A USEFUL KERNEL TO MAKE REALISTIC INFRARED SIMULATION

Jean LATGER
OKTAL company
2, Boudeville road
31 100 Toulouse FRANCE

Jean François TALARON
DGA/DCE/ETC4/ETBS
Gueret road
18 015 Bourges FRANCE

Abstract :

Both industrials and French government services are inclined to using realistic sensor simulation models in the definition and the qualification of the weapon systems.
In the field of sensor simulation, an important domain is the infrared spectrum.
ONDE is precisely a simulation tool for the modeling of the battlefield « seen » by an infrared sensor. ONDE has been intensively used since the beginning of 97 for furtivity studies of ground vehicles and for characterization works of pyrotechnic counter measures.
ONDE is based on a generic kernel consisting of efficient ray tracing functionalities. This tool possesses original capabilities : computation time is nearly independent on the scene complexity and the number of polygons, databases are enhanced by precise physical and thermal data, the ray casting is linked off line with specific software simulating meteo and environment effects (LOWTRAN, EOSAEL), special mechanisms of antialiasing have been developed that enable to take into account very accurate details in the field of view, a generalization of texture definition allows to simulate directional dependence of the emissivity and reflection factors, specific categories of objects are characterized such as 3D clouds, obscurants and flares (IR decoys).
The approach is quite generic so very independent from the sensor. As a consequence, this product is now being transformed into a generic kernel for transverse applications of simulation.
Besides, this kernel is evolving from infrared spectrum to millimetric radar spectrum which doesn't concern this paper's topic.
To sum up, this kernel obeys to several important requirements :

- Be the more physical and realistic as possible, in any case be as realistic as the sensor model
- Be the more generic and independent from the sensor model
- Be the more independent from satellite software e.g. atmospheric propagation models
- Be the less time consuming to afford very complex and high resolution 3D databases.

INTRODUCTION

Due to the increasing cost of validation and qualification studies concerning the current and future weapons systems, both government services and industrials companies deal with simulation.

Besides, increasing computation power of workstations today available opens up new horizons in the research simulation field, and this is particularly obvious with intelligent weapon systems.

Intelligent systems distinguish themselves by the accurateness of the perception and the signal processing devices associated. In this case, realistic simulation is quite convenient.

ONDE, SPECRAY, KERNEL

ONDE is a numerical tool useful to simulate realistically the environment. This tool is being used by both ETBS government study center and ETAS government study center.

ONDE is a first step in the French DGA unification quest.

ONDE is generic enough to satisfy several needs.

ONDE is a complete workshop including standard tools developed by OKTAL and specific modules specified by the government centers.

ONDE enables to create a virtual 3D battlefield and then to simulate various sensor's perception of this theater.

The core of ONDE is the ray tracing based simulation. This simulation is achieved by SPECRAY. SPECRAY is the generic OKTAL's ray tracing.

SPECRAY presents two advantages very important regard to ONDE :

1. The time performances don't strongly depend on scene complexity,
2. The interaction model is open and easy to customize for different wave bands.

The idea of the « kernel » comes from the observation that many government centers and industrial companies share very similar needs. Actually the needs are not identical but are based on a common kernel. The « kernel » is a response to these needs. It's a means to federate within the same toolkit various needs concerning multi sensor simulation.

The « kernel » is a library that enables to build specific simulation application though sharing the same basic functions.

The idea resemble ONDE's approach. But there is two main differences :

1. ONDE is specific to ETBS and ETAS's requirements,
2. ONDE is not a library used to design multi-sensor simulation applications.

The « kernel » is a key component that enables industrial companies and government services to focus on their own specialties. This is coherent with new French DGA reorganization's settlements.

EXAMPLE OF NEEDS SATISFACTION (ONDE)

The problem tackled by ONDE is the evaluation of the weapon system concerning the detection functionality, the system being in development or being in study.

There are three main domain of investigation :

1. « intelligent » ammunitions especially missiles,
2. ground vehicle furtivity,
3. gunnery systems with infrared sensor.

Global simulation need several components for instance shell cinematic, infrared sensor, embedded radar, atmospheric propagation, heat transfer.

ONDE is a specialized brick of a global simulation system concerning both virtual 3D terrain creation and infrared sensor associated perception.

NEEDS SATISFACTION (KERNEL)

The KERNEL's objectives are nearly similar to general ONDE's objectives. The first main difference comes from the generic character of the kernel. The KERNEL can be useful both for ground to ground, ground to air, air to air, air to ground and marine applications. The weapon system can be either a ground vehicle, an aircraft, an helicopter, a ship, a missile or a satellite.
The second main difference is that the KERNEL is a library or an API that enables to develop transverse applications of simulation.

ONDE MAIN FUNCTIONALITIES

ONDE is divided into three parts :

1. 3D database creation tools (including physical data enhancements),
2. 3D database interactive observation to prepare and check the scenario,
3. Realistic image simulation of visible and infrared sensor following the scenario computed by ray tracing.

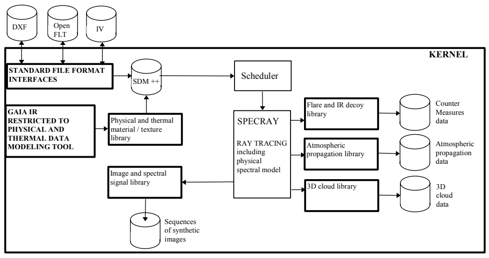
ONDE is devoted to aliment varied numeric models of sensor, from the visible wave band to the far infrared spectrum. The spectral domain spreads from 0,3 µm to 30 µm.

MAIN FUNCTIONALITIES OF THE KERNEL

The KERNEL doesn't include 3D database modeling tools. Indeed many customer already possess their own tools and the goal for the KERNEL is to interface to them.
As a consequence, the KERNEL includes a set of standard file format interfaces that enables to plug the more external tools as possible.
Nevertheless, the KERNEL includes a restricted modeling tool available to create or modify both the physical and spectral data and the thermal data.
The core consists in the ray tracing software and the physical model.
Besides the KERNEL includes several complementary libraries :

- the « physical and thermal material / texture » library which is specialized in the manipulation of database extensions to physics,
- the «atmospheric propagation » library whose aim is to compute for each ray the atmospheric phenomena,
- the « flare and decoy » library whose basic functionalities concern flare and IR decoys representation,
- the « 3D cloud » library useful to represent 3D clouds with voxels (volume elements),
- the « image and spectral signal » library which is a means to manipulate spectral images.

Finally the KERNEL's frame computation rate is controlled by a scheduler.
SDM is the proprietary OKTAL's 3D database file format when SDM++ is the extended OKTAL's file format (with extensions for physical data).



REQUIREMENTS FOR BOTH ONDE AND THE KERNEL

COMPLEXITY OF THE 3D GEOMETRIC DATABASE

The complexity of current 3D virtual databases regularly increases, and this for at least two reasons :

- the spatial resolutions requirement is more and more accurate which means that 3D databases must be enhanced with further details,

- the scene geographical size regard to the domain of application (missiles, aircrafts ...) is more and more important.

As a consequence, the ray tracing performance concerning computation time mustn't depend strongly on the scene complexity or the number of polygon and textures

QUALITY OF THE SPECTRAL PHYSICAL DATA

Even if it is time consuming, the main priority concerns the accurateness of the representation regard to physics. ONDE's core and the KERNEL appears as a « transformer » of physical data. The idea is that this « transformation » must introduce an error as negligible as possible, even if the input data are not generally very accurate.

Moreover, the error has to be precisely quantified.

Indeed, a simulation whose accuracy is unknown in terms of error due to simplification is no use.

GENERICITY

The solution must be generic and this at several levels :

- the wave band ; it means that the model can be tuned from visible to far infrared in the near term, then to radar and millimetric waves in long term,
- the numerical sensor model ; it means that radiance is intrinsic, so that the same set of images can be used for different sensors,
- the satellite softwares ; it means that satellite software concerning atmospheric propagation data computation, IR decoy data computation ... can be exchanged. For instance, LOWTRAN or MODTRAN ought to be used indifferently to precompute atmospheric propagation data.

REAL TIME / NON REAL TIME DUALITY

The duality between Real Time and Non Real Time simulation means that the same 3D database enhanced with the same set of physical data can be manipulated using either RT tools based on Computer Image Generation systems or NRT ray tracing based solution.

Classically RT applications concern training simulation when NRT applications concern research and study simulation. The idea is to merge RT and NRT approach.

There are many interests in doing this :

- first, sharing the database significantly decreases the cost,
- sharing the database solve any problem of correlation between RT and NRT applications,
- RT performance is useful for NRT applications
- finally, NRT accurateness is useful to validate and calibrate RT simulation.

DATABASE EXTENSION MECHANISM

Database extension is the key mechanism to perform physical simulation.

An extension is a set of complementary attributes associated at each nodes of the database. It can affect polygon level, object level, material level or texture level.

The implementation must obey to the following rule :

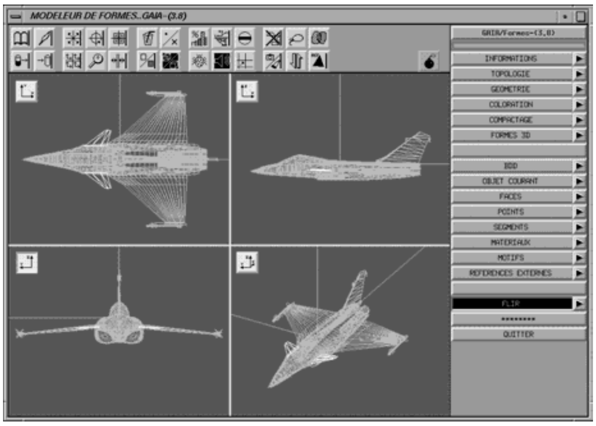
when a modeling function act on an entity enhanced by an extension (for instance splitting a polygon with an extension in two parts), the extension must be automatically inherited by the result of the acting function, without writing any source code (for instance the two parts automatically possess the extension).

This key mechanism is implemented by SDM++.

CREATION OF THE 3D DATABASES

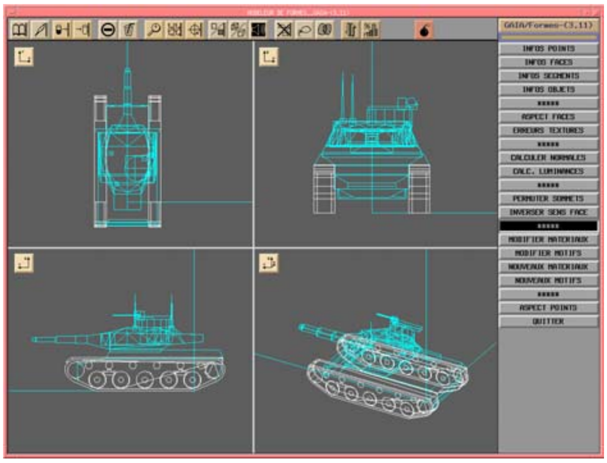
OBJECTS (ONDE)

ONDE 3D objects are designed and modified using the OKTAL's GAIA modeling tool. This tool is based on X11 :

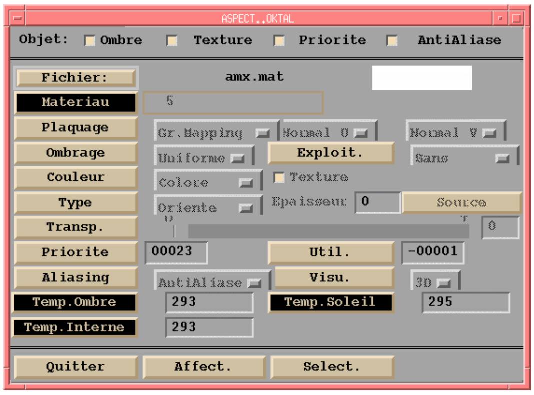


PHYSICAL ATTRIBUTES

The physical attributes can be modified using the infrared restricted modeling tool.

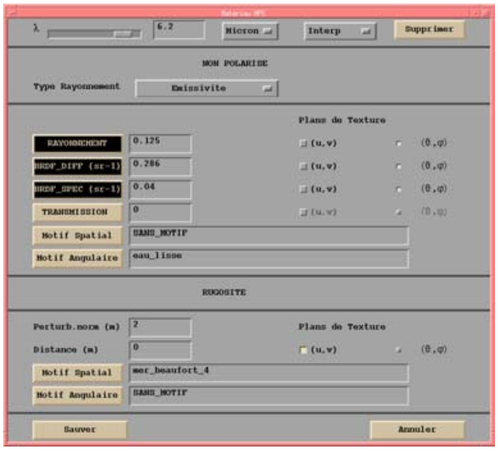


For each polygon, temperature attributes can be modified :



Each material can be enhanced with thermal complementary information :

Besides, each material can be completed with spectral physical information :



This last window enables to specify nominal values, wavelength by wavelength. Besides each nominal value can be enhanced first using spatial texture (u,v) then directional texture (θ,ϕ).

DECOY DATA (KERNEL)

The KERNEL's models are generic.
They are very open ; the interface is very simple data files whose format is public.
The KERNEL possesses 3 basic models :

- Simple light source (hot point)
- Flare
- Cloud

Flare is a complex model defined by :

1. a law of irradiance useful to simulate the scene lighting of the 3D scene
   irradiance depends on :
- wavelength
- elevation observation angle
- azimuth observation angle
- distance
- time

2. a law of radiance useful to simulate the direct perception of the flare
   radiance depends on :
- wavelength
- elevation observation angle
- azimuth observation angle
- distance
- time

The scaling of all these data is arbitrary : the ray tracing automatically makes the interpolations.
Elementary flare is considered as a generic cylinder. The flare description include the variation law in time of both the radius and the height of this cylinder.

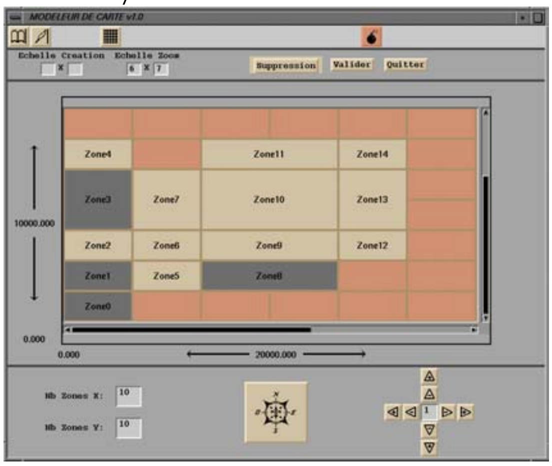
CLOUD DATA (KERNEL)

Cloud is a complex 3D model defined by a set of regular voxels. Each voxel is determined for each date, like the flares.
One voxel includes equivalent spectral emissivity and spectral transmission.
This model is very powerful because it is a real 3D set of voxels. Typically it means that the interaction with a target within the cloud is accurately simulated.
Cloud model can be applied to :

- counter measure smoke or obscurant
- dust cloud
- aircraft or helicopter plume
- ground vehicle plume

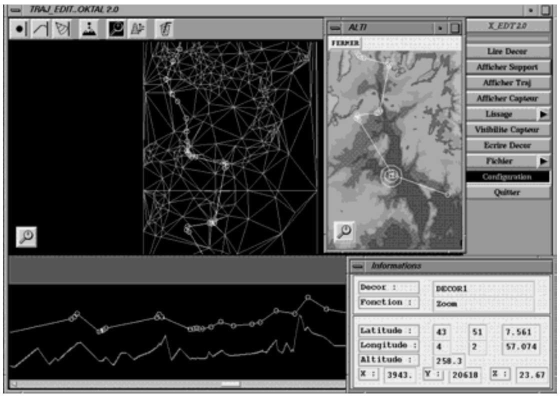
TERRAIN (ONDE)

ONDE 3D terrains are designed and tuned using the OKTAL's GAIA modeling tool. This tool is a geographic data compiler whose originality is concentrated on the feature treatments, for instance tracing 3D roads in the terrain, assembling crossroads, making the trees grow naturally ...
Available geographic data in ONDE scope is DMA/DTED for the altimetry and DMA/DFAD for the features.



PATHS (ONDE)

ONDE moving objects paths are created and tuned using the OKTAL's X-EDT modeling tool. This tool is based on X11.
This approach is available for making objects move when their behavior is fixed and don't depend on the simulation's events.



DATABASE OBSERVATION IN REAL TIME (ONDE)

Database observation in real time refers to database interactive observation to prepare and check the scenario. The scenarios are very complex (number of polygons, number of moving objects, number of decoys, number of images). Computing a scenario using ray tracing can be very long. So it is very important to closely check the scenario before achieving image computation. The best means to do it is to use real time tools.
To do this the ONDE’s scenario can be created, modified or simply checked using a special GUI based on real time image generation. The image computation is then performed using Open GL on a specific hardware platform.

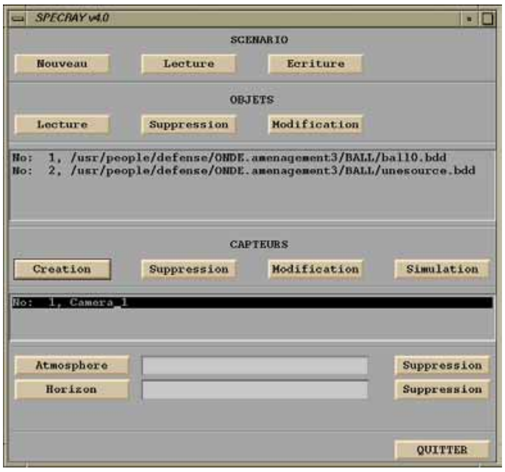
REALISTIC IMAGERY SEQUENCE COMPUTATION

GENERALITIES ABOUT RAY TRACING

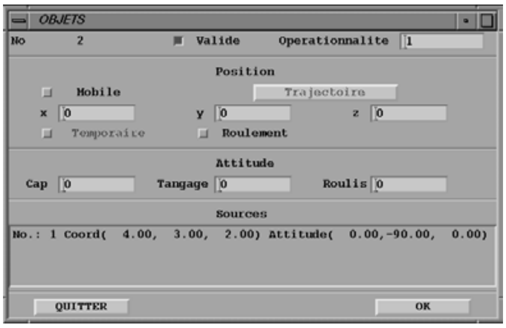
Ray tracing is a particular algorithm used to compute synthetic images. The classical Z-buffer algorithm consists in identifying a display list of polygons included in the viewing frustum and then make several treatments polygon by polygon. First perspective transformation (3D to 2D) then computation of constant data by polygon (Gouraud shading and texture mapping) finally hidden surface removal. This algorithm is classically used in Computer Image Generator.
In an other way, ray tracing algorithm consists in tracing rays from the observer to the 3D scene. The rays are distributed regularly (one by pixel) or irregularly (adaptative antialiasing) inside the viewing frustum. They are called primary rays . The shortest ranged intersection of each primary ray with a 3D surface, for instance a 3D polygon, is then computed. For each intersection point, a lighting ray is traced to each light sources, for instance the sun, which is the means to automatically compute the shadows. A secondary ray is traced from the first intersection point to the second intersection point, which is the means to automatically compute the specular reflection effects

GENERALITIES ABOUT SPECRAY

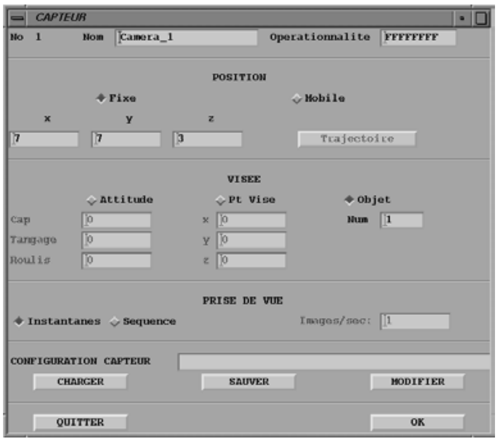
SPECRAY is the ray tracing kernel developed by OKTAL/Defence Department company which enables to compute high realism images in the infrared spectrum.
The main window of SPECRAY is useful to process scenarios. The scenario can be created using SPECRAY by defining a 3D database, a set of moving objets and associated paths, a set of sensors in the scene, a definition file of the atmospheric propagation data (including solar and sky radiance), and a definition file of the horizon/sky model.



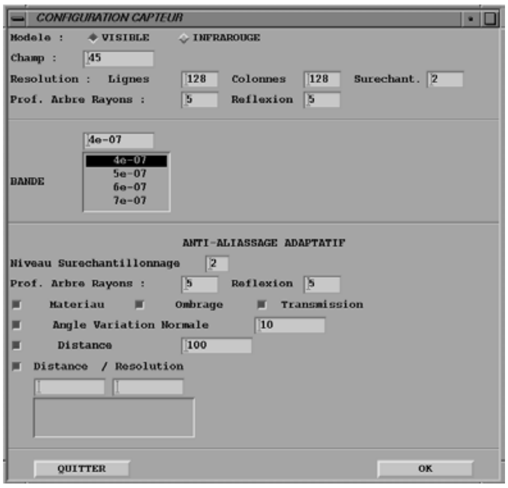
The object panel enables to define the position and Euler angles of each object when static, or the associated path when dynamic (moving objects). The object can be assigned to automatically follow the terrain altimetry using the « Roulement » button. This panel also allows to define light sources.



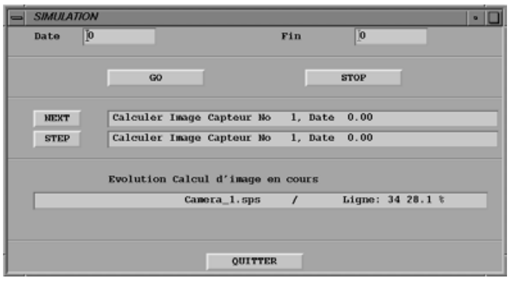
The sensor panel enables to define the position and view point of each sensor. The sensor is either static or dynamic it's to say linked to a path. Besides, the line of sight of each sensor can be attached independently to any objects of the simulation, even when moving along a path.



The sensor configuration panel enables to precisely define the sensor, of course as independently as possible of the sensor model itself, it's to say the more generically as possible. The user can define the sensor basic resolution (number of pixels), the maximum oversampling factor necessary for adaptative antialiasing, the reflection maximum depth for specular materials. Each sensor possesses its own dependence on wavelength which is characterized by an irregular scale of wavelength corresponding to the decomposition of the spectral band of the sensor into several under bands. A set of parameters can be modify which are used by SPECRAY to characterize the adaptative antialiasing algorithms. For instance the user can introduce a minimum angle for the normal vector of a surface between two basic rays so that a new ray can be traced by SPECRAY in case of high variations of the normal vector.



Finally one image or a set of images can be automatically computed using SPECRAY. Mastery of time is complete.



SPECTRAL AND PHYSICAL MODEL

The great originality of SPECRAY comes from the model based on physics. SPECRAY uses elementary pyramids defined by four adjacent rays (one basic pixel) which allows to compute elementary surfaces and solid angles. Besides SPECRAY takes into account the wavelength sampling. Actually SPECRAY works wavelength by wavelength.

A list of N+1 wavelength λ_0, λ_1, … λ_N is attached to the sensor configuration file.

For each pixel i,j SPECRAY computes N values of radiance : $L[\lambda_1][i][j]$, $L[\lambda_2][i][j]$, … $L[\lambda_N][i][j]$, where $L[\lambda_k][i][j]$ is the radiance of pixel i,j for wavelength restricted to λ_{k-1} and λ_k.

As a consequence, there is no discontinuity between visible and infrared spectrum.

The minimum spectral definition in visible spectrum is :

$\varepsilon(\lambda_{red})$, $BRDF_d(\lambda_{red})$, $BRDF_s(\lambda_{red})$

$\varepsilon(\lambda_{green})$, $BRDF_d(\lambda_{green})$, $BRDF_s(\lambda_{green})$

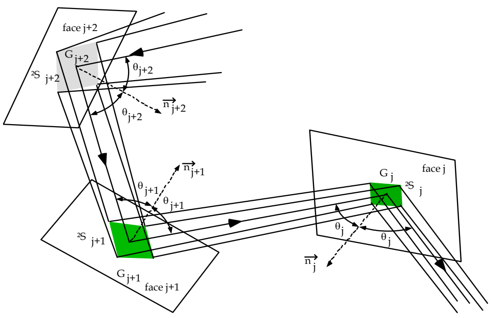
$\varepsilon(\lambda_{blue})$, $BRDF_d(\lambda_{blue})$, $BRDF_s(\lambda_{blue})$

The standard recurrence of SPECRAY is illustrated in the following figure.

G_j is the current intersection between the 3D scene and a secondary ray (depth j).

G_{j+1} is the current intersection between the 3D scene and a secondary ray (depth j+1).

G_{j+2} is the current intersection between the 3D scene and a secondary ray (depth j+2).



L_{j+2} is the luminance emitted by polygon j+2, reflected by polygon j+1 to polygon j.

The relationship is :

$$L_{j+1}(\lambda) = L_{j+2}(\lambda)\cdot \tau[G_{j+1}, G_{j+2}]\cdot BRDF(\theta_{j+1}, \varphi_{j+1}, \theta_{j+1}, \varphi_{j+1}+\pi)\cdot \cos\left(\theta_{j+1}\right)\cdot \Delta\omega_i^{j+1}$$

where $\tau[G_{j+1}, G_{j+2}]$ is the transmission coefficient between G_{j+1} and G_{j+2}.

SPATIAL SUBDIVISION AND PERFORMANCE

Time consumption is very optimized using SPECRAY. Actually performances are nearly independent on scene complexity. To do this SPECRAY uses a spatial subdivision method which enables to get a perfect knowledge of the scene topology before computing the first image. Except for moving objects, which possess a special treatment, this topology is static and available till the database don't change. Scene space is decomposed in a hierarchy of volume elements (voxels) which both contain the list of inner objects and topological relations with the other voxels. Space scene is so turned into a recursive space of voxels which improves efficiently the intersection computations.

ANTIALIASING ON GEOMETRY

Antialiasing acceptation is different in physics than in imagery. The etymology of the word antialiasing conveys the idea of fighting against something strange and alien in the image that could be called an artifact.

The first artifact is simply due to a sampling problem which is obvious when you observe a straight polygon edge onto a pixel grid.

A more subtle artifact occurs when a polygon in the image becomes smaller than the pixel, for instance due to the distance. This is the more important artifact concerning the physical aspects.

In any case the solution to improve image quality mainly consists in over sampling by tracing more rays. The best method, regard to physical requirements, is the adaptative one. The idea is that the screen density of rays is proportional to the local screen complexity. For instance a few rays are traced to the uniform sky when many rays are traced to a target, the screen size of which is very little, though its energy (hot point) is very important.

The most important antialiasing criteria are the following :

- number of different polygons « in the pixel »
- number of different materials
- normal vector variation within the pixel.

TEXTURE AND RADIOMETRIC ANTIALIASING

SPECRAY implements an original generalization of texture mapping.

Classically, a texture is a sort of photo T(u,v) being mapped onto a polygon that creates a color spatial modulation of the polygon better than an uniform color. Texture mapping is a very clever mechanism to artificially improve the scene radiometric complexity.

Two main ideas have been arisen :

- generalization of texture definition to any physical data (emissivity, BRDF, radiance ...)
- generalization of spatial u,v modulation using texture to directional θ,ϕ modulation using texture

This last generalization of texture is very important to take into account the dependence on incidence and reflection angles of the physical materials. As a consequence SPECRAY can simulate the variation of specular reflection factor with the observation angles. For instance using SPECRAY a material can be diffuse for normal incidences and quite specular for tangential incidences.

Concerning antialiasing with texture, the problem origin is the same as geometry.

The problem is the sampling of a texel (texture element) grid by a pixel grid. The problem is complex mainly because the grid planes have different orientation. OKTAL has developed an improvement of the mip mapping method (classically used in Computer Image Generators). The advantage of this method is obvious when the texel is distorted much more in one direction (for instance the u direction) than in the other direction (for instance the v direction), which contributes to the high quality level of the images.

BUMP MAPPING

Bump mapping is a clever derivation from texture that allows to use the spatial texture modulation to modulate the reflected direction of each ray.

On a flat polygon every 3D point shares the same normal vector.

Using bump mapping, each 3D point normal vector is modulated directly by the texture value.

This effect isn't time consuming.

Bump mapping enables to simulate material's roughness.

For instance bump mapping is very useful to distort the specular reflection effects (on water especially).

HYPERTEXTURES (KERNEL)

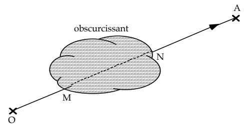
Hypertexture is a generalization in 3D of texture.

It's much more time consuming than simple bump mapping. The difference is that the surface geometry is directly modulated by the T(u,v,w) texture. Using the kernel, each polygon in the database can be enhanced by a double layer which produces an artificial thickness to the flat surfaces of the 3D scene. Within this layer, the geometry is modulated by an analytic expression of the 3D texture. This mechanism is especially useful to simulate dynamic waves on the sea for instance.

3D CLOUD REPRESENTATION (KERNEL)

3D voxels clutters called « clouds » can be defined using external data file. Each voxel is characterized by a self emission component and a transmission component. These data are spectral depending and time depending to simulate the cloud's expansion. When a ray crosses a voxel, self emissivity and transmission of the voxel is computed taking into account the optical range within the voxel. This 3D voxel data can be very easily computed by an external software because the format is open.

This mechanism enables to simulate varied types of clouds : plumes of the vehicles (targets), dust clouds, obscurants and counter measures.



FIXED VIEW POINT IMAGERY (KERNEL)

An interesting optimization of the ray tracing occurs when the Line Of Sight is static.

In this case, everything is constant from one image to the next one, except the targets and moving objects. The idea is to compute only the changes of the images, it's to say a rectangle of pixels (including shadows) bounded to each target.

This presents a double advantage.

- less time consuming
- less memory consuming.

SPECIAL FUNCTIONS

Agenda

The image computation is controlled by a scheduler that simulates time evolution during the simulation exercise. This scheduler manages an agenda which precises at each frame, the position and viewing LOS of each sensor and the position and orientation of each object (taking into account the target's paths).

Telemetry

Telemetry is a means to get the range of each 3D object point from the observer.

Terrain following

When a moving object or a sensor is assigned to follow the terrain (ground vehicle), pitch and roll angles are computed automatically to stick the object onto the terrain. Heading is not modify.

This is performed using a typical vertical telemetry.

Object identification (KERNEL)

Each object can be assigned a particular identifier. This is quite useful when you make picking request in the image. Using object identification function, each pixel is characterized by its own identification flag.

PHYSICAL INFRARED MODEL

Self emissivity

Self emissivity is fundamental in the infrared domain.

Self emissivity can be characterized differently :

- using temperature T and emissivity ε
- using radiance L
- using irradiance law (diffuse sources)
- using intensity law (light sources)

The main characterization is expressed by the Black Body Law or Plank's law (CN) :

$$L'_e(\lambda, T, \theta, \varphi) = \varepsilon(\lambda, \theta, \varphi) \cdot L'_{CN}(\lambda, T)$$

where

$$L'_{CN}(\lambda, T) = \frac{dL_{CN}(\lambda, T)}{d\lambda} = \frac{2hc^2 \cdot \lambda^{-5}}{e^{\frac{hc}{\lambda kT}} - 1}$$

Diffuse and specular reflections

The flux reflected by a surface is :

$$d\Phi_r(\lambda,\theta_r,\varphi_r)=\rho(\lambda,\theta_i,\varphi_i,\theta_r,\varphi_r)d\Phi_i(\lambda,\theta_i,\varphi_i)$$

where

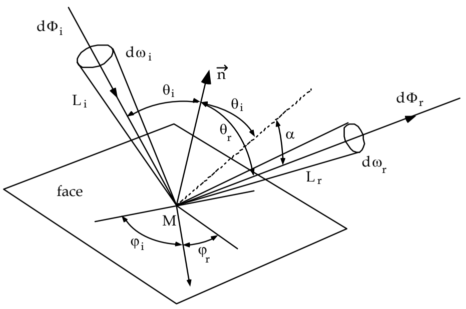
$d\Phi_i(\lambda,\theta_i,\varphi_i)$ is the incident flux

$d\Phi_r(\lambda,\theta_r,\varphi_r)$ is the reflected flux

$\rho(\lambda,\theta_i,\varphi_i,\theta_r,\varphi_r)$ is the bi-directional reflection factor

θ_i, φ_i are the incident angles

θ_r, φ_r are the angles of the observation direction



The BRDF (Bi-directional Reflectance Distribution Function) is the only serious data that can be used in physical simulation. It is defined as :

$$BRDF(\lambda,\theta_i,\varphi_i,\theta_r,\varphi_r)=\frac{dL_r(\lambda,\theta_r,\varphi_r)}{dE_i(\lambda,\theta_i,\varphi_i)}$$

By definition :

$$dE_i(\lambda,\theta_i,\varphi_i)=L_i(\lambda,\theta_i,\varphi_i)\cdot\cos(\theta_i)\cdot d\omega_i$$

Integration is performed on the solid angle $\Delta\omega_i$ corresponding to the source visibility from the reflection point.

$$L_r(\lambda,\theta_r,\varphi_r) = BRDF(\lambda,\theta_i,\varphi_i,\theta_r,\varphi_r)\cdot L_i(\lambda,\theta_i,\varphi_i)\cdot\cos(\theta_i)\cdot\Delta\omega_i$$

Artificially a diffuse and a specular part can be distinguished :

The diffuse component can be characterized by $BRDF_d(\lambda,\theta_i,\varphi_i)$.

The specular component can be characterized by $BRDF_s(\lambda,\theta_i,\varphi_i,\alpha)$.

This factor depends of the α angle between the ideal specular reflection and the observation direction.

The physical reflection model is based on an automatic function for factorization $BRDF(\lambda,\theta_i,\varphi_i, \theta_r,\varphi_r)$ into $BRDF_d(\lambda,\theta_i,\varphi_i)$ and $BRDF_s(\lambda,\theta_i,\varphi_i,\alpha)$ terms.

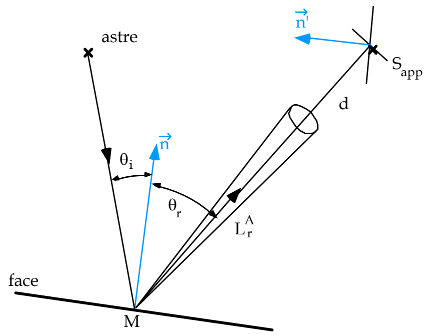
Direct sun lighting

Direct sun or moon lighting takes into account the atmospheric attenuation and diffusion between the astral source and any point M in the 3D scene.

An external data file (typically based on LOWTRAN) contains $E_\perp^A(\lambda,M)$ values for discrete values of the wavelength and of the altitude.

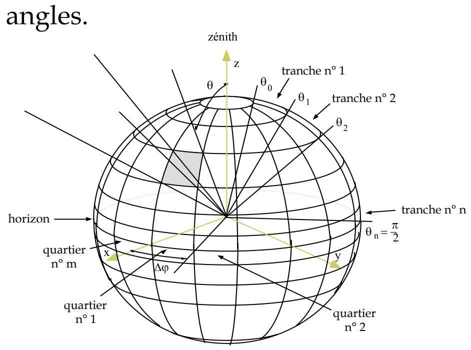
Reflected radiance is :

$$L_r^A(\lambda) = BRDF\left(\lambda,\theta_i,\varphi_i,\theta_r,\varphi_r\right)\cdot\cos(\theta_i)\cdot E_\perp^A(\lambda,M)$$

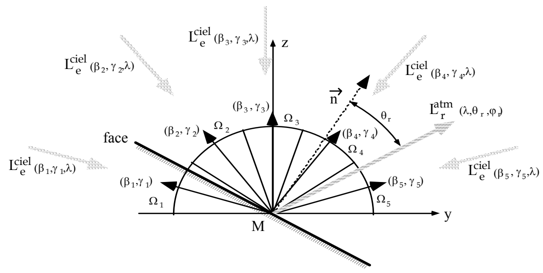


Diffuse sun lightning and sky illumination

Sky is considered as a global entity providing energy in any space direction. When loading the database, the canopy is tessellated in discrete solid angles defined using elevation and azimuth angles.



Each polygon is then computed using this tessellation



If N is the degree of tessellation, the directional reflected energy for each polygon i, due to sky radiance is computed as :

$$L_r^{atm}(\lambda,\theta_r,\varphi_r) = \frac{2\pi}{N}\sum_{\substack{\text{Available directions}\\ \text{for polygon i}}} L_i^{sky}(\lambda,\beta_i,\gamma_i).\cos(\theta_i).BRDF(\lambda_i,\theta_i,\varphi_i,\theta_r,\varphi_r)$$

Self emission of atmosphere

An external data file (typically based on LOWTRAN) contains L_e atmospheric diffusion for discrete values of :

- wavelength
- altitude
- elevation
- azimuth
- range

For each ray, both primary, secondary or lighting ray, the best value of L_e atmospheric diffusion is determined using linear interpolation.

Atmospheric attenuation

An external data file (typically based on LOWTRAN) contains τ atmospheric attenuation for discrete values of :

- wavelength
- altitude
- elevation
- range

For each ray, both primary, secondary or lighting ray, the best value of τ atmospheric attenuation is determined using linear interpolation.

Sky, horizon and cloud cover

Sky and horizon is a pure analytic model.

Horizon model is used to complete the 3D database up to the theoretical horizon range, based on the earth radius and on the observation height.

Sky model is simply computed ,ray by ray, using interpolation of sky luminance stored in external data files (LOWTRAN).

Representing the sky cover is much more complex due to the non homogeneity of depth.

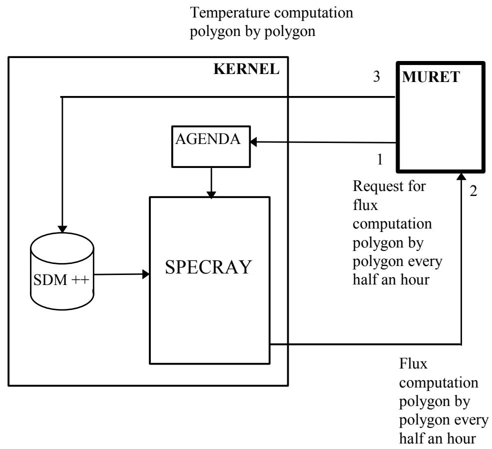
The idea is to merge a spatial u,v modulation and a directional θ,ϕ modulation : $T(u,v).T(\theta,\phi)$.

INTERFACING THE KERNEL WITH THERMAL SOFTWARE

One interest of the KERNEL is to aliment an external thermal software. A thermal software is very important in infrared domain. Indeed, most of radiance values comes from temperature and a thermal software enables to automatically compute surface temperature.

MURET is a particularly compliant to SPECRAY thermal software. This product is a common development shared by French ONERA Research Center and OKTAL.

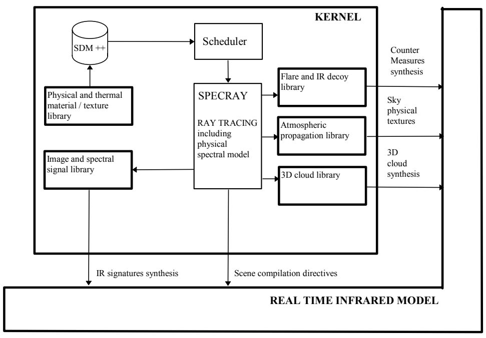
MURET is very accurate specially for the computation of the incident flux mainly due to solar and sky illumination. MURET automatically creates thermal shadows on each polygon. The thermal shadow corresponds to the heat history on a diurnal cycle. These thermal shadows can be achieved easily because it is SPECRAY that computes the incident flux to each 3D point of each surface of the database, taking into account sun, sky and ground visibility (it's to say sun, sky and ground « irradiance ») at each step of the diurnal cycle.



INTERFACING THE KERNEL WITH REAL TIME IR SIMULATION

An other interest of the KERNEL is to aliment a real time infrared simulation model.

Basically most of the entities in the scene are fixed and can profitably been enhanced by ray tracing base computations.
The idea is to divide the scene into fixed (terrain) and mobile objects (targets).
For fixed objects we find both static treatments (self emissivity, diffuse reflection effects, thermal shadows ...) and dynamic treatments (specular reflection effects, atmospheric transmission to the sensor ...).
For dynamic objects the treatments are purely dynamic.
The KERNEL can be a means to take advantage of ray tracing power specifically concerning static treatments, though in real time.
Static treatments can be based directly on the KERNEL algorithms when dynamic treatments can simply be oriented.
Besides, the KERNEL can be very useful to calibrate then validate and qualify the simplification due to the real time model.



POTENTIAL EVOLUTION OF THE KERNEL

The field of evolution is very open.
It can be divided into several axis :

- Global illumination can be improved using radiosity techniques. This means to be able to simulate accurately the thermal radiation, the convection effects and the radiative effects on an object due to the interaction with the surrounding objects.
- Time computation can be highly optimized using parallelism both at image level and pixel level.
- The KERNEL will be enhanced with new functionalities particularly focused on 3D clouds representation and hypertexture.
- A very wide field of improvement concerns the thermal aspects.
- The main evolution concerns radar and telecommunication applications.

KERNEL'S USEFULNESS IN THE LIFE OF THE WEAPON SYSTEM

The KERNEL can be useful at different level in the weapon system cycle life.
First at specification level.
Then at conception level.
Finally at qualification and validation level.
At specification level, using such a model can prevent from making wrong choices it's to say building aberrant sensor solutions on the weapon system.
At conception level the KERNEL can be useful to tune the implementation of specification requirements.
At qualification level, the KERNEL is available to assess the limit performance of the sensor component of the weapon system.

CONCLUSION

Multi sensor simulation is just starting and will certainly be a master piece of technology development. ONDE then the KERNEL are two pioneer steps into this huge field of development. The concept is a winning concept because it is fitted to future natural evolution regard to simulation. The idea is to add other steps in the same direction always keeping in mind the main goal : increase the confidence level to the simulation realism.

REFERENCES

[1] Defence and Optronics symposium '96
« ONDE, a numerical tool for describing environment »
J. LATGER (OKTAL) & JF. TALARON (DGA/ETBS)

[2] Eurographics '94
« Integer logic traversal algorithm for Ray Tracing »
P. PITOT (OKTAL)