gantt Chart	2 days	Thu 04/02/21	Fri 05/02/21	3
5	Define Scope and Aims	4 days	Thu 04/02/21	Tue 09/02/21	
6	Define Project Plan	2 days	Mon 08/02/2	Tue 09/02/21	
7	Write Interim Report	7 days	Wed 10/02/21	Thu 18/02/21	1,2,3,4,5,6
8	**Interim Report**	**8 days**	**Mon 08/02/2**	**Wed 17/02/2**	
9	Project Background	5 days	Wed 10/02/21	Tue 16/02/21	
10	Title Page	1 day	Wed 17/02/2	Wed 17/02/2	
11	**Project Plan**	**3 days**	**Mon 08/02/2**	**Wed 10/02/2**	
12	Scope and Aims	1 day	Tue 09/02/21	Tue 09/02/21	
13	Project Plan	1 day	Wed 10/02/2	Wed 10/02/2	
14	Gantt Chart	1 day	Tue 09/02/21	Tue 09/02/21	
15	Risk Table	1 day	Mon 08/02/2	Mon 08/02/2	
16	Interim Report Submission	0 days	Fri 19/02/21	Fri 19/02/21	7,8
17	**Final Report**	**70 days?**	**Mon 15/02/2**	**Fri 21/05/21**	
18	Research 3D Modelling Environment	2 days	Mon 15/02/21	Tue 16/02/21	6FS+3 days
19	Create a basic 3D environment	5 days	Wed 17/02/21	Tue 23/02/21	18
20	Define Platform characteristics	2 days?	Fri 19/02/21	Mon 22/02/21	7
21	Define medium characteristics	2 days	Fri 19/02/21	Mon 22/02/21	7
22	Simulate Antenna beam	5 days	Tue 23/02/21	Mon 01/03/21	20
23	Simulate first order contributions	7 days	Wed 24/02/21	Thu 04/03/21	19
24	Implement Matched Filtering	2 days	Fri 05/03/21	Mon 08/03/21	23,22
25	Create output image	5 days	Tue 09/03/21	Mon 15/03/21	24
26	Simulate second order contributions	5 days	Tue 16/03/21	Mon 22/03/21	25



Project: fyp
Date: Fri 12/02/21

Task | Split | Milestone | Summary | Project Summary | Inactive Task | Inactive Milestone | Inactive Summary | Manual Task | Duration-only | Manual Summary Rollup | Manual Summary | Start-only | Finish-only | External Tasks | External Milestone | Deadline | Progress | Manual Progress

ID	Task Name	Duration	Start	Finish	Predecessors
27	Increase to higher orders	2 days	Tue 23/03/21	Wed 24/03/21	26
28	Model a scene with multiple buildings	3 days	Tue 23/03/21	Thu 25/03/21	26
29	Extract scene from external mapping tool	6 days	Tue 23/03/21	Tue 30/03/21	26
30	Decide on secondary method	1 day	Thu 25/03/21	Thu 25/03/21	27
31	Implement secondary more efficient method	10 days	Wed 31/03/21	Tue 13/04/21	29,30
32	Extra time for overspill or extra features	15 days	Wed 14/04/21	Tue 04/05/21	31
33	**Report Writing**	**52 days**	**Wed 24/02/2**	**Thu 06/05/2**	
34	Create Background	45 days	Mon 01/03/21	Fri 30/04/21	
35	Methods	48 days	Wed 24/02/2	Fri 30/04/21	
36	Documentation	48 days	Wed 24/02/2	Fri 30/04/21	
37	Milestones review	20 days	Mon 05/04/21	Fri 30/04/21	
38	Deliverables description	20 days	Mon 05/04/21	Fri 30/04/21	
39	Results exploration	20 days	Mon 05/04/21	Fri 30/04/21	
40	Uncertainty	20 days	Mon 05/04/2	Fri 30/04/21	
41	Conclusion	20 days	Mon 05/04/2	Fri 30/04/21	
42	Further work	15 days	Mon 12/04/2	Fri 30/04/21	
43	Extra parts	15 days	Mon 12/04/2	Fri 30/04/21	
44	Review and revise	4 days	Mon 03/05/21	Thu 06/05/21	
45	Final Report Submission	0 days	Fri 07/05/21	Fri 07/05/21	33,44
46	Design Poster	7 days	Wed 28/04/2	Thu 06/05/2	
47	Poster Submission	0 days	Fri 07/05/21	Fri 07/05/21	46
48	Presentation and Viva Voce	10 days	Mon 10/05/21	Fri 21/05/21	
49	Poster Exhibition (Interim)	0 days	Fri 21/05/21	Fri 21/05/21	



Project: fyp
Date: Fri 12/02/21

Task
Split
Milestone
Summary
Project Summary
Inactive Task
Inactive Milestone
Inactive Summary
Manual Task
Duration-only
Manual Summary Rollup
Manual Summary
Start-only
Finish-only
External Tasks
External Milestone
Deadline
Progress
Manual Progress

1.4 Risk Table

Risk	Impact	Mitigation
Computer dies and can't be used	Delay project, some data may be lost	Data backed up automatically to iCloud and three weeks extra are built into the project to account for major delays, if there are more delays then non-key milestones can be removed
Discover chosen tools can't perform required functions	Delays project while new tools have to be found and existing work is ported over	Three week grace period as mentioned previously, spend more time initially to ensure chosen tools can perform required functions
Feature creep during design and implementation phase	Important features are delayed while unimportant features are added	Once the plan is finished stop adding features until all of the planned features are added then revisit the planned features, extra time is added into the project plan in order to facilitate this
Eye strain	Work will be slowed down due to having to recover	Ensure workspace is set up properly initially following the University of Bath Health and Safety Standard provided, follow 20-20-20 rule
Bug in program that makes the program unusable	Delay project due to debugging and re-writing	Extra time built in to project plan to deal with this
Progress grinds to a halt due to the difficulty of implementation	Delay project	Fallbacks methods have been identified within MATLAB which can be used as a platform for more advanced implementation

Part 2

Project Background

Synthetic aperture radar (SAR) is a method of radar imaging that allows for a greater dimension of radar aperture than is possible in a single element or an array of elements. This is generally achieved by mounting a radar transceiver to a moving platform and moving it past the target to be imaged however the transceiver can be stationary and the target moved past it. The image uses the relative movement between the platform and the target to build up an image based on the responses received and then time multiplexing all the responses. This movement, and hence the larger aperture created, allows for a finer spatial resolution than can be achieved by more conventional beam scanning radar. These images can be useful for a variety of applications such as remote mapping of the surface of the Earth for glaciology and geology among others. Interferometric SAR can also be used to create elevation maps by taking two measurements from different positions and comparing the differences. Simulating a SAR image is desirable as flight time can be expensive and time consuming to get quality images so defining the flight paths and angles before any imaging is carried out saves both time and money.

There are three main methods that have previously been explored for simulating SAR. The first of these, which is the most accurate is simulating the effects of the beam itself, as detailed in Franceschetti *et al.* 2003 [1]. This describes creating the raw data as would be expected in a SAR system in the real world and then transforming it to create a SAR image. The ground work for this was laid in Franceschetti *et al.* 1992 [2], which itself builds on Francescetti and Schirinzi's work on a SAR processor based on two-dimensional FFT codes [3]. This approach has the advantages of being able to take into account backscattering as well as the higher-order signal contributions created by scattering as the signal reflects off of both a wall and the ground. A lot of the preliminary work with regards to the behaviour of electromagnetic backscattering was conducted in [4]. This describes a model that allows for the return from a structure to a microwave sensor to be analysed and so determine its dielectric properties as well as its geometric properties. The optics can be altered to simulate the roughness of the surface the signal is reflected off of last before it is received by the sensor. This can also be expanded to simulate backscattering. This simulator as a whole works very well with individual objects, however the simulation time was found to scale linearly with the number of objects present in the scene. Simulations in [1] were carried out using a Pentium IV processor from Intel Corp, released in 2001 which means that the simulation times achieved are not necessarily reflective of the performance achievable on more modern processors, however at the time it was found that while one object in a 512x512 pixel image required approximately 34 seconds, two objects in the same sized image required 1'02" and 16 objects increased this computation time up to 7'38". This is not ideal if the aim is real-time or even near real-time simulation capabilities as most urban environments that are to be simulated will contain far more than 16 structures. This efficiency does seem to have been improved in Franceschetti *et al.* [5] as in this paper it is used to simulate a SAR image of a $400 \times 600\text{m}^2$ area of the centre of Munich, incorporating the Technische Universität and the Alta Pinakothek, which leads to a reasonably complicated scene.

A similar approach building on the previously mentioned work is presented by Zheng *et al.* [6] which uses the scattering model in order to accurately simulate a SAR scene. This is achieved by making the assumption that the scene is made up of vertical buildings distributed on a rough dielectric terrain. This also discusses the computation of scattering coefficients under different conditions. As with the previous approach (and the following raw SAR approach) the Kirchhoff approach is used in this case.

A similar approach to robust SAR simulation is described in [7], which uses an electromagnetic approach closely following Maxwell's equations to create a finite-difference time domain method of simulation, which differs from [1] as that instead manipulates the signal in the frequency domain in order to create the raw SAR signal. This system has similar drawbacks to Franceschetti's approach with regards to computation complexity and time required, but is even more computationally complex, due to its ability to handle dispersive materials as well as phase changes. This approach could be of some interest for creating an incredibly robust simulator due to the advances in computing performance between the publication date and today.

There are two methods that are less complex computationally for simulating the image achieved by SAR. These both arise from graphics manipulation and more specifically 3 dimensional modelling. These are rasterisation and ray tracing. Rasterisation is the process of taking a 3 dimensional area and creating a 2 dimensional image using some sort of image transformation based on the location of the camera to the object in the 3D area. Ray tracing follows a similar process to light, just in reverse. It sends beams out from the camera in all directions that the camera can see and simulates them reflecting off of objects until they hit a source of illumination. Using this the rendering engine can determine if an object is illuminated or not and adjust appropriately. These two approaches have both been used in the past for simulation of SAR images. Rasterisation was used by Balz 2006 [8] and ray tracing was explored by Auer *et al.* 2008 [9] and Mametsa *et al.* 2002 [10]. This second paper was used in Hammer *et al.* [11] along with Balz 2006 to compare the relative merits of these two approaches. Ultimately the advantages of Balz's approach as detailed in [12] and [8] is that the simulator is real-time so can be used to determine optimal azimuthal directions when recording a SAR image using a physical airborne platform. This form of simulation however doesn't take into account the contributions of higher order reflections so if it is being used for a demonstration tool, the images produced are less representative of an actual SAR image. This also means that corners aren't visible in the image, meaning that this form of simulation can't be used to test feature extraction algorithms. The two ray tracing simulators tested within [11] can simulate these higher-order contributions meaning they can be used to test feature extraction algorithms, however due to the computationally intensive nature of ray tracing these can't be simulated in real time, so this form of simulation is less useful for planning purposes.

All of the forms of simulation presented here so far are only as good as the models used, as generally the modelled buildings are assumed to be made of one material with a fixed dielectric constant, while the modelled ground is made of a different material. In [11], the buildings modelled are created entirely of stone, with a grass ground. In real life however this becomes more complicated as buildings do tend to be made up of multiple materials, and often in urban areas the ground is made up of a material with a similar dielectric constant to the buildings. This also doesn't include any forms of dispersive materials such as metal. It is unclear whether any of the simulators covered in [11] can simulate these sorts of materials accurately, which is something that should be taken to account in the future.

Yet another method similar to the image processing approaches is presented by Lu *et al.* [13] and uses methods most commonly used by video games. In this case the raw mathematic SAR simulation is not carried out, similar to [14] and [9] and uses an orthogonal projection to cast the 3D model to a 2D image based on the slant range of the camera. This again has the potential of improving the performance of image generation in incredibly complicated scenes such as city centres.

Chen *et al.* 2011 [15] proposes a hybrid method of simulating these images using analytical models. In this approach, the contributions provided by backscattering are shown using an electromagnetic analytical model ultimately based on that described by [1], and the object position vectors are given by a geometric model using ray tracing. In this paper the analysis is mostly focused on cylindrical or cylinder-like buildings, with some analysis devoted to flat-roof buildings as well. This approach has a potential to be more accurate than the ray tracing models proposed previously due to the electromagnetic models involved, but also be less computationally complex than the more canonical SAR image simulators.

Xu and Jin 2006 [16] is a paper that is not immediately relevant to this situation due to it modelling natural environments for SAR simulation however this is potentially interesting and relevant due to the variety of materials present in urban environments to be modelled. This is an algorithm that can deal with randomly and heterogeneously distributed objects within an image with varying dielectric constants and can deal with them in a reasonable length of time. This is something that is worth exploring if there is time in the future as it would add robustness to the system as a whole.

Bibliography

[1] G. Franceschetti, A. Iodice, D. Riccio, and G. Ruello, "SAR raw signal simulation for urban structures," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 41, no. 9, p. 10, 2003.

[2] G. Franceschetti, M. Migliaccio, D. Riccio, and G. Schirinzi, "SARAS: A synthetic aperture radar (SAR) raw signal simulator," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 30, no. 1, pp. 110–123, Jan. 1992, ISSN: 1558-0644. DOI: 10.1109/36.124221.

[3] G. Franceschetti and G. Schirinzi, "A SAR processor based on two-dimensional FFT codes," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 26, no. 2, pp. 356–366, Mar. 1990, ISSN: 1557-9603. DOI: 10.1109/7.53462.

[4] G. Franceschetti, A. Iodice, and D. Riccio, "A canonical problem in electromagnetic backscattering from buildings," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 40, no. 8, pp. 1787–1801, Aug. 2002, ISSN: 1558-0644. DOI: 10.1109/TGRS.2002.802459.

[5] G. Franceschetti, R. Guida, A. Iodice, D. Riccio, G. Ruello, and U. Stilla, "Simulation Tools for Interpretation of High Resolution SAR Images of Urban Areas," in *2007 Urban Remote Sensing Joint Event*, Apr. 2007, pp. 1–5. DOI: 10.1109/URS.2007.371841.

[6] Y. Zheng, B. Zhou, and Z. Zong, "Simulation method of SAR raw echo for urban scene," in *2008 International Conference on Radar*, Sep. 2008, pp. 503–507. DOI: 10.1109/RADAR.2008.4653976.

[7] J. Delliere, H. Maitre, and A. Maruani, "SAR measurement simulation on urban structures using a FDTD technique," in *2007 Urban Remote Sensing Joint Event*, Apr. 2007, pp. 1–8. DOI: 10.1109/URS.2007.371837.

[8] T. Balz, "Real-time SAR simulation of complex scenes using programmable Graphics Processing Units," Jul. 1, 2006.

[9] S. Auer, S. Hinz, and R. Bamler, "Ray Tracing for Simulating Reflection Phenomena in SAR Images," in *IGARSS 2008 - 2008 IEEE International Geoscience and Remote Sensing Symposium*, vol. 5, Jul. 2008, pp. V - 518-V –521. DOI: 10.1109/IGARSS.2008.4780143.

[10] H.-J. Mametsa, F. Rouas, A. Berges, and J. Latger, "Imaging radar simulation in realistic environment using shooting and bouncing rays technique," in *SAR Image Analysis, Modeling, and Techniques IV*, vol. 4543, International Society for Optics and Photonics, Jan. 28, 2002, pp. 34–40. DOI: 10.1117/12.453975.

[11] H. Hammer, T. Balz, E. Cadario, U. Soergel, U. Thoennessen, and U. Stilla, "Comparison of SAR simulation concepts for the analysis of high-resolution SAR data," in *7th European Conference on Synthetic Aperture Radar*, Jun. 2008, pp. 1–4.

[12] T. Balz and N. Haala, "Improved Real-Time SAR Simulation in Urban Areas," in *2006 IEEE International Symposium on Geoscience and Remote Sensing*, Jul. 2006, pp. 3631–3634. DOI: 10.1109/IGARSS.2006.930.

[13] Y. Lu, K. Wang, X. Liu, and W. Yu, "A GPU based real-time SAR simulation for complex scenes," in *2009 International Radar Conference "Surveillance for a Safer World" (RADAR 2009)*, Oct. 2009, pp. 1–4.

[14] T. Balz and U. Stilla, "Hybrid GPU-Based Single- and Double-Bounce SAR Simulation," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 47, no. 10, pp. 3519–3529, Oct. 2009, ISSN: 1558-0644. DOI: 10.1109/TGRS.2009.2022326.

[15] H. Chen, Y. Zhang, K. Tang, W. Xiong, and C. Ding, "Radar imaging simulation for typical urban structures based on analytical models," in *Proceedings of 2011 IEEE CIE International Conference on Radar*, vol. 2, Oct. 2011, pp. 1362–1365. DOI: 10.1109/CIE-Radar.2011.6159811.

[16] F. Xu and Y. Jin, "Imaging Simulation of Polarimetric SAR for a Comprehensive Terrain Scene Using the Mapping and Projection Algorithm," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 44, no. 11, pp. 3219–3234, Nov. 2006, ISSN: 1558-0644. DOI: 10.1109/TGRS.2006.879544.