

Simulation of FLIR and LADAR Data Using Graphics Animation Software

Gavin Powell, Ralph Martin and David Marshall
Department of Computer Science,
Cardiff University
PO Box 916
Cardiff, CF24 3XF, UK
{g.r.powell, ralph, dave}@cs.cf.ac.uk

Keith Markham
British Aerospace/Matra Dynamics,
Filton,
Bristol, UK
keith.markham@bae.co.uk

Abstract

This paper presents an implementation of Forward Looking Infrared (FLIR) and Laser Radar (LADAR) data simulation for use in developing a multi-sensor data-fusion automated target recognition (ATR) system. Through the use of commercial models and software we can create a highly detailed scene model, which provides a rich data set for later processing. We embed our own modules within this software to extract data from the scene and then present it to the FLIR and LADAR sensor models. These models produce simulated range and temperature readings for objects within the scene. These data are used to create a variety of images to aid in the visualisation of the data and test our ATR system. Frames from approach sequences show the LADAR and FLIR data in different formats. The simulated data and subsequent images are accurate and rapid to produce and provide invaluable data resources for testing an ATR system.

1. Introduction

We are creating an Automated Target Recognition (ATR) system for use in Air-to-Air homing missiles that tracks to target through fusing data from LADAR and FLIR sensors.

A necessity of our research is the ability to simulate sensor data. It allows for data to be collated that would otherwise be unavailable to us due to cost, security restrictions, or other impracticalities. Simulation software also allows the user to tailor data to their specific domain allowing systems to be developed that will hopefully provide superior performance when introduced to the real world. In future work we intend to fuse LADAR and FLIR data using Dempster-Shafer techniques and tabu search, and through doing so enhance the capabilities of our Air-to-Air ATR system compared to one reliant on single modality sensor data. Air-to-Air scenarios are not the only

application of such a system, the ideas can easily be adapted to other vision based tasks such as autonomous vehicle guidance and object recognition.

At present there is no available simulation software that meets the needs of our ATR system, so the development of such a system was imperative. Test systems can be introduced into dynamic environments, allowing interactions to take place between the simulation environment and the system under development, so that tests can evolve over time producing a much closer match to the real world. To accurately model the real world, atmospheric effects should be taken into account, as they can play a major role in sensor readings, especially for LADAR which can be sensitive to dust, smoke and moisture particles within the atmosphere. Advances in particle modelling systems within computer graphics have made this a feasible task.

The importance of data simulation in the field of airborne target recognition is prominent due to the high cost involved in gathering real world data. The ability to test systems in this area within a computer generated environment can accelerate algorithm development dramatically while keeping costs down. In our system the fusion of FLIR and LADAR data should enable accurate detection, recognition and tracking of aircraft and aim point refinement of missiles.

The use of infrared is a mature technology [1, 2, 3, 4, 5] of which imaging is one of many applications. In imaging, infrared is a cheap means of accumulating data with high information content. Its ability to sense heat allows us to 'see' in dark and other low visibility conditions which may include, fog, smoke or dust. Infrared can also detect hotspots in the scene, which is particularly useful if we are looking for vehicles of one kind or another, as they tend to emit high levels of heat compared to the natural objects in the scene. Due to the maturity of this technology it is well understood and much research has taken place in understanding, modelling and simulating infrared sensors [1, 2, 3, 4, 5].

Recent advances in laser technology have meant that LADAR has now become a viable option where before it

was too expensive, or the data that it provided was of poor quality. Its uses in vision are obvious as it can directly obtain three dimensional shape data from the object it is scanning. As sensor technology advances and new sensors become viable then so the simulation of these sensors and their environments must advance too if we are to create a true representation of current day sensing environments. The workings of LADAR sensors are well understood [6, 7] and simulations of such models have been implemented [8, 9, 10]. Recent developments consider the effect that factors such as part occlusion, beam profile and scanning errors can have on the signature and how we can predict it more accurately.

The fusion of LADAR and FLIR data is a useful technique as the sensors complement each other when used in an ATR scenario: where FLIR fails LADAR excels and vice versa. FLIR can readily detect a target in a wide field of view, especially airborne vehicles that expel very hot exhaust gasses. But it has a limited resolution and can only detect in two dimensions. On the other hand it is costly for LADAR to detect over a wide field of view. It is better suited to scanning a small area where it can accurately detect features of an object in three dimensions. It is for these reasons that the two can be fused to produce a more robust ATR system.

Our simulation process involves the use of commercial modelling and raytracing software to extract the necessary data from within the scene, which is then presented to the sensor models so that the simulation can be carried out.

2. System overview

The aim of this software is to accurately simulate FLIR and LADAR data. This is achieved by modelling 'blue sky' Air-to-Air scenarios in a commercial raytracing package. Surface ray data is produced by this software which is fed into our FLIR and LADAR simulators, in turn, these produce simulated temperature and range readings at a sub-pixel level. This output is the required simulation of FLIR and LADAR sensors which is used as input by the ATR system.

The basic design of our system is shown in Figure 1. The process begins by entering our scenario into the commercial modelling package SoftImage. This creates a 3-D database of the scene that describes all properties and entities for later retrieval. Once the modelling and scenario is complete we can render it at a sub-pixel level using the commercial MentalRay software. Each interaction between the traced ray and a vehicle surface creates an interaction event which in turn causes MentalRay to access the 3-D database, along with our software, and extract all the necessary data about that point such as angle of incidence, position in space and surface normal. This data is then fed to the LADAR and FLIR models which

synthesise range and temperature values respectively. These values are then manipulated into various formats, such as a movie sequence for ATR processing, VRML or imported back into Softimage. This allows us to visualise and compare results in interactive 2D and 3D environments. A sequence of images is created and joined to produce a movie showing, for example, an approach path towards an aircraft. The 3D data is re-introduced into SoftImage so that it can be overlaid onto the original model for examination and comparison. Finally the 3D data is converted into VRML for interactive 3D viewing of the scene.

3. Scene modelling

The modelling of scenes is a very important part of any simulation process as it is the source of data that the sensor

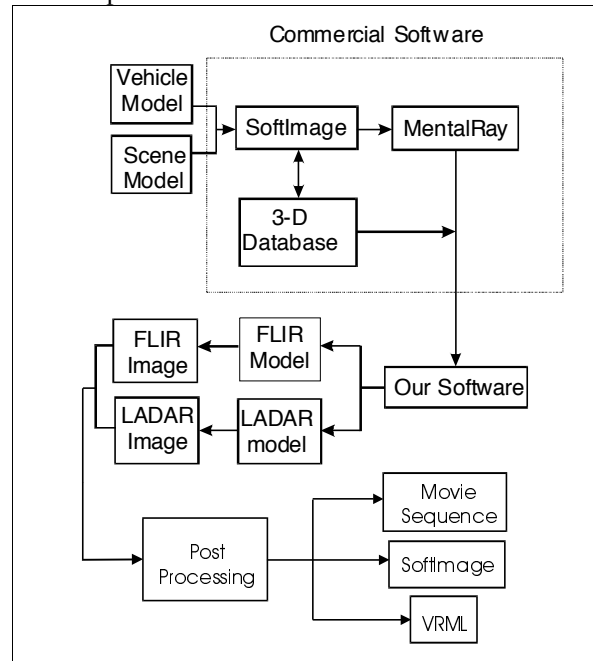


Figure 1. System Flow Diagram

model will acquire. Data must be accurate as the sensor model will be sensitive to ambiguities and inaccuracies. For this reason we have used the most powerful modelling and animation software that is commercially available at present, Avid's Softimage3D. Due to its popularity and flexibility, many thousands of models are available on the Web, commercially or free, allowing us to use 20000+ polygon vehicle models in our system. We then introduce them to our scene, which we model as required (see Figure 2). Modelling of objects in SoftImage can be simple, but there are many complex features that allow a vast array of special effects to be created. Scenes can be created from a variety of entities including polygons, NURBS, surface patches and meta-balls. The scene in

Figure 2 was modelled using two polygonal grids, a polygonal sphere and two vehicle models purchased from a modelling company. The grids represent the lake and mountains. All objects are texture mapped with displacement to create a more realistic image.

Such a scene model provides us with a strong basis for extracting data to present to our fusion and image processing system for further processing. Such ease of use is necessary to make our approach a viable option for data simulation.

3.1. Particle systems

Particle systems allow the visualisation and simulation of fluid movements of various elements or objects such as rain, swarms of bees, and so on. Such systems are commonly used for animating elements such as fog, which are difficult to animate manually. The particles are assigned behaviours and interaction rules between themselves and their environment to create desired visual effects.

Modern particle systems have many complex features allowing for more realistic scenes to be generated. Many particle suites now allow for advanced collision detection, multiple sources and interactions with other objects. Natural forces such as friction, gravity, wind and electromagnetic fields may affect the particles themselves.

This type of system is also very useful for simulating real world data. Recent advances in the simulation of jet exhaust plumes from infrared signatures have been based on particle systems[11, 12].

When designing our system we noticed that few environment simulators model atmospheric effects well, and simply use data obtained from tables, which provides a very static environment. After some initial investigation we have concluded that particle systems are ideal for such modelling. SoftImage has a particle module (see Figure 3) that allows for particles to be created as small polygonal objects that can be animated thus allowing for atmospheric conditions to be included in our scene model.

4. Data collection

Our sensor models act upon a variety of data, which we need to extract from our modelled scene. The necessary data is obtained from the three-dimensional scene database at rendering time through software interfaced to the rendering software.

Rendering describes the overall process of going from a database representation of three dimensional objects to a two dimensional projection on a view surface. One of the best known rendering methods is ray tracing. The final image has pixels that correspond to ray interaction events within the scene. The visible surface and hence colour and

intensity at that point are obtained by tracing a ray backwards from the imaginary eye through the pixel in the image into the scene. Where this ray intersects an object, a local calculation determines the colour that results from direct illumination at this point. If the object is partially reflective, transparent or both, the colour of the point in the image plane includes a contribution from reflected and transmitted rays from elsewhere in the scene. These rays must be traced backwards to discover their contribution. The level of recursion of ray reflections or refractions to be followed is the ray tracing depth.

This technique for extracting information from a three dimensional database is very similar to the way in which LADAR works. The LADAR system releases a beam of light that hits an object, is reflected, and is then analysed on its return to determine properties of the object that it has just struck.

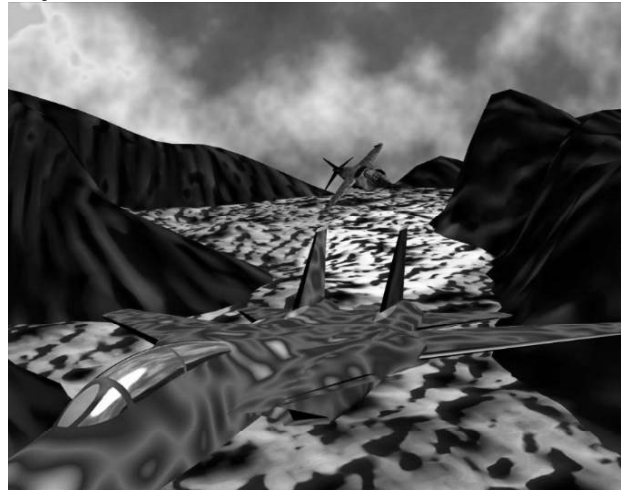


Figure 2. SoftImage scene of F-14 and GR7

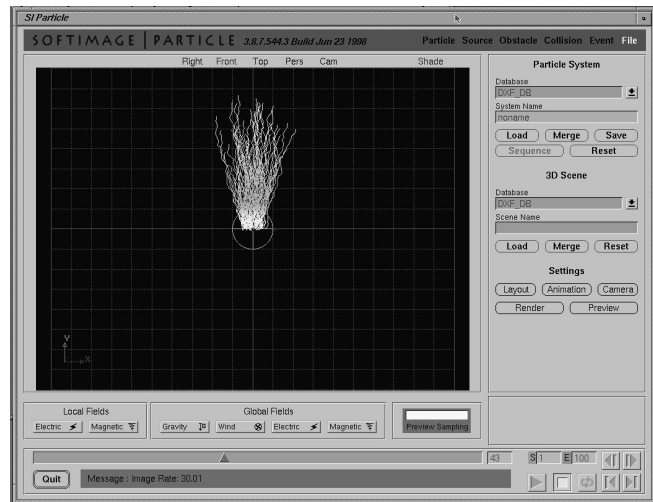


Figure 3. SoftImage Particle

We use Avid's MentalRay commercial software to ray trace the scene. A software developers kit allows us to

create plug-in modules that extract all necessary information from the scene at each ray interaction event. The versatility of MentalRay and SoftImage allows for extremely useful data to be modelled and extracted from the scene. Alpha values can be used to signify transparency of objects that may occlude an object and block a percentage of the LADAR signal [10]. The number of rays per pixel can be adjusted allowing for the simulation of different beam profiles [10]. This all fits together with SoftImage to provide a simple, accurate and effective means of modelling and extracting data from a scene.

The FLIR data is collected at the same instant as the LADAR data. A different data set is created as the FLIR model requires distances from heat source to surface, about the object, as well as sharing some measurements which are common to both sensor models.

5. FLIR simulation

The FLIR process models surface heat due to heat flux from the sun and also heat flow through a model that has a heat source within it, such as an engine. We use a simplified heat transfer model that assumes the heat source, in this case the engine, is a point source. For an initial system this is sufficient. A more sophisticated approach might use a more complex heat transfer function or multiple heat sources as found in twin-engined aircraft. The temperature due to heat flow from the sun and the heat flow from the point heat source within the object are calculated. These are then combined to give overall temperature of the object's surface.

5.1. Heat flow due to internal heat source

We simplify the model by assuming one dimensional heat flow within the object from a point source. Fourier's Law gives the basic equation for one-dimensional heat conduction within an object

$$q_x'' = k \frac{T_2 - T_1}{L} \quad (1)$$

where q_x'' is the heat flux, k is the thermal conductivity, T_1 is the temperature at the exterior surface, T_2 is the temperature at the point source, and L is the distance that the heat radiates along a straight line.

5.2. Heat flow due to solar radiation

It can be assumed that heat flow into the object equals heat flow from the object [2] so the net heat flow at a surface of the object is given by

$$W_{abs} = W_{cv} + W_{rad} + W_{bal} \quad (2)$$

where W_{abs} is the heat flux absorbed, W_{cv} is convectational heat loss at the surface and W_{bal} is heat lost by other processes. Solar input is given by

$$W_{abs} = \alpha_s W_s \cos \theta, \quad \text{where} \quad (3)$$

$$W_s = \frac{A}{\exp(B/\sin a)}$$

and W_s is the solar radiation of a surface whose normal is parallel to the Sun's rays, θ is the angle made by the sun's rays, a is the angular altitude of the sun and α is the solar absorbtivity of the surface. The values of A and B are parameters for the time of year that are related to the position of the object on the Earth with respect to the Sun [13].

The Stefan-Boltzman law gives the net heat flux lost to the surrounding atmosphere due to radiation

$$W_{rad} = \epsilon_0 \sigma (T_s^4 - T_{amb}^4) \quad (4)$$

where T_s is the surface temperature, T_{air} is the temperature of the surrounding air, ϵ_0 is the emissivity of the material and σ is the Stefan-Boltzman constant.

Heat flux lost to the surroundings by convection is given by

$$W_{cv} = h(T_s - T_{air}) \quad (5)$$

h being the heat transfer coefficient. The value of h is dependent on various properties of the surrounding air including its speed relative to the object [14].

The final term in Equation (2) is W_{bal} . This can be found from

$$R = \frac{W_{abs}}{W_{bal}} \quad (6)$$

where Nandhakumar and Aggarwal [2] show how to evaluate R .

Substituting into Equation (3) we get

$$\sigma \epsilon T_s^4 + h T_s + (R - 1) W_s \alpha_s \cos \theta - h T_{air} - \sigma \epsilon_0 T_{air}^4 = 0 \quad (7)$$

The real roots to this equation are found using Laguerre's method [4], [15]. It is extremely difficult to model heat flux for surfaces that are in shadow. A surface is in shadow if $\theta \geq \pi/2$. When this is the case the third term of Equation (7) becomes zero and forces the equation to be $T_s = T_{air}$ so we assume that the sun does not heat the surface.

5.3. Surface temperature

We now have two elements providing a temperature change at the surface through Equation (1) and Equation (7). The temperature of the surface is given by

Symbol	Description
A_{sys}	System gain
Γ	Apparent complex reflection coefficient
U	Complex envelope function
τ	Round trip delay
u_{LO}	Local oscillator complex envelope
f_{IF}	Intermediate frequency
f_d	Doppler frequency
ρ_i	Optical detector responsivity
P_T	Average transmitter output power

Symbol	Description
P_{LO}	Local oscillator power
A_e	Effective receive aperture area
η_H	Heterodyne efficiency
β_{opt}	Receiver optical efficiency
γ	Normalised complex scattering coefficient
η	2-way path transmission coefficient
ρ	Diffuse reflectivity
φ	Target surface incident angle

Table 1. List of symbols

$$T = T_s + T_1 \quad (8)$$

Finally two types of noise are added. Firstly shot noise is added where a portion of the pixels are coloured black and the same portion are coloured white. This type of noise simulates the dead and saturated pixels found in FLIR images [3]. Gaussian noise is also added to simulate sensor and system noise.

6. FM LADAR simulation

FM LADAR sensors have been modelled previously, and used to synthesise images [10]. We use such techniques in our system according to our requirements. The simulated LADAR system uses a modulated laser beam to

scan the target. The returned beam is then coherently mixed with a portion of the original beam via a local oscillator and passed to the optical detector to be processed. The detector transforms the light energy into an electrical current. This current is sinusoidal and its frequency is relative to the range of the target [10]. Coherent detection LADAR systems can be effectively immune to background noise if the local oscillator power is large enough: their signal-to-noise ratio (SNR) is directly proportional to the power received as opposed to power squared in incoherent systems [7].

The optical detector will provide a current from a target at range r of

$$s(t) = A_{sys} \frac{1}{r} \text{Re}\{\Gamma u(t - \tau) u_{LO}(t) \exp(-j2\pi(f_{IF} + f_d)t)\} \quad (9)$$

where

$$A_{sys} = 2\rho_i \sqrt{P_T P_{LO} A_e \eta_H \beta_{opt}} \quad (10)$$

$$\Gamma = \gamma \sqrt{\eta \frac{\rho}{\pi} \cos \varphi} \quad (11)$$

Table 1 gives a brief description of the various terms in Equations 9-11. A fuller description of the terms in

Table 1 is provided by Bevington [10], Jelalian [7] and Cox [6].

To be able to compute the range of the target we need to use Fourier analysis. It has been shown that the profile produced is a function of the range [10] (see Equations (12) to Equation (19))

$$s_o(v) = \frac{1}{2} [s_{o+}(v) + s_{o+}(-v)] \quad (12)$$

$$s_{o+}(v) = A_{sys} \frac{1}{r} \Gamma \exp(-j2\pi v \tau) h(v - (f_d \mp \tau B_{FM} / T_p)) \quad (13)$$

The amplitude of the Fourier transform of the local oscillator at frequency v is

$$h(v - (f_d \mp \tau B_{FM} / T_p)) \quad (14)$$

This contains the up and down sweep of the frequency that eradicates any Doppler shift due to radial motion of the target.

The remainder of Equation (13)

$$A_{sys} \frac{1}{r} \Gamma \exp(-j2\pi v \tau) \quad (15)$$

provides the signal return from the surface broken down into its relative sinusoidal and co-sinusoidal components at frequency v .

6.1. Details

We take a rectangular pulse waveform and find the Discrete Fourier Transform (DFT)

$$\text{for } n = 0, 1, \dots, N-1 \quad F(n) = \sum_{k=0}^{N-1} h_k \exp(-j2\pi kn / N) \quad (16)$$

This gives us a set of points in Fourier space which are symmetrical about $N/2$ when we plot $|F(v)|$.

As we take more samples of the pulse within the same time duration the sample interval decreases. This gives us a wider bandwidth of frequencies, which allows us to effectively estimate the original waveform to a better resolution:

$$f_n = \frac{n}{N\Delta} \quad \text{where} \quad (17)$$

$$n = -\frac{N}{2}, \dots, \frac{N}{2}$$

here $N\Delta$ is the duration of the pulse, T_p . Knowing the bandwidth, B_{FM} , of the pulse envelope function provides us with v at $N/2 = B_{FM}/2$. We interpolate between the frequencies to get better definition without increasing the bandwidth by zero padding the data sequence with $L-N$ zeros.

The frequency and amplitude values obtained from the DFT are placed into a look up table for future reference by the system.

Once we have evaluated Equation (12) we must find the magnitude

$$y(v) = \left| \sum s_o(v) \right| \quad (18)$$

This is repeated for all samples in the up and down frequency sweep. The frequencies of the peak values are obtained to enable us to calculate the range

$$\hat{r} = \frac{cTp}{4B_{FM}} \left(\hat{f}_{down} - \hat{f}_{up} \right) \quad (19)$$

From this data we can produce a simulated 3D depth map of the scene this 3D shape data of a target aircraft is used in the ATR system for object recognition purposes.

7. Results

The previously described simulations have been used to create a variety of image sequences of military aircraft. The results of these can be seen in Figure 4 and Figure 5.

Both show images taken in ‘blue sky’ conditions with no white out from the sun and no background scene, as such would be found when looking skyward from the sensor to the target. Figure 4 shows an F14 broadside. In the LADAR image lighter colours denote near and darker colours far objects. The “blockiness” of the image shows the quite low resolution of the LADAR scan. The FLIR image shows a noisy image, where lighter colours denote hotter areas; the rear of the aircraft where the engine is situated is hottest. Figure 5 shows the same aircraft but at a higher resolution, which would be obtained on approach to the aircraft.

Figure 6 shows a SoftImage model of an F-14 which has had simulated LADAR data overlaid onto it. This is just one of the many manipulations that SoftImage can do with such graphical data to aid visualisation of the modelling of sensor output and to check that results are acceptable.

Even though the FLIR model is simplistic in nature the results shown provide images of a quality that are acceptable to work with. The introduction of a more complex model using a more detailed analysis and

implementation of sensor noise, for example, would be simple.

The LADAR results are quite promising and the flexibility of our system allows us to quickly produce results using various parameters, which can easily be viewed in 3-D in SoftImage or in VRML through a converter that we have written. Similar real airborne data is very hard to obtain so comparisons between our data and real data cannot be made. This work is part of an ongoing project for British Aerospace and even though security considerations present them from sharing real data with us, they have seen and accepted the current results. Readers wishing to view real LADAR imagery can consult the report by Beveridge [16]. Figure 7 (a) and (b) show a sample of a jet exhaust simulated using SoftImage particles which we aim to add to our simulations later. In this case polygonal objects have not been used but instead coloured points are entered into the image by SoftImage. A new fluid-dynamic simulator from Phoenix Tools has also been made available that can produce very good fog and smoke like effects which we also aim to utilise at a later date.

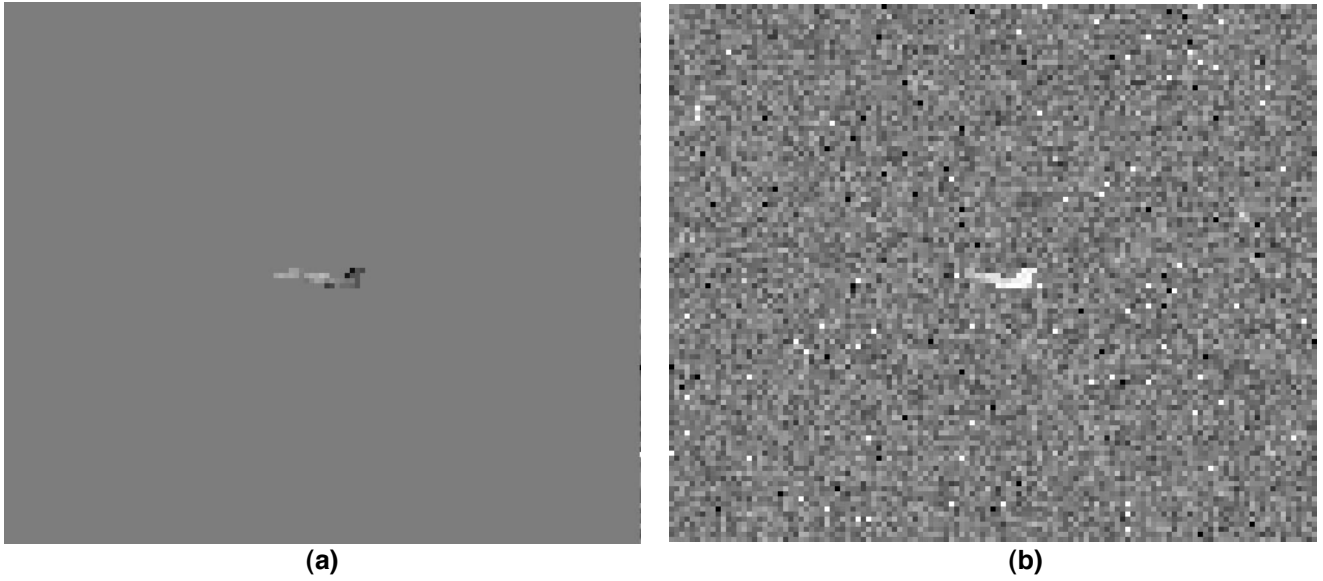
8. Conclusion

We have presented a simulation environment that should provide useful in developing and testing algorithms in airborne ATR scenarios. We have shown that high quality simulated data can be easy and quick to produce and that modelling of sequences is also easy. These are useful for testing algorithms more fully than with static images, especially when modelling approaches of missile to target. Future work in the simulation of LADAR will focus on modelling of external factors such as atmospheric conditions.

Through the use of alpha values for scene objects, beams passing through semi transparent objects, such as foliage, can be modelled in the scene and sensor [10]. This is expected in future to be a useful tool for LADAR image synthesis. Our system has been so designed that additions and improvements of such a nature are easy to ‘plug in’ once the appropriate model has been devised, leaving room for continual improvement.

Acknowledgements

This work was performed under sponsorship from British Aerospace/Matra Dynamics. The authors would like to thank BAe/Matra Dynamics for their support through this research. We also wish to thank John Black for his assistance with infrared simulation.



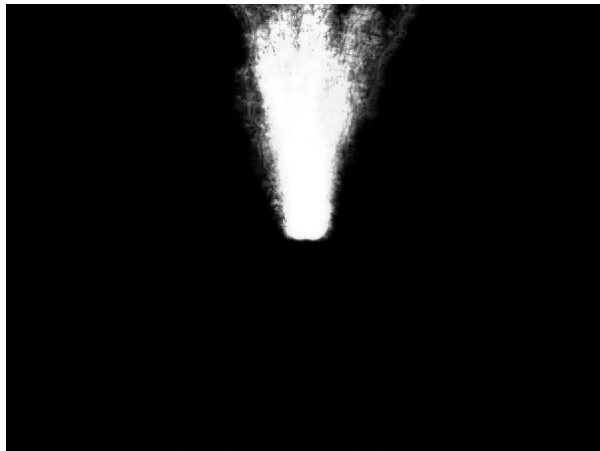
(a) (b)
Figure 4. (a) LADAR simulation (b) FLIR simulation.



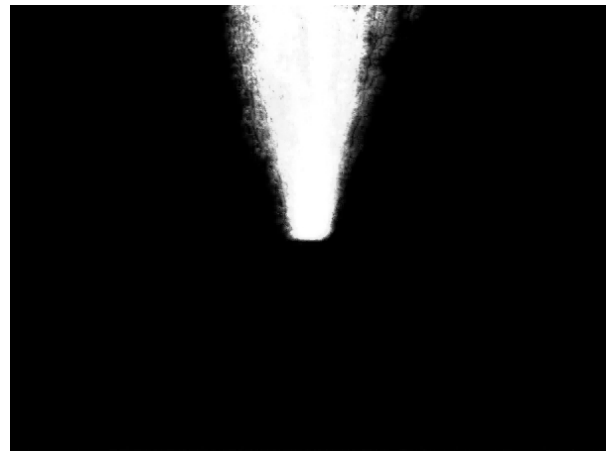
Figure 5. Frame from LADAR and FLIR simulation movie sequence



Figure 6. F-14 with LADAR data overlaid.



(a)



(b)

Figure 7 (a) and (b). Frames from Jet Exhaust in SoftImage Particle.

References

1. N. Nandhakumar and J. K. Aggarwal, "Multisensory Computer Vision", in *Advances in Computers*, **Vol. 34**, Pub Academic Press, 1992
2. N. Nandhakumar and J. K. Aggarwal, "Integrated Analysis of Thermal and Visual Images for Scene Interpretation", *Multisensor Integration and Fusion for Intelligent Machines and Systems*, pp 375-406, Ablex Publishing Corp, NJ, 1995.
3. A. Lanterman, M. Miller and D. Snyder, "Automatic Target Recognition via the Simulation of Infrared Scenes", in *Proc. Of the Sixth Annual Ground Target Modeling and Validation Conference*, Keweenaw Research Center, Michigan Tech. Univ., p. 195-204, August 1995.
4. J. V. Black, "Fusion of Infrared and Visible-Light Images", *Command Information Systems Workshop: 1st CIS Workshop*, pp316-325, December, 1994
5. J. Michael, N. Nandhakumar, T. Saxena and D. Kapur, "Using Elimination Methods to Compute Thermophysical Algebraic Invariants from Infrared Imagery", *Physics Based Modeling Workshop in Computer Vision, IAAA-96*, pp1110-1115, Cambridge, MA, June 18-19, 1995
6. C. Fox, *Active Electro-Optical Systems Handbook*, **Vol. 6**, SPIE Press, January 1993.
7. A. Jelalian, *Laser Radar Systems*, Artech House, London, 1992.
8. A. Sheffer and F. Thompson, "Simulation of Laser Radar Imagery", in *Thermal Imaging; Proceedings of the Meeting, Orlando*, **Vol. 636**, pp92-95, April 1986.
9. J. Kostakis, M. Cooper, T. Green, M. Miller, J. O'Sullivan, J. Shapiro and Donald Snyder, "Multispectral Sensor Fusion for Ground-Based Target Orientation Estimation: FLIR, LADAR, HRR", *SPIE conference on Automatic Target Recognition IX*, **Vol. 3717**, pp 14-24, April 1999.
10. J. Bevington and K. Siejko, "Ladar Sensor Modeling and Image Synthesis for ATR Algorithm Development", *Automatic Object Recognition IV, Proc of SPIE*, **Vol. 2756**, pp 62-75, May 1996.
11. J. Tourtellott, C. F. Coker and D. R. Crow, "Simulation of plume dynamics using particle graphics", *SPIE conference on Technologies for Synthetic Environments: Hardware-in-the-Loop Testing V*, **Vol. 4027**, April 2000.
12. A. J. Cantle, M. Devlin, E. Lord, R. Chamberlain, "High-frame-rate low-latency hardware-in-the-loop image generation: an illustration of the particle method and DIME", *SPIE conference on Technologies for Synthetic Environments: Hardware-in-the-Loop Testing V*, **Vol. 4027**, April 2000.
13. W. P. Jones, *Air Conditioning Engineering*, Edward Arnold, London, 1973.
14. F. Incopera and D. DeWitt, *Fundamentals of Heat and Mass Transfer*, Wiley, 2nd Edition, 1985.
15. W. Press, *Numerical Recipes in C: The Art of Scientific Computing*, Cambridge University Press, 1993.
16. J. Ross Beveridge, Durga P. Panda, Theodore Yachick, *November 1993 Fort Carson RSTA Data Collection Final Report*, http://www.cs.colostate.edu/~vision/ft_carson/index.html, 1993.