

NAME

rtcontrib - compute contribution coefficients in a RADIANCE scene

SYNOPSIS

```
rtcontrib [ -n nprocs ] [ -V ] [ -c count ] [ -fo | -r ] [ -e expr ] [ -f source ] [ -o ospec ] [ -b binv ] [ -bn nbins ] { -m mod | -M file } [ $EVAR ] [ @file ] [ rtrace options ] octree
rtcontrib [ options ] -defaults
```

DESCRIPTION

Rtcontrib computes ray coefficients for objects whose modifiers are named in one or more *-m* settings. These modifiers are usually materials associated with light sources or sky domes, and must directly modify some geometric primitives to be considered in the output. A modifier list may also be read from a file using the *-M* option. The RAYPATH environment variable determines directories to search for this file. (No search takes place if a file name begins with a '.', '/' or '~' character.)

If the *-n* option is specified with a value greater than 1, multiple *rtrace* processes will be used to accelerate computation on a shared memory machine. Note that there is no benefit to using more processes than there are local CPUs available to do the work, and the *rtcontrib* process itself may use a considerable amount of CPU time.

By setting the boolean *-V* option, you may instruct *rtcontrib* to report the contribution from each material rather than the ray coefficient. This is particularly useful for light sources with directional output distributions, whose value would otherwise be lost in the shuffle. With the default *-V-* setting, the output of *rtcontrib* is a coefficient that must be multiplied by the radiance of each material to arrive at a final contribution. This is more convenient for computing daylight coefficients, or cases where the actual radiance is not desired. Use the *-V+* setting when you wish to simply sum together contributions (with possible adjustment factors) to obtain a final radiance value. Combined with the *-i* or *-I* option, irradiance contributions are reported by *-V+* rather than radiance, and *-V-* coefficients contain an additional factor of PI.

The *-c* option tells *rtcontrib* how many rays to accumulate for each record. The default value is 1, meaning a full record will be produced for each input ray. For values greater than 1, contributions will be averaged together over the given number of input rays. If set to zero, only a single record will be produced at the very end, corresponding to the sum of all rays given on the input (rather than the average). This is equivalent to passing all the output records through a program like *total(1)* to sum RGB values together, but is much more efficient. Using this option, it is possible to reverse sampling, sending rays from a parallel source such as the sun to a diffuse surface, for example. Note that output flushing via zero-direction rays is disabled for accumulated evaluations.

The output of *rtcontrib* has many potential uses. Source contributions can be used as components in linear combination to reproduce any desired variation, e.g., simulating lighting controls or changing sky conditions via daylight coefficients. More generally, *rtcontrib* can be used to compute arbitrary input-output relationships in optical systems, such as luminaires, light pipes, and shading devices.

Rtcontrib calls *rtrace(1)* with the *-oTW* (or *-oTV*) option to calculate the daughter ray contributions for each input ray, and the output tallies are sent to one or more destinations according to the given *-o* specification. If a destination begins with an exclamation mark ('!'), then a pipe is opened to a command and data is sent to its standard input. Otherwise, the destination is treated as a file. An existing file of the same name will not be clobbered, unless the *-fo* option is given. If instead the *-r* option is specified, data recovery is attempted on existing files. (If *-c 0* is used together with the *-r* option, existing files are read in and new ray evaluations are added to the previous results, providing a convenient means for progressive simulation.) If an output specification contains a "%s" format, this will be replaced by the modifier name. The *-b* option may be used to further define a "bin number" within each object if finer resolution is needed, and this will be applied to a "%d" format in the output file specification if present. The actual bin number is computed at run time based on ray direction and surface intersection, as described below. If the number of bins is known in advance, it should be specified with the *-bn* option, and this is critical for output files containing multiple values per record. A variable or constant name may be given for this parameter if it has been defined via a previous *-f* or *-e* option. Since bin numbers start from 0, the bin count is always equal to the last bin plus 1. Set this value to 0 if the bin count is unknown (the default). The most recent *-b*, *-bn* and *-o* options to the left of each *-m* setting are the ones used for that modifier. The ordering of other

options is unimportant, except for $-x$ and $-y$ if the $-c$ is 0, when they control the resolution string produced in the corresponding output.

If a $-b$ expression is defined for a particular modifier, the bin number will be evaluated at run time for each ray contribution from *rtrace*. Specifically, each ray's world intersection point will be assigned to the variables Px, Py, and Pz, and the normalized ray direction will be assigned to Dx, Dy, and Dz. These parameters may be combined with definitions given in $-e$ arguments and files read using the $-f$ option. The computed bin value will be rounded to the nearest whole number. This mechanism allows the user to define precise regions or directions they wish to accumulate, such as the Tregenza sky discretization, which would be otherwise impossible to specify as a set of RADIANCE primitives. The rules and predefined functions available for these expressions are described in the *rcalc(1)* man page. Unlike *rcalc*, *rtcontrib* will search the RADIANCE library directories for each file given in a $-f$ option.

If no $-o$ specification is given, results are written on the standard output in order of modifier (as given on the command line) then bin number. Concatenated data is also sent to a single destination (i.e., an initial $-o$ specification without formatting strings). If a "%s" format appears but no "%d" in the $-o$ specification, then each modifier will have its own output file, with multiple values per record in the case of a non-zero $-b$ definition. If a "%d" format appears but no "%s", then each bin will get its own output file, with modifiers output in order in each record. For text output, each RGB coefficient triple is separated by a tab, with a newline at the end of each ray record. For binary output formats, there is no such delimiter to mark the end of each record.

Input and output format defaults to plain text, where each ray's origin and direction (6 real values) are given on input, and one line is produced per output file per ray. Alternative data representations may be specified by the $-ffio$ option, which is described in the *rtrace* man page along with the associated $-x$ and $-y$ resolution settings. In particular, the color ('c') output data representation together with positive dimensions for $-x$ and $-y$ will produce an uncompressed RADIANCE picture, suitable for manipulation with *pcomb(1)* and related tools.

Options may be given on the command line and/or read from the environment and/or read from a file. A command argument beginning with a dollar sign ('\$') is immediately replaced by the contents of the given environment variable. A command argument beginning with an at sign ('@') is immediately replaced by the contents of the given file.

EXAMPLES

To compute the proportional contributions from sources modified by "light1" vs. "light2" on a set of illuminance values:

```
rtcontrib -I+ @render.opt -o c_%s.dat -m light1 -m light2 scene.oct < test.dat
```

To generate a pair of images corresponding to these two lights' contributions:

```
vwrays -ff -x 1024 -y 1024 -vf best.vf | rtcontrib -ffc 'vwrays -d -x 1024 -y 1024 -vf best.vf' @render.opt -o c_%s.hdr -m light1 -m light2 scene.oct
```

These images may then be recombined using the desired outputs of light1 and light2:

```
pcomb -c 100 90 75 c_light1.hdr -c 50 55 57 c_light2.hdr > combined.hdr
```

To compute an array of illuminance contributions according to a Tregenza sky:

```
rtcontrib -I+ -b tbin -o sky.dat -m skyglow -b 0 -o ground.dat -m groundglow @render.opt -f tregenza.cal scene.oct < test.dat
```

ENVIRONMENT

RAYPATH path to search for $-f$ and $-M$ files

AUTHOR

Greg Ward

SEE ALSO

cnt(1), genklemsamp(1), getinfo(1), pcomb(1), pfilt(1), ra_rgbe(1), rcalc(1), rpict(1), rsensor(1), rtrace(1), total(1), vwrays(1), ximage(1)