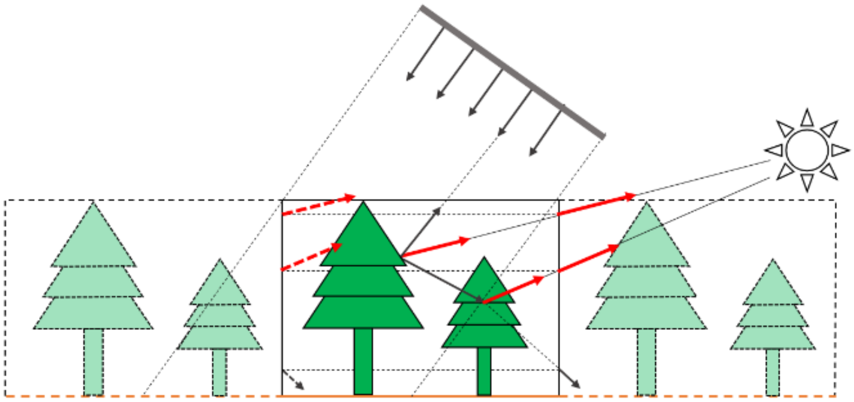
LESS User's Manual

Version 1.8.4



By Jianbo Qi, Yue Xu

September 10, 2018

Table of Contents

1. Introduction

Three-dimensional (3D) radiative transfer (RT) modeling of the transport and interaction of radiation through earth surfaces is a challenging and difficult task. The difficulties lie in the complexity of the landscapes and also in the intensive computational cost of 3D RT simulations. To reduce computation time, current models work with schematic landscapes or with small-scale realistic scenes. The most accurate and efficient models (known as renderers) are from the computer graphics community, but the usages are not straightforward for performing scientific RT simulations. That's the reason why we developed this model.

Basically, LESS employs a forward photon tracing method to simulate bidirectional reflectance factor (BRF) or flux-related data (e.g., downwelling radiation) and a backward path tracing method to generate sensor images (e.g., fisheye images) or large-scale (e.g. 1 km^2) spectral images from visible to thermal infrared spectral domain. We also provide a user-friendly graphic user interface (GUI) and a set of tools are developed to help construct the landscape and set parameters. LESS has already been evaluated with other models in terms of directional BRF and pixel-wise simulated images. It can be used as benchmarks for validating physical models or training artificial neural network (ANN) to do parameter inversion.

2. Fundamentals of LESS

2.1. Forward photon tracing (FPT)

2.1.1. Photon tracing

FPT traces photons from the light sources (e.g., sun). Each photon is represented as a wavelength-specific energy packet $P(\lambda)$. The initial energy $P^0(\lambda)$ of each photon is determined by the power of light sources and the total number of generated photons N. When generating photons in a scene with multiple lights, a light source is randomly chosen according to the importance weights w_k, which is proportional to the power of each light source, i.e., $w_k = \frac{L_k(\lambda)}{\sum_{k=1}^{K} L_k(\lambda)}$ with $L_k(\lambda)$ the power of light k and K the total number of lights. This mechanism guarantees that a light with larger power has more sampled photons. The initial energy of each photon for a randomly chosen light k, in terms of watt (W), is given as

$$P^0(\lambda) = \frac{L_k(\lambda)}{N w_k} \tag{1}$$

When a photon enters the scene along a path defined by its origin and direction of propagation, its intersection with scene elements is tested. If an intersection is found, the energy of this photon will be scaled according to the optical properties of the intersected surface, i.e., the reflectance or transmittance. For a photon with Q times of scattering before it escapes from the scene, the energy of a photon becomes

$$P^Q(\lambda) = P^0(\lambda) \cdot \prod_{q=1}^{Q} \pi f_{BSDF}(q, \lambda) \tag{2}$$

where the $f_{BSDF}(q, \lambda)$ is the bidirectional scattering distribution function (BSDF) for the q^{th} intersection point during its trajectory. Since the scattering law of surfaces in LESS is defined as Lambertian, the BSDF is interpreted as bidirectional reflectance distribution function (BRDF) or bidirectional transmittance distribution function (BTDF), according to the incident photon direction ω_i, surface normal ω_n and outgoing direction ω_o, i.e.,

$$f_{BSDF}(\omega_i, \omega_n, \omega_o) = \frac{1}{\pi} \begin{cases} \rho_{\perp} \operatorname{sgn}(-\omega_n \cdot \omega_i) + \tau \operatorname{sgn}(\omega_n \cdot \omega_i), \text{if } \omega_o \cdot \omega_n \geq 0 \\ \rho_{\top} \operatorname{sgn}(\omega_n \cdot \omega_i) + \tau \operatorname{sgn}(-\omega_n \cdot \omega_i), \text{if } \omega_o \cdot \omega_n < 0 \end{cases}, \operatorname{sgn}(x) = \begin{cases} 1, \text{if } x \geq 0 \\ 0, \text{if } x < 0 \end{cases} \tag{3}$$

where $\frac{\rho_{\top}}{\pi}$ and $\frac{\rho_{\perp}}{\pi}$ is the upper and bottom surface BRDF, respectively. $\frac{\tau}{\pi}$ is the BTDF, which assumes that the transmittances of the surface from both the upper and the bottom side are the same.

The outgoing direction of a photon after scattering is determined by randomly sampling the BSDF function. For Lambertian surfaces, it simply choses a random direction in the outgoing hemisphere. Since the outgoing direction is not wavelength-specific, therefore, the same photon trajectory can be used to compute reflectance for any wavelength. Please note that we have omitted the symbol λ in this equation and following equations for simplicity.

A photon is collected by the sensor if it exits the scene through the top boundary. Lateral boundary effects are considered in order to simulate horizontally infinite scenes with a repetitive pattern. As shown in **Figure 1**, the photon which goes out from the lateral boundaries will re-enter the scene from the opposite side with the same photon direction until the photon escapes through the scene top boundary.

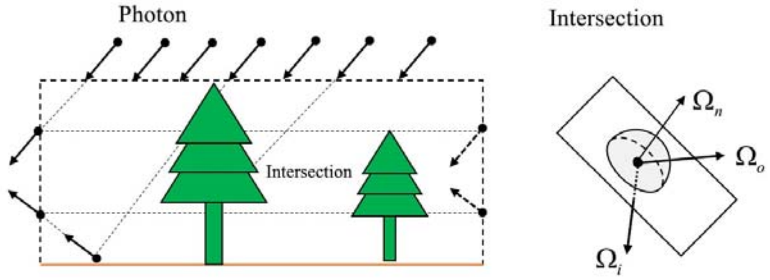


Figure 1. Forward photon tracing.

The collection of the escaped photons is achieved by placing a hemisphere above the scene [1]. As shown in **Figure 2**, the hemisphere is partitioned into small patches with the equal surface area (solid angle) by utilizing the partition scheme of a disk via the equal area projection, i.e., $\Delta\Omega = \Delta S$ [2]. By defining the total number of patches $k_{i-1} = N_P$, we start the partition from zenith angle $\theta_{i-1} = \frac{\pi}{2}$, the next zenith angle θ_i and the total number of patches k_i inside this new zenith angle are defined by

$$\theta_i = \theta_{i-1} - \frac{2}{a_{aspect}} \sin\frac{\theta_{i-1}}{2}\sqrt{\frac{\pi}{k_{i-1}}}, \qquad k_i = k_{i-1}\left(\frac{r_i}{r_{i-1}}\right)^2 \tag{4}$$

where r_i is related to θ_i by $r_i = 2\sin\frac{\theta_i}{2}$ for a unit sphere due to the equal area projection. a_{aspect} is the aspect ratio of each patch, which is approximately enforced to 1. Then the partition in azimuth direction between zenith angle θ_i and θ_{i-1} is given as $\Delta\varphi = \frac{2\pi}{k_{i-1} - k_i}$. After the partition, each patch on the hemisphere is represented as $S_i(\theta_i : \theta_i + \Delta\theta, \varphi_i : \varphi_i + \Delta\varphi)$. The center of this patch is (θ_i^c, φ_i^c)=($\theta_i + \Delta\theta/2, \varphi_i + \Delta\varphi/2$).

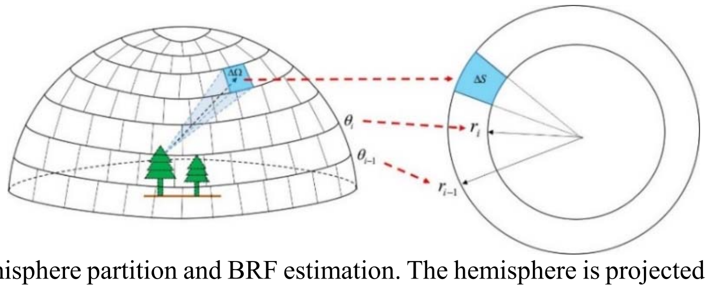


Figure 2. Unit hemisphere partition and BRF estimation. The hemisphere is projected to horizontal plane as a disk using equal area projection. The radius r_i in the disk indicates zenith angle θ_i in the hemisphere.

When a photon exits the scene, the outgoing patch (solid angle) is determined only by the photon direction, i.e., the hemisphere is placed at an infinite position. Each patch on the hemisphere records the cumulated energy of all the incident photons. The BRF in this patch can be estimated by

$$f_{BRF_i} = \frac{\pi P_i^A}{\Delta\Omega_i \cdot \cos\theta_i^c \cdot P_{scene}} \tag{5}$$

where P_i^A is the total cumulated energy (watt) of all the captured photons in patch i, i.e., $P_i^A = \sum_{P^Q \in \Delta\Omega_i} P^Q$; $\Delta\Omega_i = \frac{2\pi}{N_P}$ is the solid angle of each patch; P_{scene} is the cumulated total energy of all the direct incident photons on a reference plane at the top of the scene, i.e., the total incident radiation at the top of the scene. Once the energy of each path is determined, the scene albedo is computed as

$$\omega_{albedo} = \frac{\sum_{i=1}^{N_P} P_i^A}{P_{scene}} \tag{6}$$

2.1.2. Virtual photon

The hemisphere approach (described in section 2.2.1) estimates BRF by using small patches on the sphere. However, even with very small patches, the directions of all the photons captured in a patch are still different [3], which may give incorrect BRF for some specific configurations, such as hot spot. On the other hand, more photons are needed to reduce the variance when smaller patches are used. To solve this problem, a virtual photon approach is introduced, which is similar to the *virtual direction* in DART model [4] or secondary rays in Rayspread model [5]. If a photon is intercepted by an object in the scene without complete absorption, the photon will be scattered in a direction which is randomly sampled by the BSDF function, and a virtual photon is scattered to each of the defined virtual directions. The term "virtual" means that we only calculate the possible energy that can be scattered in these directions by evaluating the BSDF of the intersected surface. It does not change the energy of an actual photon after scattering. Thus it will not break the energy conservation. The possible scattered energy, in terms of intensity ($\mathrm{W \cdot sr^{-1}}$), is calculated as

$$I = V \cdot P^q \cdot f_{BSDF}(q) \cdot \cos < \omega_v, \omega_n > \tag{7}$$

where p^q is the power of the incident photon at intersection q^{th} intersection point along its trajectory; $\vec{d}_v$ is a virtual direction; $\vec{n}$ is the local surface normal; V is a visibility factor which equals to zero if the a landscape element occludes the virtual photon, and equals to 1 otherwise. When sending the occlusion testing rays, the lateral boundary effect is also considered (**Figure 3**). The final BRF is then given as

$$f_{BRF_v} = \frac{\pi I_v^A}{\cos\theta_v \cdot P_{scene}} \tag{8}$$

where I_v^A is the cumulated intensity ($\mathrm{W \cdot sr^{-1}}$) in virtual direction v ; θ_v is the zenith angle of the virtual direction. An advantage of calculating a directional BRF using virtual photons is that the BRF is estimated within an infinity small solid angle, which is the real directional BRF of a scene. Compared to the solid angle approach, it does not fully sample the whole hemisphere, which makes it difficult to calculate the albedo. However, the virtual photon provides an efficient way to oversample some particular directions (e.g., around hot spot), which serves as a beneficial complement. Usually, BRF calculation using virtual photon is more efficient (see section 5.1) than the real photon approach described in section 2.2.1 [6].

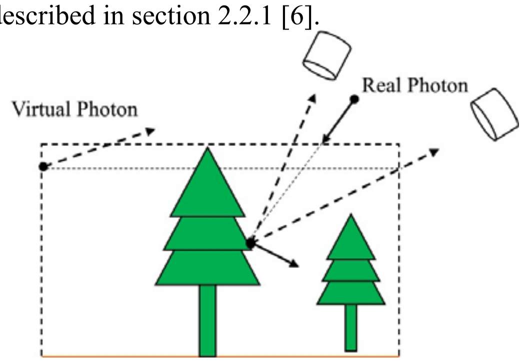


Figure 3. Virtual photon approach to calculate BRF.

2.2. Backward path tracing (BPT)

Instead of tracing photons from light sources, backward path tracing sends rays from sensors into the scene. The ray directions are controlled by sensor configurations (field of view, position, orientation, etc.). The main task of this ray-tracing algorithm is to determine the directional radiance incident onto the sensor and to create the associated image. Radiance along direction ω_o can be calculated according to a rendering equation [7]

$$L_o(q,\omega_o) = L_e(q,\omega_o) + \int_{4\pi} f(q,\omega_i,\omega_o) L_i(p,\omega_i) |\cos\theta_i| \, d\omega_i \qquad (9)$$

where $L_o(q,\omega_o)$ is the outgoing radiance of point q along direction ω_o; $f(q,\omega_i,\omega_o)$ is the BSDF of the intersected surface, which determines the outgoing radiance along direction ω_o at point q induced by incoming radiance along direction ω_i; $L_i(q,\omega_i)$ is the incoming radiance; θ_i is the angle between ω_i and the surface normal at point p; $L_e(q,\omega_o)$ is an emission term (e.g., thermal emission), which is described in detail in section 2.4. The algorithm that solves this equation is illustrated in **Figure 4**. A ray from the sensor intersects the elements in the scene at the point q, then the outgoing radiance induced by the sun (or sky) is calculated according to the BSDF. To calculate the multiple scattering radiation, a new ray is launched from the point q with a direction that samples randomly the BSDF. If this ray intersects the scene at another point q_1, the same procedure is applied to q_1, then the outgoing radiance at point q_1 along the randomly selected direction is the incoming radiance of q along direction ω_1. The multiple scattering procedure is performed recursively until reaching the maximum scattering order (e.g., 5) specified by the user. To prevent energy loss due to termination of scattering, a randomly cut-off technique named "Russian roulette" [8] is applied: after the given number of scattering, a random probability Pr (e.g., 5%) if applied to stop the ray. If the ray is not terminated, its energy is multiplied by $\frac{1}{1-\text{Pr}}$.

When simulating a horizontally infinite scene, the lateral boundary effect is considered for both the sensor ray and the illumination ray. At each intersected point (q_i), an illumination ray, which is built by randomly sampling a point q_e on the emitter, is sent towards the emitter. If this ray traverses the lateral boundary of the scene, it is also reintroduced into the scene to test whether the intersected point is occluded by other landscape elements or not.

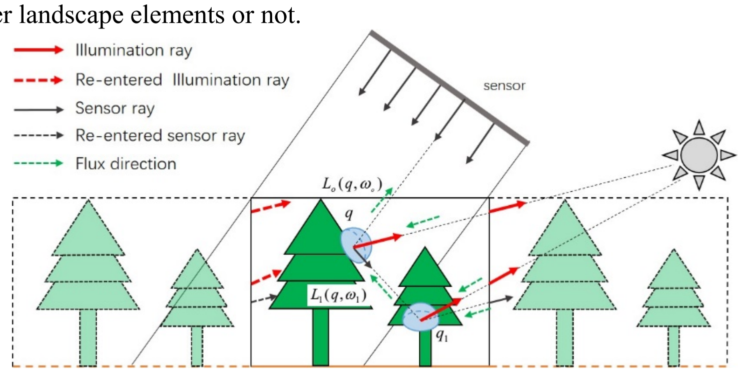


Figure 4. Backward path tracing.

2.3. Simulating thermal infrared radiation

When simulating thermal infrared, the object (e.g., leaves) itself becomes an "emitter", which emits thermal radiation according to Planck's law and its emissivity, instead of the sun. Since the incident energy of sun in thermal infrared bands (e.g., 10 μm) is too weak, we ignore this part of energy in our simulation. However, the presence of sun radiation will greatly influence the temperature distribution of objects by casting shadows between them. This generally classifies the scene elements into four components with specific temperatures, i.e., sunlit soil, shaded soil, sunlit leaves and shaded leaves. The determination of these four components is computed on the fly instead of a precomputing step adopted by most of other forward models (e.g., DART). This on-the-fly approach avoids the storage of emission points, which can greatly reduce the memory usage, especially for scenes with a large number of leaves.

If a sensor ray is intersected in the scene (i.e., q in the scene), an emission term at this point is added. To determine the emission energy, an occlusion ray is traced towards the sun. If this point is directly illuminated, i.e., the occlusion rays intersects nothing, it uses the temperature of the sunlit component. Otherwise it uses the temperature of the shaded component. The emitted radiation is calculated by using Planck law with emissivity (1-absorption) provided. Except for the emission term, another part of the energy that goes into the sensor is the reflected energy, which is emitted by other objects in the scene or sky radiation. In order to consider this part of the energy, a random point on a randomly selected object (including the sky) is sampled (e.g., q_e in **Figure 5**). The emission energy at this sampled point is also determined by sending an occlusion ray. If this point is not occluded through the path to the intersection point q, the contribution of energy from this point is calculated by using the BSDF defined at point q. This procedure can be recursively repeated, which is the same as the procedure described in section 2.2 except for the emission term. Finally, a thermal infrared image, which records the radiance value, can be simulated.

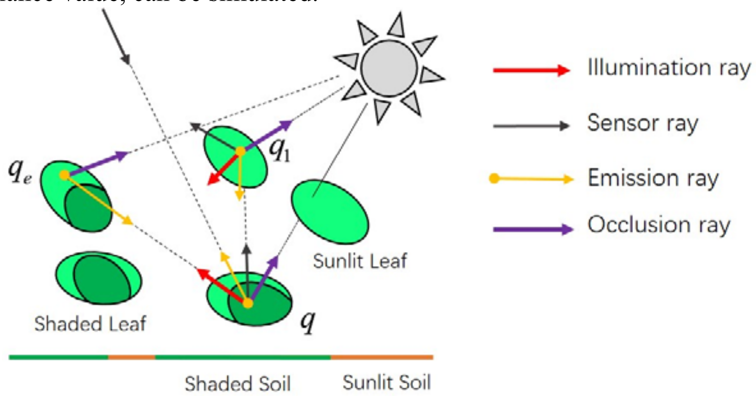


Figure 5. Simulating thermal infrared radiation using backward path tracing.

3. Installation

Download the installer from http://www.lessrt.org. All you need is just to choose a place to install it. (Usually, LESS cannot be installed in c:\program files, since it does not have administrative rights.). After installation, LESS will be automatically started.

4. Graphic User Interface (GUI)

LESS window is divided into four areas (**Figure 6**):

- ✧ The area 1 is Preview Panel
- ✧ The area 2 is Parameter Control Panel
- ✧ The area 3 is Progress Panel
- ✧ The area 4 is Menu Panel

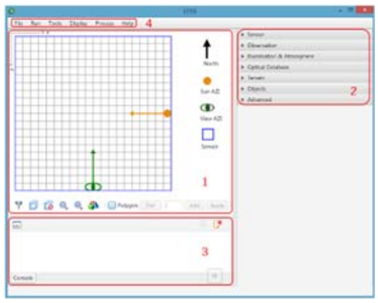


Figure 6. Main Screen of LESS

4.1. Preview Panel

Preview Panel (Panel 1) is for displaying some information, such as view azimuth angle ( ), sun azimuth angle ( ), sensor footprint ( ) and tree positions. This will make it easier for user to set parameters, because this display is automatically updated when related parameters are changing.

Background Image ( ) - When you click the icon, you can choose a picture as background image, make sure that the image represents the same area defined in LESS.

Delete Background Image ( ) - When you click the icon, background image will be delete.

Zoom-in ( ) - This icon allows you to zoom-in the area covered by grids.

Zoom-out ( ) - This icon allows you to zoom-out the area covered by grids.

3D Viewer ( ) - If you click this icon, 3D Viewer will be activated. In 3D Viewer windows, you can visualize the scene in advance (before actual simulation) to verify the 3D scene.

Polygon (Polygon) - When you check the icon, you can draw a polygon in the area covered by grid and set polygon parameter (such as the minimum distance between trees, tree position in the polygon) in the panel after it (Del 3 Add Apply). If you just want to allocate objects in some particular areas, the solution is using Polygon tool. Check the option of Polygon under the display panel. This time when your mouse enters the display panel, it will become a hand. you can create a polygon by left click. If you want to remove the polygon, just use your right click of mouse. After the creation of polygon, you can delete or add object instances in this area. If you click Add, then all the generated instances will only apprear in the polygon, the position and number of each object are still random within the polygon. However, if you choose one of the objects in the Objects List. then the generated instances only contains this object. This tool can create some forest scene which contains one species of trees in some area and another species of tree in another areas.

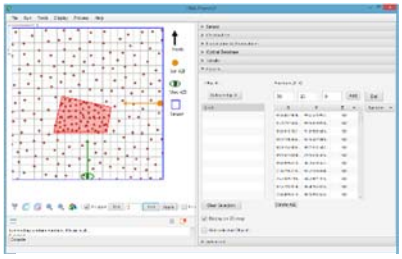


Figure 7. Polygon tools

Point (Point) - When you check the icon, you will see the cursor changing to a hand. You can choose the position of tree by clicking the place where you want the tree to be, without inputting coordinates.

4.2. Parameter Control Panel

Parameter Control Panel (Panel 2) is for setting parameter which can control the scene you simulate.

4.2.1. Sensor

To simulate a sensor you can input parameters here to control image type, the number of pixels in width and height, sample count per pixel.

Type - Up to now, there are four types that you can choose. They are **orthographic, perspective, CircularFisheye, PhotonTracing**.

Width - The number of pixels in width.

Height - The number of pixels in height.

Samples - sample count per square meter. Usually 128 is sufficient. If you increase to 256, 512, the quality of the simulated image will be better.

Spectral Bands - It represents the which band you want to simulate. For example, RED and NIR: 660nm，900nm. Now, you must also input a bandwidth for each band with the format of “center band: bandwidths”, e.g. 660:10,900:10. These bandwidths will be used for determining the irradiance. If you have no special requirement, it can just set the bandwidth to zero. And when you click [Define], you can define the bands by input some parameter in the pop-up window (**Figure 8**).

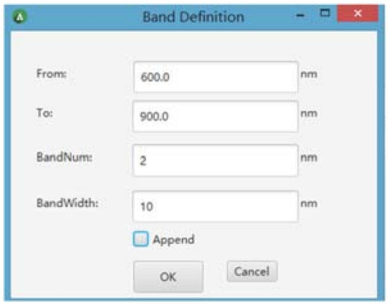


Figure 8. Spectral bands

Image Format - “Spectrum” means generating spectral image. “Synthesized RGB Image” means generating RGB image.

Only First Order? - If it is true, it means LESS only simulate the first-scattering event (sun * radiation reflected by objects only one time). If set to false, then both first-scattering and multi-scatting will be simulated.

Virtual Plane – When you check it, a virtual plane is defined, and only the radiant that exits through the plane is calculated. In **Figure 9**, you can define the location of the virtual plane in “Center” and the size of the virtual plane in “Size”. Usually, the height of the plane (Z) should be left as default, i.e., MAX.

Figure 9. Virtual plane

Thermal Radiation - When you need to get thermal infrared image, you can check it. After checking, you can input surface temperature parameter in Optical Database (**Figure 10**).

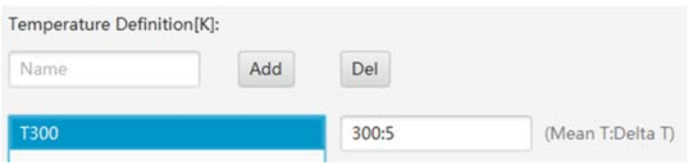


Figure 10. Temperature definition

NoData Value - Set the background value when sensor footprint is

beyond the scene scale. If the image opened by envi, you should set it as 0.
Repetitive scene - Sets the number of copies that distributed around the scene.
Width Extent/Height Extent - The actual extend of the orthographic sensor can simulate. It is similar to FOV of perspective sensor.
Four Components Product - If you check it, a classification image will be generated. The classification image has four components, and they are Light canopy, shadow canopy, light background, shadow background.

4.2.2. Observation

View Zenith - Set view zenith angle (angle between vertical direction).
View Azimuth - Set view azimuth （0° is north, clock-wise to 360°).
Sensor Height - Set the height of the sensor.

4.2.3. Illumination & Atmosphere

There are two groups of illuminations, one is from sun, other is from atmosphere.
Sun Zenith - Zenith angle of sun.
Sun Azimuth - Azimuth angle of sun (0° is north, clock-wise to 360°).
This defines the position of sun. That means if you set it as 90° (East), then it will produce shadows in West.
Sun Position Calculator - If you clicked [Sun Position Calculator], you can fill in above two parameters by inputting the time and place parameters for the scene.
Next is the parameter setting for the sky.
Type - It represents the type of atmosphere radiation, now only one option can be used - “SKY_TO_TOTAL”, it means the ratio between diffuse radiation and total incident radiation. Thus (1-SKYL) * T (T is sun radiation above atmosphere) is the sun radiation under atmosphere. Under this mode, atmosphere is isotropic diffuse radiation from upper hemisphere.
Percentage - It define the actual ratio between atmosphere radiation and total radiation. What should be noticed is that when you input the value, the number of values should be equal to the number of bands (under the sensor section). That is, for each band, it may have different values.
Input solar spectrum manually - If you check it, you can input the solar spectrum and the sky spectral in terms of wavelength manually (in $W/m^2/nm$).

4.2.4. Optical Database

Since objects in LESS are represented as triangular meshes, thus for each triangle, it can have at least three kinds of optical properties: reflectance of front side, reflectance of back side, and transmittance (we assume transmittance of both side is the same). In Optical database, we should first define some optical model, and then they can be used in the following terrain or forest definition. By default, there are three optical model defined, you can delete them, or modified them directly. you can also add new optical model. For each reflectance or transmittance, values of different bands are connected by comma.

4.2.5. Terrain

There are mainly three types of terrain: PLANE, RASTER, MESH.
PLANE just represents simple plane lies at altitude of 0. RASTER is a raster image file in ENVI standard format.
XSize - Size in the X direction.
YSize - Size in the Y direction.
There are two BRDF types: **Lambertian, Soilspect**. If Lambertian is chosen, it means you think of the ground as a Lambertian. If Soilspect is chosen (), you can input parameters to define the model.
Land Cover - If you check it, you can input surface classification data generated by envi, and then set different spectrum for different ground classes.

4.2.6. Objects

Objects define what you want to put in the scene. For example, forest is formed by a number of single trees, which are describe by triangle mesh (obj file) in LESS. Usually, we cannot input a obj for each single trees, since forest contains a lot of trees and the tree itself contains numerous triangles, it may not be possible even for large memory computers. The alternative is to use a “instance” technique. That means we define a single tree, and we can place it at different places, just using reference, thus the program only keeps one copy of the triangle mesh, but it represents trees at different places (they have exactly the same structures, but we can do rigid transformations).

Objects window is divided into three areas:
• The area (1) is Objects Define Panel;
• The area (2) is Position Parameter Control Panel;
• The area (3) is Display Panel. Define Panel

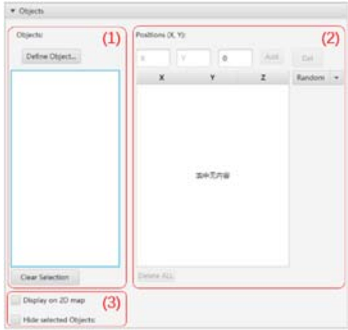


Figure 11. Objects definition

(1) Objects Define Panel
The first step is to define some objects (single tree).
Define Object… - You can click it to define objects. It will open a new dialog that allows to import obj file (**Figure 11**).
Add - We give a name for our first object, such as "birch".After clicking [Add] button, "birch" will appear in the Objects list.
Import OBJ - Selecting the name we write in "Objects" area, then the button [Import OBJ] is activated. If you clicked the button [Import OBJ], you can choose a obj file in the window and input it as the object (**Figure 12**).

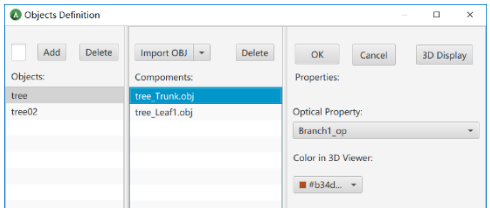


Figure 12. Import obj file

And then you need to determine the scale by the units of the tree model that you enter into. If the units of the tree model that you enter into is cm, the scale should be 0.01. And if the units of the tree model that you enter into is m, the scale should be 1.00.
Import from RAMI - Import objects from the model file on the rami website. After importing objects, each component of the obj file will appear in the "Components" list. Component is a part of the object, e.g. a tree contains leaves and branches (usually they have different optical properties). **You should be aware that the obj file itself will be copied to the simulation folder, so you can safely delete it from your original place.**
Optical Property - Selecting one of the components, the [Optical Property] is activated and you can choose an optical property for the selected component. These optical properties are also from "Optical Database". If you need to define a new optical model, you can go back to "Optical Database" page without closing the current window. When you finish, the optical models are automatically synchronized.
3D Display - After clicking it, you can see the 3D model of object (**Figure 13**).
After defining objects, you should click [OK] button, and then we will go back to Objects page with the defined objects shown in Objects list.
(2) Position Parameter Control Panel
If we select the defined objects it, we can define its positions. There are three methods to define positions. The first method is inputting coordinate for the object directly. The second method is using [Point] tool provided by LESS (introduced in Panel 1). The third method is using [Random] tool provided by LESS. The last method is import CHM data to control position.
Add - You can enter coordinates in the table before [Add] button, and then clicking [Add], the object is added to the specified location.

Figure 13. 3D Display

Random - Clicking the button [Random], it will open a new window, Now the only choice is Poisson distribution (This is very normal for trees in forest). The only parameter you need to provide is the minimum distance between two objects. Click OK, LESS will automatically generate instances of defined objects. The number and position of each object in the Objects list is also random. Thus, you can get a reasonable distribution of objects.
From CHM - Clicking the inverted triangle next to the button [Random], you can open the drop-down menu, and find the [From CHM] button. If click it, you can choose a CHM data as a basis for objects location.
(3) Display Panel
Display on 2D map - If a position is added when you check it, the position will appear on the grid area in Preview Panel, which will help you to check whether it is correct.
Hide selected Objects - If there are multiple plants in this scene, you can hide the location of the model you don't want to see by checking it.

4.3. Progress Panel

This panel shows the running state of the current program. When an error occurs, you can find the reason of the error by reading this panel.

4.4. Menu Panel

Menu bar holds 6 menus: **File, Run, Tools, Display, Process, Help**

4.4.1. File menu

File > New Simulation - Create a new simulation.
File > Open Simulation - Opens a chosen LESS file and loads its parameter values into LESS modeler.
File > Save - Saves current parameter values on a disk as LESS file.
File > Save as - Saves the tree image into another file.
File > Close - Close the simulation.

4.4.2. Run menu

Run > Run All - Run this program and output results.
Run > Generate 3D Model – Generate 3D objects, e.g., convert obj file into binary format.
Run > Generate View & Illumination – Generate viewing parameters for simulation.
Run > less – Run lessrt simulation

4.4.3. Tools menu

Tools > Open Results Folder - Open Results Folder of current simulation.
Tools > Batch Tools – Do batch processing.
Tools > Server Setting – Do network parallel simulation
Tools > LAI Calculator - Calculate LAI in different resolution, and output the results as txt.
Tools > Python Console – run Python scripts (under experiment).

4.4.4. Display menu

Display > 3D Viewer - Display scene you simulate from multiple angles.
Display > 3D Viewer (Bounding BOX) - Display scene you simulate from multiple angles with replacing objects with boxes.
Display > 2D Polygon - Draw a polygon in the area covered by grid and set polygon parameter (such as the minimum distance between trees, tree position in the polygon) in the panel after it.

4.4.5. Process menu

Process > BRF Processing - Generate BRF image.
Process > Brightness Temperature Processing -

4.4.6. Help menu

Help > Documentation – Documentation of LESS.

5. Examples

5.1. Surface reflectance

5.2. Image simulation

5.3. Solar Radiation

6. References

[1] Y. M. Govaerts and M. M. Verstraete, “Raytran: A Monte Carlo ray-tracing model to compute light scattering in three-dimensional heterogeneous media,” *Geoscience and Remote Sensing, IEEE Transactions on*, vol. 36, no. 2, pp. 493–505, 1998.

[2] B. Beckers and P. Beckers, "A general rule for disk and hemisphere partition into equal-area cells," *Computational Geometry*, vol. 45, no. 7, pp. 275–283, Aug. 2012.

[3] M. I. Disney, P. Lewis, and P. North, "Monte Carlo ray tracing in optical canopy reflectance modelling," *Remote Sensing Reviews*, vol. 18, no. 2–4, pp. 163–196, 2000.

[4] T. Yin, N. Lauret, and J.-P. Gastellu-Etchegorry, "Simulating images of passive sensors with finite field of view by coupling 3-D radiative transfer model and sensor perspective projection," *Remote Sensing of Environment*, vol. 162, pp. 169–185, Jun. 2015.

[5] J.-L. Widlowski, T. Lavergne, B. Pinty, M. Verstraete, and N. Gobron, "Rayspread: A virtual laboratory for rapid BRF simulations over 3-D plant canopies," *Computational methods in transport*, pp. 211–231, 2006.

[6] R. L. Thompson and N. S. Goel, "Two models for rapidly calculating bidirectional reflectance of complex vegetation scenes: Photon spread (PS) model and statistical photon spread (SPS) model," *Remote Sensing Reviews*, vol. 16, no. 3, pp. 157–207, Mar. 1998.

[7] J. T. Kajiya, "The rendering equation," in *ACM Siggraph Computer Graphics*, 1986, vol. 20, pp. 143–150.

[8] H. Kobayashi and H. Iwabuchi, "A coupled 1-D atmosphere and 3-D canopy radiative transfer model for canopy reflectance, light environment, and photosynthesis simulation in a heterogeneous landscape," *Remote Sensing of Environment*, vol. 112, no. 1, pp. 173–185, Jan. 2008.