

Visualization Insider

Critical V-Ray Settings – Part I

Introduction

To say that V-Ray is a complicated program would be an understatement! Ironically enough, however, it has become the most ubiquitous engine in the visualization industry because of, among other things, its simplicity. Although it contains a large number of settings which should ideally be separated from critical settings and grouped into an advanced section all alone, it is nonetheless a fairly easy program to digest when you know which settings are critical and which can be left for exploration down the road.

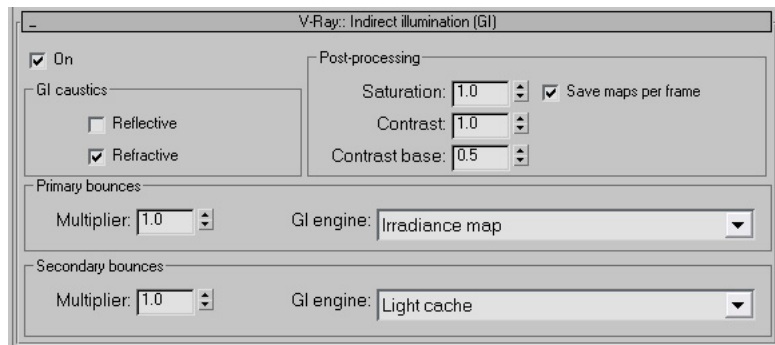
We believe knowing V-Ray means knowing how to troubleshoot your scenes. When you render a scene, you either get a result you want or don't want. If you get what you want, great - but if you don't, and don't know how to fix a problem, such as blotchiness, noise, flickering, etc, then you are bound to spend precious time testing numerous settings. Even when you do get the results you want, you may not realize that you can achieve the same or better results in less rendering time.

Rather than explaining the critical settings in a top to bottom manner, starting with the top-most rollout and ending with the bottom-most, this discussion presents V-Ray features in a more logical manner, from most critical to least critical. For example, there is an option to disable glossy effects in one of the first rollouts, Global Switches; however, it wouldn't make sense to discuss an option for a concept that has yet to be explained. Therefore, such a setting is reserved for later.

For the purpose of this discussion, we are defining critical settings as those which tend to require attention at least once during the course of a typical visualization animation. This means that while a particular setting's use may not be guaranteed during any given visualization, its use will probably have been at least considered as an option because of its special qualities or characteristics that it offers a user. By the same token, having not listed a setting does not at all mean that it is not something worth exploring, but rather it is a setting most likely not as widely used, as unique in its affect, or as practical in a production environment. At the end of Part II of this discussion are a few 'gee-whiz' type features that may not be as practical as those presented earlier, but are nonetheless, interesting enough to make mention.

Indirect Illumination

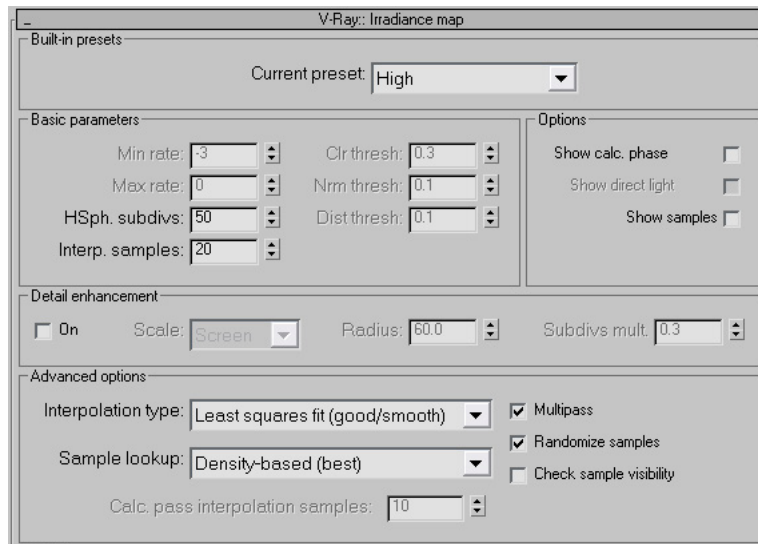
When light strikes an object, it illuminates the object with **Direct Light**. After direct light bounces off the object, the light illuminates other surfaces as **Indirect Light** (also known as indirect illumination, global illumination, or GI). GI in a scene is approximated through the use of complex algorithms that require an incredibly large number of calculations to complete. Like all advanced render engines, V-Ray has several options for making this approximation and each has their own distinct advantages and disadvantages. The Indirect Illumination rollout controls which methods are used.



Within V-Ray, indirect illumination is categorized into two areas. Indirect illumination that is directly visible by a camera's view or visible in the reflection and refractions from other objects is referred to as a **Primary Bounce**. Indirect illumination that simply used in the overall calculation of GI in a scene is referred to as a **Secondary Bounce**. V-Ray uses several different methods to approximate these bounces and each contains parameters that control the quality of the approximation and the time it takes to make the approximation. This discussion looks into the most widely used primary and secondary bounces types; the **Irradiance Map** and **Light Cache**, respectively. We'll return later to see which settings in this rollout warrant further discussion as a critical setting.

Irradiance Map

An irradiance map is a method for storing the calculation of radiant energy, or light, which strikes a set number of points in 3D space. With a 3D point map of these stored illumination values, V-Ray can approximate the amount of illumination that arrives at every point on every surface, and thus, better determine how to color the pixels in a rendering. What makes algorithms behind the irradiance map method so great is that they approximate GI more accurately in areas that require more detail, such as shadow transitions and places where different surfaces interact, and less accurately in areas where greater detail is not necessary, such as areas with unvarying illumination. To do this, V-Ray renders a camera's view in a number of different passes, starting with a lower resolution pass and progressively adding higher resolution passes. With each successive pass, V-Ray determines if the GI of each area of the rendered view is adequately approximated by samples taken in the previous pass. If so, no further samples are taken – if not, more samples are taken. With each pass, the resolution of the irradiance map pass is doubled, and therefore, V-Ray can potentially quadruple the number of samples taken. It would be impractical and far too time consuming to calculate a sample for every pixel in the rendered view, so instead, V-Ray takes what it believes is the required number of samples (to approximate the GI) and interpolates (or estimates) the areas in between. The results of these samples are stored in the irradiance map and the accuracy of the irradiance map is directly dependent on the number of samples and the quality of those samples. The settings in the top part of the Irradiance map rollout (shown below) control both of these factors, while the bottom half deals with animations.

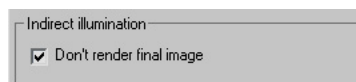


Min /Max Rates

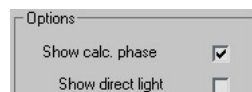
The **Current preset** drop-down list at the top of the rollout contains numerous configurations for the six grayed-out settings below it. These presets provide a quick and easy way to achieve good settings for various levels of accuracy needed in the irradiance map. There is also a Custom setting that gives you individual control over the six settings below it that are otherwise dictated by the presets. The first two settings that should be discussed are the **Min rate** and **Max rate**. These settings have the most direct impact on the number of samples taken during the calculation of an irradiance map. Before going any further, let's take a close look at the purpose of these two settings. The following exercise helps demonstrate what their role is.

Exercise – Specifying sample density with Min / Max rates

1. Open the file **irradiance.max**.
2. In the **Global switches** rollout, enable **Don't render final image**, as shown below. This prevents V-Ray from covering up the irradiance map with a rendering, therefore, allows us to see the results of the irradiance.



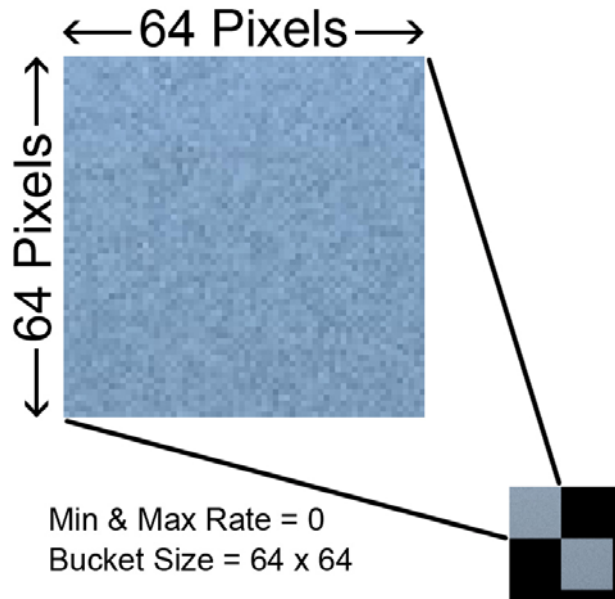
3. In the Irradiance map rollout, enable the **Show calc. phase option**, as shown below. This allows us to see the various resolutions of the irradiance map as they're being rendered.



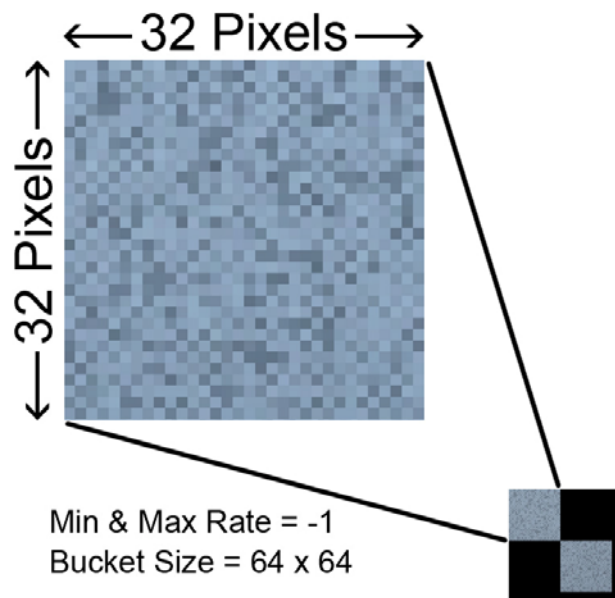
4. Render the scene and stop it after the first buckets are processed (this should only take approximately 5-10 sec).
5. Click on the rendered window and scroll your middle mouse button upwards to zoom into the rendered window. Continue scrolling into the window until you see an **8:1** zoom factor and use the scroll bars to get a closer view of the first bucket in the top left corner.

- Count the number of individual squares, from left to right, that make up the bucket.

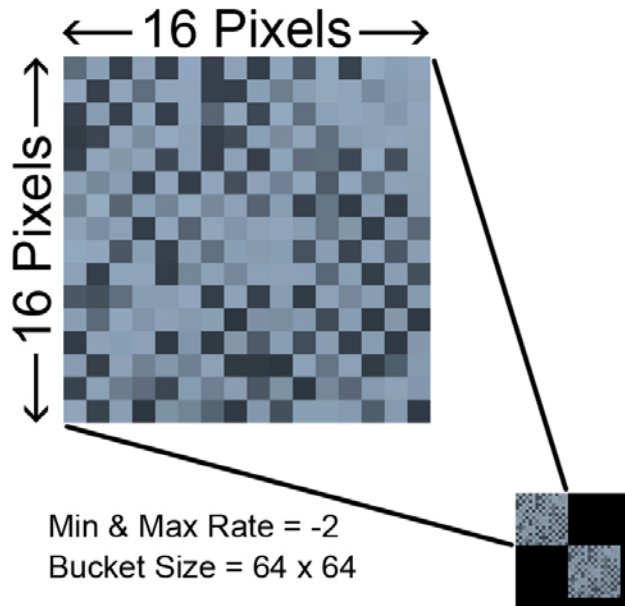
You should notice that there are 64 individual squares horizontally and vertically that make up each bucket. What does this mean? Well, if you look in the **System** rollout, you will see that each **Render region division**, i.e. bucket, is 64 pixels in both the X and Y directions. This means that there is exactly the same number of squares in the irradiance map as there are in the rendered image. Therefore, the irradiance map resolution is the same as the final rendered image when the Min rate and Max rates are set to 0. In this case, the rendered output size (as specified in the Common Tab of the Render Scene dialog box) is 640x480; therefore, the irradiance map resolution is 640x480 as well. This is illustrated below.



- Now change the Min rate and Max rate to -1 and render the image again. If you zoom into the render window again, you should now see that there are only 32 pixels horizontally and vertically that make up each 64x64 pixel bucket. This means that the irradiance map resolution is $\frac{1}{2}$ the final rendered image resolution.

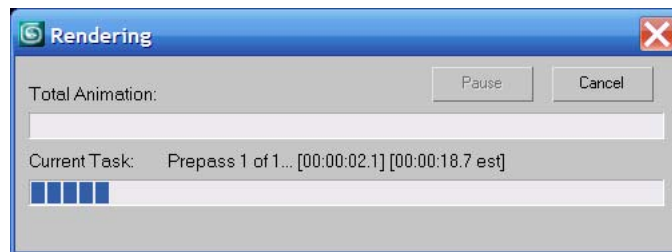


8. Reduce the Min and Max rate once more to -2 and render the image again. If you zoom into the render window again, you should now see that there are only 16 pixels horizontally and vertically that make up each 64x64 pixel bucket. This means that the irradiance map resolution is $\frac{1}{4}$ the final rendered image resolution.



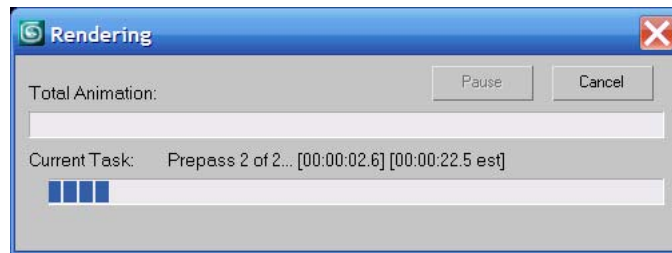
So far, we have kept the Min and Max rate values the same. This was done to simplify the discussion up to this point; however, by leaving these values the same, we are not allowing V-Ray to become adaptive to its environment, which is what makes irradiance so great in approximating GI. If we separate the Min and Max rate values, we allow V-Ray to apply a larger irradiance map resolution where greater sampling is needed, and a smaller irradiance map where less sampling is acceptable. To see how this works, let's continue with the exercise.

9. With the Min and Max rate still at -2, render the scene again. Notice that the render window displays the message **Prepass 1 of 1**, as shown below. This means that only one rendered pass is being made and this is because both the Min and Max values are the same. Click Cancel to stop the rendering, if necessary.

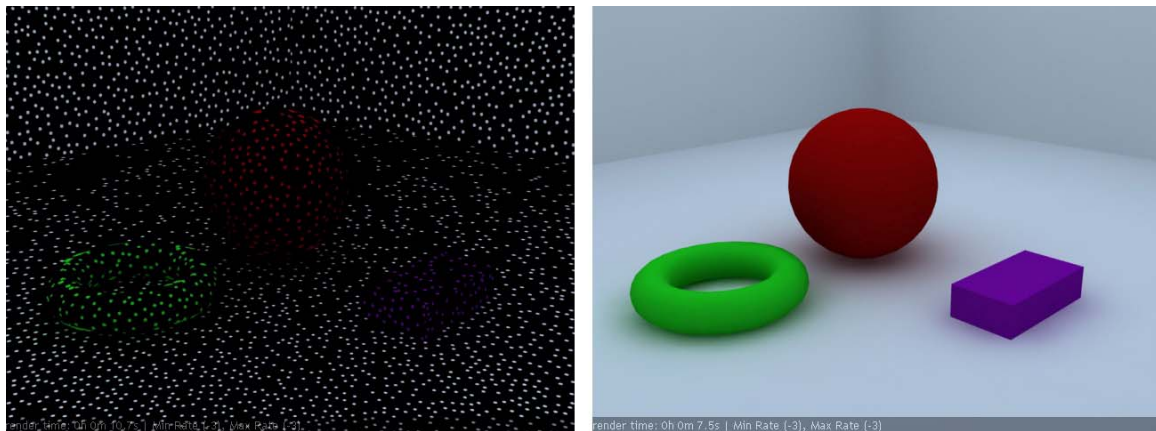


10. Change the Min rate to -3 and render the scene again. Notice now that the render window displays the message **Prepass 1 of 2** and then **Prepass 2 of 2**, as shown below. This means that two unique passes are rendered, the first one dictated by the Min rate value and the second one dictated by the Max rate value. For every pixel on the screen, V-Ray made a determination about whether the samples taken in the first pass were sufficient to approximate the GI. If not, more samples were taken in the second pass,

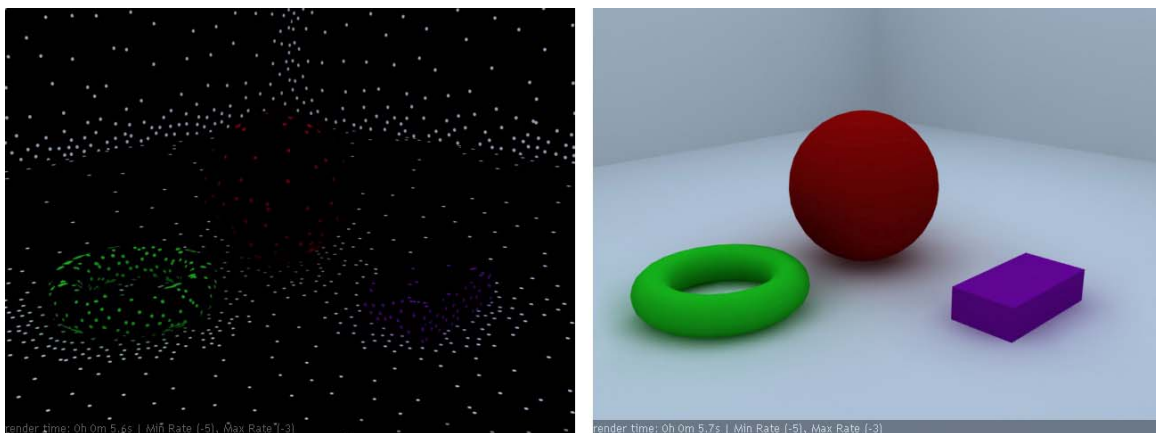
potentially four times as many (twice the resolution means four times the area). The more the Min and Max rate are separated, the more passes are taken and the more adaptive the irradiance map becomes.



These next 2 sets of images summarize the value of adaptive sampling with irradiance maps. In the left image below, you can see that a large number of samples (displayed as dots) are taken throughout the scene, with little care to areas that need more attention than others. This is because both the Min and Max rate were set to -3 and only one sampling pass was made. The result is a decent image but at the cost of greater rendering time (7.5 secs).



In these next two images, you can see that the same level of quality is achieved in the GI in less time (5.7 secs). Notice how samples were taken in greater density in certain areas.



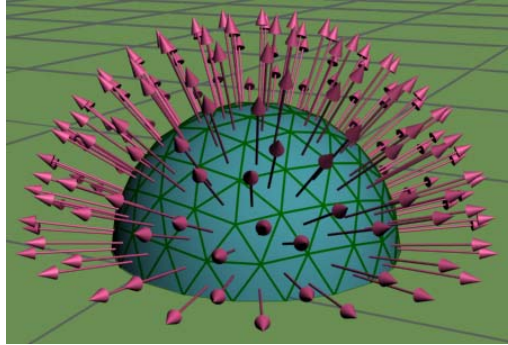
So what are good Min / Max rate settings for a typical visualization? Well, as was mentioned in the introduction, knowing V-Ray really means knowing how to troubleshoot. Ideally, you want to

use the lowest settings that give you the look you want without the undesirable effects that sometimes plague a scene where irradiance maps are used with low quality settings. Lower quality settings obviously mean faster render times. But what might be good settings for one scene may not work so for another. Nonetheless, the presets offer a great place to start and if you experiment a little, you'll probably find a preset that provides the best mixture of speed vs. quality.

At 3DAS, we generally use a Very Low for test renders of large scenes, a Low preset for test renders of small scenes, and a Medium preset for production renders. If we determine that a Medium preset allows for too much noise or too little detail, we often switch to a High preset. But as you will see in the next few sections, there are other settings within the Irradiance map that should be address besides just the Min / Max rates.

HSph. subdivs

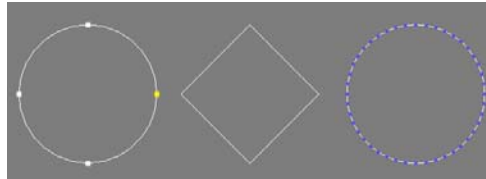
The next most critical setting in the rollout is the **Hsph. subdivs** value. At each point that a sample is taken, rays are traced out from the sample, mathematically speaking, to determine the amount of light arriving from each direction and improve the quality of the GI approximated at each point. The more rays that are traced, the more accurate the sample becomes. In simplistic terms, if you place a virtual hemisphere above each sample that is taken and trace rays from the sample through each subdivision of the hemisphere, the more subdivisions, the more rays that are traced. The image below gives a very rough graphical representation of this process. The actual number of rays that are traced are equal to the square of the Hsph. Subdivs value, although this can change depending of settings with the QMC Sampler rollout (discussed later).



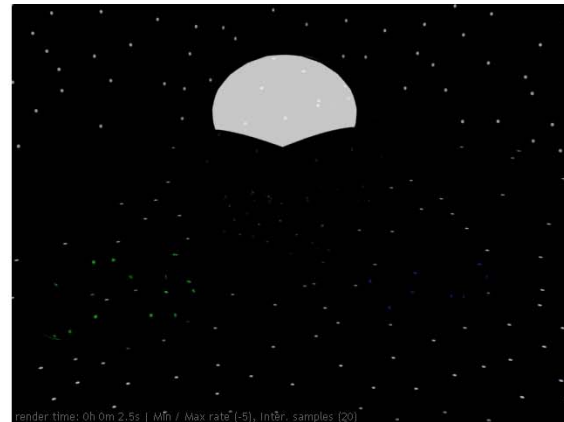
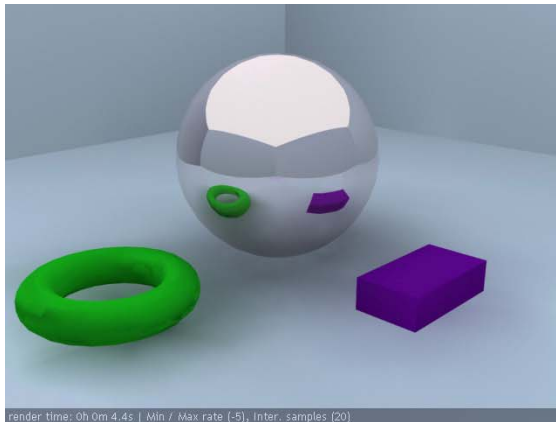
By default, HSph. subdivs are set to 50, however, this value is often much higher than necessary and can lead to excessive render times. Because of the impact this setting can have on both quality and render times, some experimentation may be in order. A good value to start with is 20 and if no problems exist in the GI of your scene, then you should not increase the value. If problems such as splotches arise, then experiment with small increases in this value. Values above 50 should be avoided because of the increase in render times that result (at the benefit of indiscernible gain or gain that can be achieved through other settings with smaller render times). Likewise, values below 20 should be avoided because the savings in render times are not worth the great loss of quality that can result.

Inter. samples

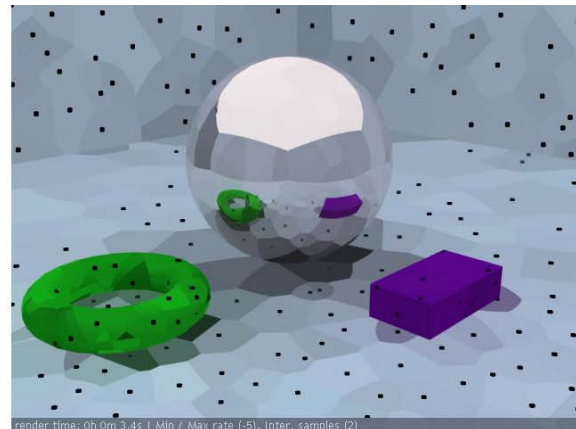
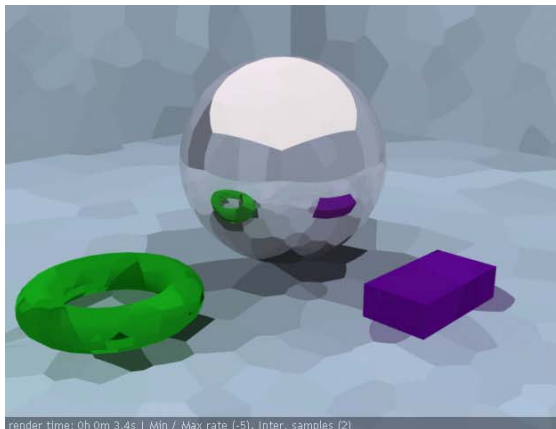
When you create a simple circle, like the one shown in the far left image below, four vertices are created automatically to help guide its shape. By default, 6 interpolation steps are used automatically to give the circle its curved appearance. If you remove all interpolation, all curvature is lost and you are left with a diamond shaped object. If you increase the interpolation steps to 10, as shown in the far right image, you allow the circle's curvature to be better defined. Interpolation in the irradiance map works in much the same way.



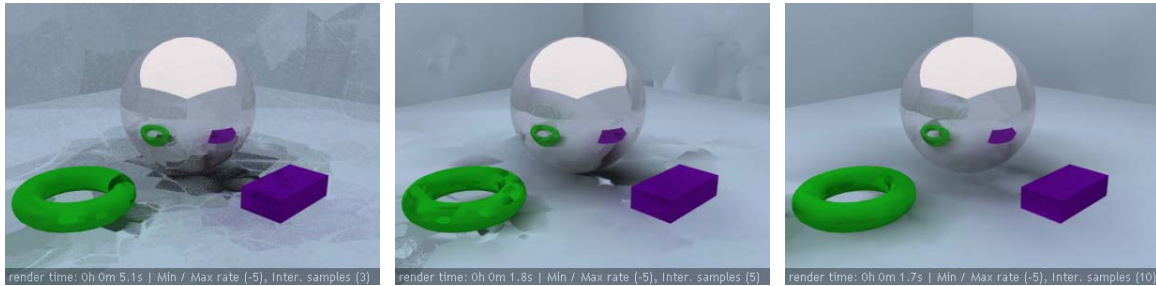
The Inter. Samples value determines how many of the irradiance map samples that were taken will be used to estimate the GI between those samples. In the far left image below, the Min rate and Max rate was set to -5, which results in a small number of samples, as shown on the right. This is obviously not a good setting but it helps illustrate the effect of interpolation.



In the left image below, the only thing that was changed was Inter. Samples. At a value of 2, V-Ray only takes into consideration 2 samples to determine the GI at any point. The result is a cell-like appearance to the GI, and as you can see in the right image, each sample is centered on a uniformly shaded cell. The sides of each cell lie directly in between the 2 samples that were used to determine the shading on each side of the cells' border.



In the images below, the Inter. samples value used was (from left to right) 3, 5, and 10. As you can see, increasing the interpolation results in greater smoothing (or blurring) of the GI.



The default value of 20 works great in most situations. You can try increasing this value to 30 to remove any blotchiness that may persist in your GI or you could try reducing to 10 to help bring out more detail in your GI.

Detail enhancement

As its name implies, this is an option that can be used to prevent the GI from being blurred so much, thus bringing out more detail in the rendered output. Using this option can have a dramatic effect on the rendering times because of the higher precision sampling that it generates, and therefore, it should only be used after less processor hungry methods have been used to bring out detail. To see the effects of this option versus other methods of detail enhancement, let's look at the following exercise. We will return to this same exercise later for part II, where other methods will be used.

Exercise – Improving irradiance map details (Part I)

1. Open the file **building01.max**.
2. Render the scene. Notice that with an irradiance map preset of Low and detail enhancement off, there is a significant amount of detail missing in the reveals of the building. In some areas, the lines are completely removed. These are structural elements, not mapped elements; however the same concepts apply to mapped elements.



3. In the Irradiance map rollout, enable detail enhancement and render the scene again. Notice how there is almost no noticeable change to the details of the reveals even though the render time increased almost 6-fold. This would be a poor use of detail enhancement. There is something else wrong with the settings in this scene that should be fixed before the Detail enhancement option should be utilized.



4. Disable detail enhancement and change the irradiance map preset to High.
5. Render the scene again. Notice that even with a High preset, there is still no noticeable improvement in the detail of the reveal lines. You could use some of the settings within the detail enhancement section to improve the effects, but the default settings are sufficient for most situations and improving them here just won't be worth the cost of increased render times.

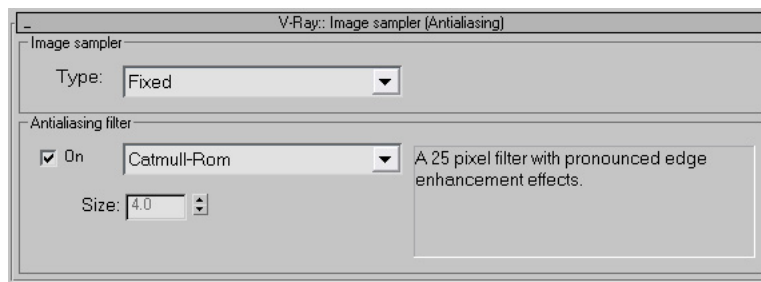


This demonstration does not mean that detail enhancement is not worthwhile, but its effects are negated by settings in another area of V-Ray. Can you guess which? We'll return to this exercise shortly, but for now let's depart a discussion on irradiance maps to look at the area that is causing problems with the irradiance maps in this scene. The Image Sampler!

Image Sampler

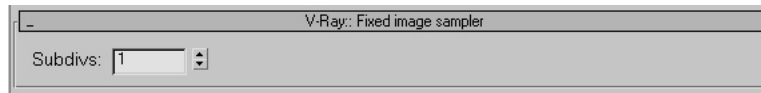
With the exception of post-rendering effects, the final step in any rendering process is antialiasing. Aliasing is a term to describe imperfections in the rendering process caused by color changes that are too drastic, and that occur over too small an area of screen space to be adequately depicted by the pixels that define that space. Anti-aliasing techniques attempt to smooth jagged edges by properly handling fragment pixels (i.e., adjusting the pixel color according to the amount of pixel coverage). V-Ray supports all standard 3ds Max filters and for more information on which ones should be used for any given scene, refer to the [Insider](#) article entitled [Antialiasing and its Side Effects](#). But before antialiasing can be performed, V-Ray has to sample each pixel to determine what objects in a scene lie within the boundary of each pixel and what color the objects are in the area of each pixel. The image sampler is the tool used to do this, and therefore, it plays a critical role in the final rendered output of a scene. Without the necessary imager sampler configuration, high quality settings in so many other areas of V-Ray and 3ds Max are negatively affected. Such was the case in the last exercise.

The **Image sampler** rollout, shown below, controls which of the 3 available image samplers are used, but the actual settings for each individual sampler are found within the unique rollout that appears for the chosen sampler. The next section discusses how each of samplers works and the advantages and disadvantages of each.

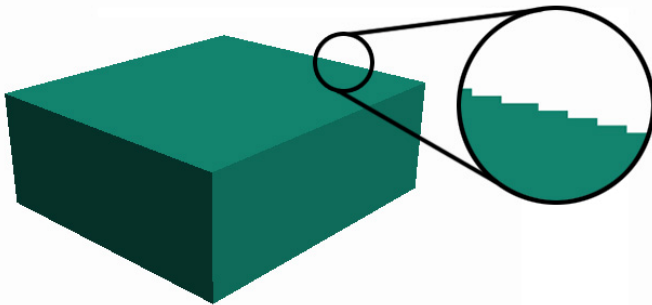


Fixed image sampler

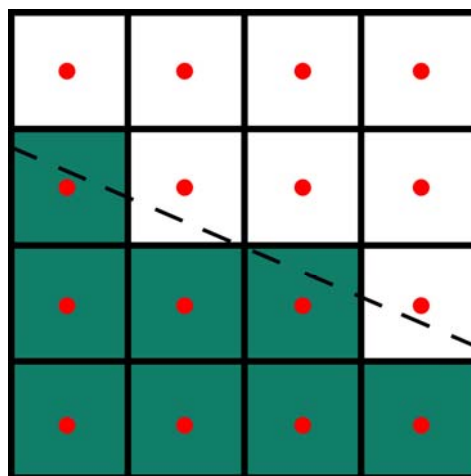
The Fixed image sampler is clearly the most simple of the 3 with only one parameter available, as shown below.



In the image below, a box is rendered with a very noticeable jagged appearance to the borders. This is because the edges of the box do not lie perfectly on the edges of the pixels that are used to show the transition from one color to the other. The result is that some pixels receive the white background color and some receive the blue-green object color. The box was rendered with the Fixed image sampler using the default Subdivs value of 1 and no antialiasing. Antialiasing could have been used here (as it always should be) to blur the perimeter into a more natural appearance, but it was disabled to show the effect of the image sampler. But how does V-Ray perform the sampling process and how can you control and improve it.

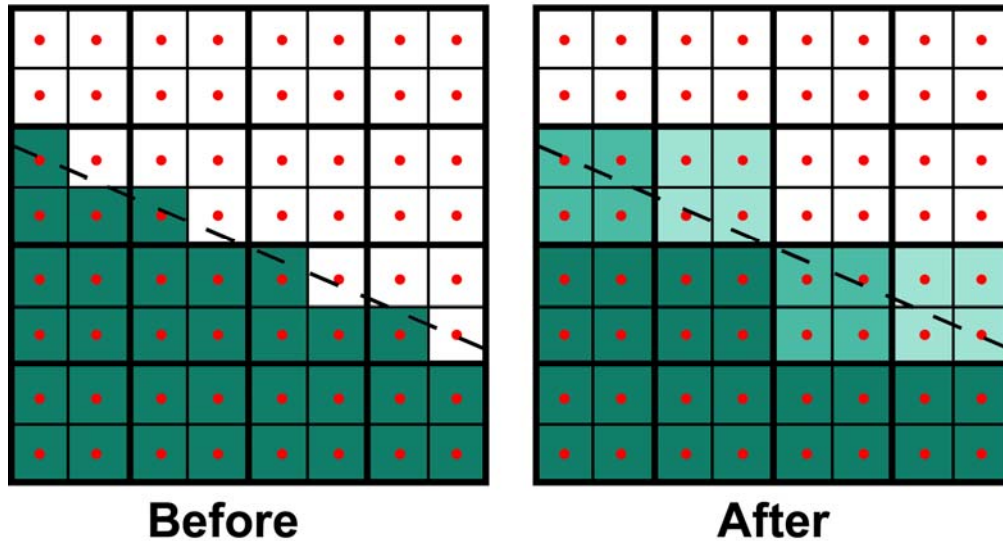


As it was previously mentioned, the edge of an object or the details of a map will rarely align perfectly to the edge of a pixel. Such is the case in the box in the image above. The image below shows a simplified and blown up version of the box's edge using a 4x4 pixel grid (16 pixels total). The dashed line represents where the edge of the box wants to be displayed, but because of the limited resolution of a computer screen, this edge can never be perfectly represented. With an image sampling using 1 subdivision, 1 sample is taken in the center of each pixel. The samples are shown with red dots, and the color that a sample detects is the color that is assigned to the entire pixel. As you will see, 1 subdivision is usually far from sufficient for a decent transition.



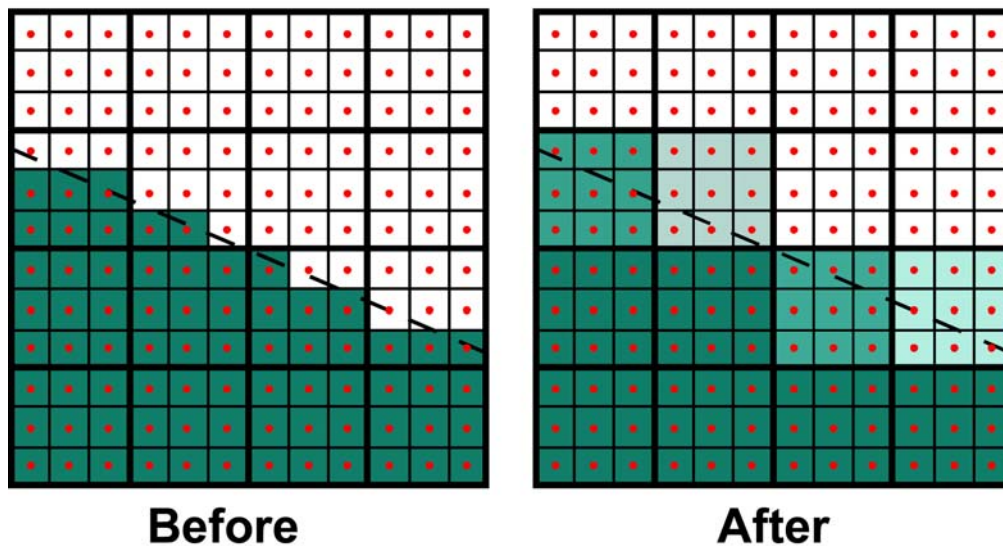
1 Subdivision

With a subdivs value of 2, each pixel is subdivided into 2 subpixels horizontally and vertically and a sample is taken in the center of each subpixel, as shown in the left image below. Because of the greater subdivisions, the true location of the box's edge can be better determined. After determining what percentage of subpixels receive the background color and what percentage receives the box's color, the image sampler adjusts the color of the entire pixel based on this percentage. In the image on the right, you can see that the 2 pixels that have 3 subpixels covered by the box receive a darker color than the 2 pixels that have only 1 subpixel covered.



2 Subdivisions

In the next set of images below, 3 subdivisions were used and each pixel is subdivided into 3 segments horizontally and vertically (for a total subpixel count of 9). The result is greater accuracy in determining the location of the box's edge, and therefore, better adjustments in the coloration of each pixel.



3 Subdivisions

As the name implies, the Fixed image sampler takes the same number of samples everywhere in an image. The primary advantage of this sampler type is that an image's quality and render time is more predictable than the other 2 sampler types. The primary disadvantage is that because it's not adaptive like the other 2 samplers, it can lead to greater render times. This sampler is best utilized in scenes with a great amount of detail (either structurally or in maps), and/or blurry effects. As a general rule, if 75% or more of a rendered image contains areas where a high level of detail is needed, we recommend using the Fixed image sampler. By not using this sampler on scenes with a large amount, you run the risk of having V-Ray guess wrong as to which areas should receive greater sampling.

To see the effects of this sampler, let's return to the last exercise which discussed irradiance maps. We will be able to see how changing the image sampler effects the details that can be achieved in the irradiance map.

Exercise – Improving irradiance map details (Part II)

1. Open the file **building02.max**. The scene is using the adaptive subdivision sampler, the Low irradiance map preset, and no detail enhancement.
2. Render the scene. The reveal lines still lack detail in many areas, as shown below.



3. Change the image sampler type to Fixed and change the Subdivs value to 4. This means that each pixel will contain 16 subpixel samples. The improved results are visible in the image below. Notice that the rendering time increased approximately 2.5 times.

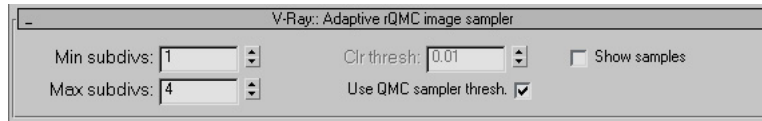


4. In the Irradiance map rollout, enable the detail enhancement option and render again. The render time increases, but you can see an improvement in details that weren't previously brought out using the detail enhancement. In particular, notice the 1st reveal to the right of the wall corner, which is far sharper than in the previous image.



Adaptive QMC sampler

The Adaptive QMC sampler works similar to the Fixed sampler except that, like any adaptive setting, it adapts to its environment and applies a range of possible subdivisions based on the Min and Max values used. In areas of less detail, the Min value may be used and in areas of greater detail, the sampler is more likely to use the Max value.



By default, the Min and Max values are 1 and 4 respectively, which provides decent quality for test purposes. However, these values will often need to be raised for production renders. Values such as 3 and 6 are typically good settings for most production purposes. Caution should be taken in using higher values than 6 because of the possibility of excessive render times. Additionally, it's a good idea not to separate the Min and Max values by more than 2 or 3. Doing so can result in a wide range in quality of detail because it leaves V-Ray more room to guess incorrectly.

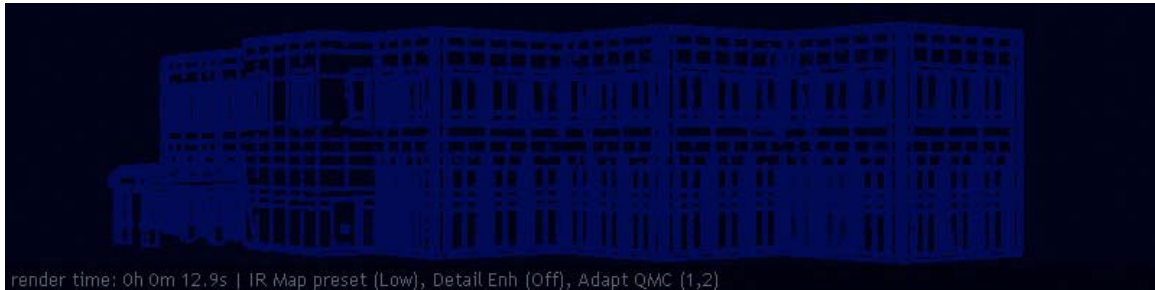
To see how similar the Fixed and Adaptive QMC samplers really are, try rendering the same scene with both sampler types and make the Min and Max subdivs values of the Adaptive QMC sampler be the same value as the Fixed subdivs. The image below shows that the same level of detail is achieved with nearly identical render times, compared to the middle image on the previous page.



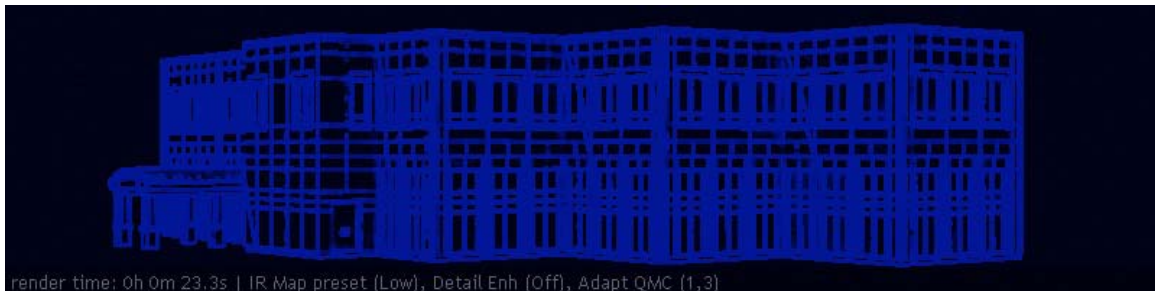
The Adaptive QMC sampler works best in scenes with a large amount of detail or blurry effects. As a general rule, we suggest using this sampler in scenes where a high level of detail is needed anywhere, but not more than 75% of the rendered image. As mentioned before, when the combined area of an image needing a high level of detail is more than 75%, the Fixed sampler is probably the best choice simply because the small amount of time saved with the Adaptive QMC sampler is not worth the risk of having some areas under sampled. In either case, you should definitely experiment with several configurations to determine the optimal mix of quality versus render times.

Displaying Adaptive QMC Samples

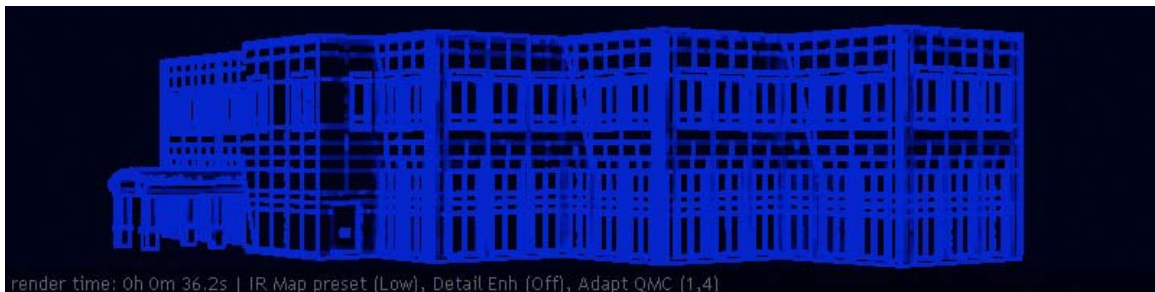
An additional benefit of using the Adaptive QMC sampler is that you can actually see where V-Ray applies the different subdivs. value in the scene. In the below, Adaptive QMC was used with a Min and Max subdivs. value of 1 and 2, respectively. The result is an image displaying 2 different colors. The lighter color represents where V-Ray applied the 2 subdivisions and the darker color represents where only 1 subdivision was applied.



In this next image, the Min and Max values were set to 1 and 3, thereby, bringing about 3 different colors.



And with a Min and Max value of 1 and 4, 4 colors were used. The brightest color indicates the area of greatest subdivisions.



Adaptive subdivision sampler

The Adaptive subdivision sampler is unique in that it is the only sampler that allows you to take fewer than one sample per pixel. Notice that instead of using a Min / Max **subdivs.** setting, it instead uses a Min/Max **rate**. Instead of describing the sampling as subdivisions of a pixel, it is saying that pixels are sampled at the rate of ____ pixels for every 1 sample. A rate of 0 means one sample per pixel. A rate of 1 means 4 samples per pixel. Conversely, a rate of -1 means that pixels are sampled at the rate of 4 pixels for every 1 sample. Because of this capability, you have the capability of greatly reducing rendering times in a scene. However, the adaptive subdivision sampler does not work great for all types of scenes. This sampler should be restricted to use in scenes that contain few blurry effects and a minimal amount of highly detailed areas. In scenes with large areas of smooth surfaces, this sampler works great.

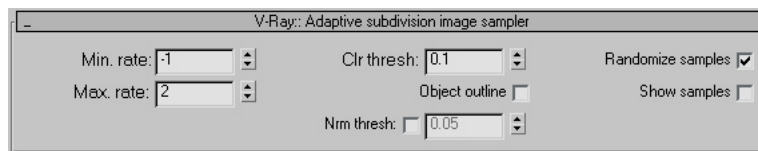


Image sampler examples

Before you even start to test, you should have a pretty good idea of which sampler would be best to use for a particular scene. Here are 3 examples images where we would use the 3 different samplers.

In the image below, we would recommend the Fixed image sampler because there are so many areas where a high level of detail is needed, such as the crown molding, and it is found throughout the entire image.



In this next image, the Adaptive QMC would be a good choice to try because there are definitely areas where a high-level of detail is needed, such as the palm fronds and building details, but there are also areas where the detail is not needed, such as the sky and water. If you try this sampler and are not pleased with the results, simply try higher subdivisions or try the Fixed sampler.



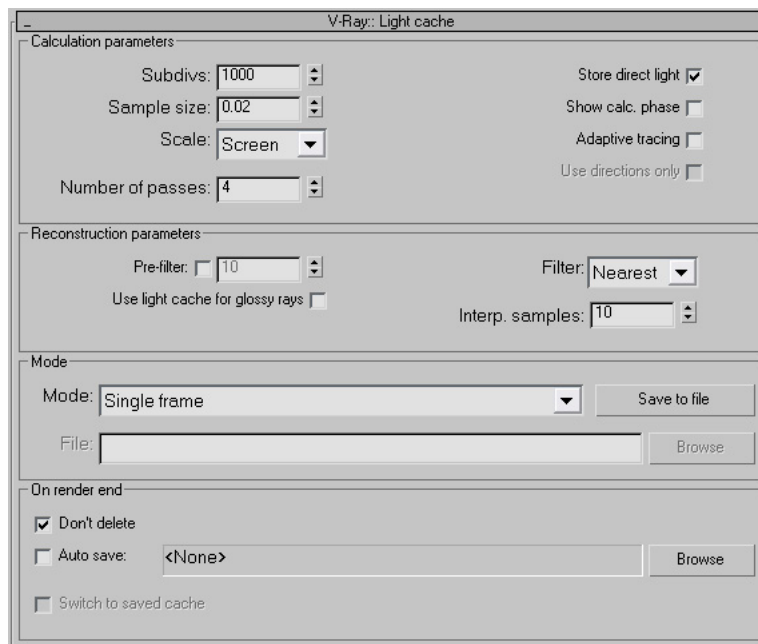
This last image is an example where the Adaptive subdivision method would probably shine. It would be effective here because of the large amount of smooth surfaces, such as the ceiling and furniture. Even in areas where there is a lot of detail, such as the paintings and rug, a high level of detail is not critical.



Light Cache

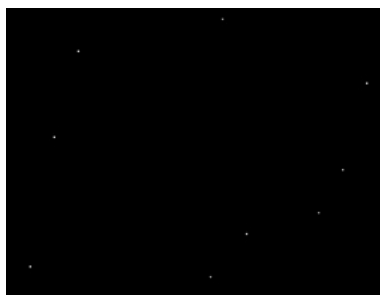
The light cache is arguably the most widely preferred secondary bounce engine among V-Ray users, not only because of its great mixture of speed and quality, but also because it's very simple to use. The light cache does not have variables that control adaptive behavior like the irradiance map and because of this alone it is a much simpler solution with fewer variables having a large impact on its use.

The light cache has numerous advantages over the other options for secondary bounce engine types, but it primarily boils down to quality versus speed and to achieve the same level of quality with the other types, the settings have to be adjusted so high that their rendering times will almost always exceed that of light cache.



Subdivs

The light cache works by shooting virtual rays from the camera and the **Subdivs** value directly controls the number of rays that are used. The actual number of rays that are propagated is equal to the square value of this setting. A good way to see this is to set both the primary and secondary bounces engines to light cache, enable the **Don't render final image** option, and render a scene with a small number of subdivs. In the image below, for example, the Subdivs were set to 3, and therefore, 9 samples were used.



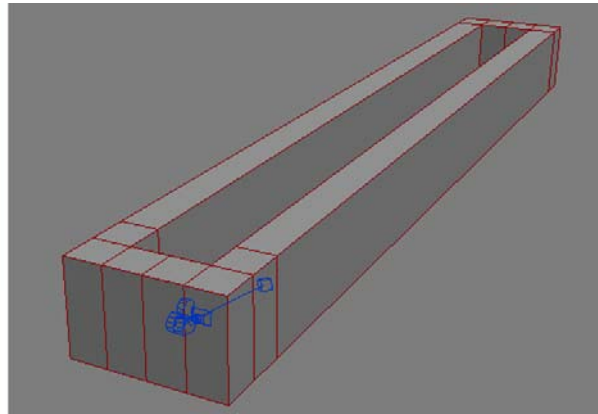
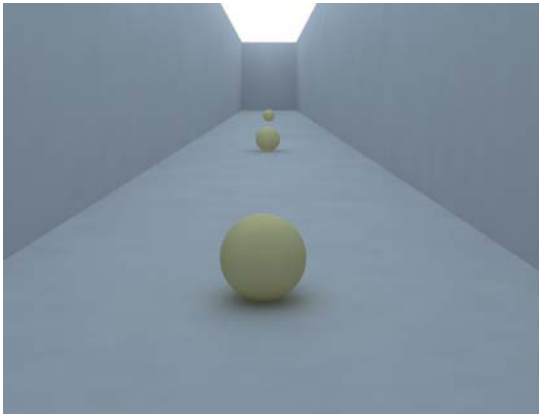
As with most settings, we recommend two different ranges for the Subdivs value; one for test renders and one for production. For test renders, try values around 100 for large scenes and 250 for small scenes. For production renders, simply multiply these values 10 times. For production renders, try values around 1000 for large scenes and 2500 for small scenes. Values above 2500 should be avoided because of the negligible benefit that comes at a greater cost of rendering time. Likewise, values below 100 can provide really poor results and not save a significant amount of time.

Sample size

The **Sample size** setting is interconnected with the Subdivs setting and specifies the maximum allowable size of a sample if, and only if, there are enough samples to allow for the size specified with this setting. In other words, if the Subdivs value is set too low and there just aren't enough samples being taken, it doesn't matter how small you make the sample size because all the samples have to be spread out to cover the screen. The following exercise demonstrates the relationship between the number of samples and the sample size.

Exercise – Light cache sampling

1. Open the file **corridor.max**. As shown in the images below, this is a simple scene comprised of a single object representing the walls and floor of a corridor and 3 spheres spaced throughout the length of the corridor.
2. Render the Camera01 view. It should look similar to the left image below. This image was rendered with an irradiance map, which provides a nice look, but for purposes of this demonstration, we will need to make both the primary and secondary bounces light cache.

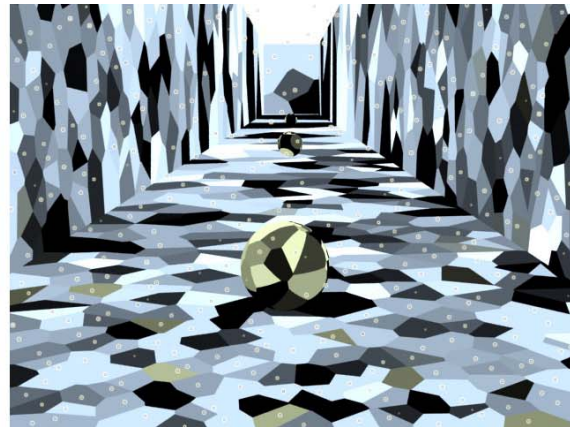


3. Change the primary bounce engine to light cache.
4. In the Light cache rollout, reduce the number of passes from 4 to 1. This option allows you to take advantage of calculating a light cache with multiple threads and/or other computers by letting each thread/computer calculate part of the total solution. This is great for network rendering, especially when animations are being rendered. When the final light cache is shared among those threads/computers, it will lead to less variance in the GI from one frame to another, such as flickering. For the purpose of this demonstration, we need to change this value to 1, simply to prevent a blending of samples from the multiple threads used.
5. Reduce the Subdivs value from 1000 to 20. This reduces the number of samples from 1,000,000 to 400.
6. Reduce the Interp. samples 10 to 1. We'll discuss the relevance of this setting in a moment.

7. Render Camera01 again. Notice how cellular-like the final rendering is, as shown in the image below. You could actually count the individual 400 samples.



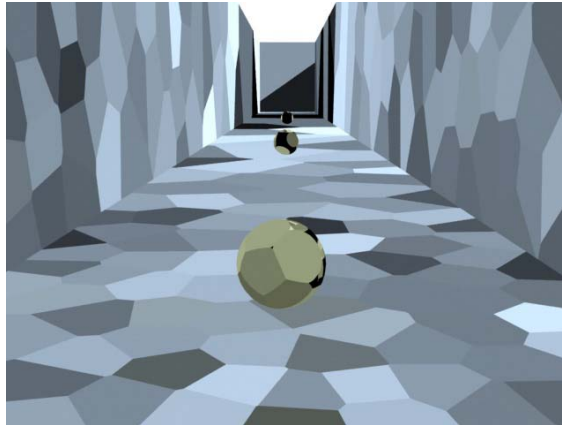
8. In the Global switches rollout, enable the **Don't render final image** option.
9. Render Camera 01. You should now see the individual sample spots to which the sample rays were cast, as shown in the left image below. If you superimpose these dots on the final rendered image, you can see that they lie in the center of the cell-like sample areas.



10. Disable the **Don't render final image** option.

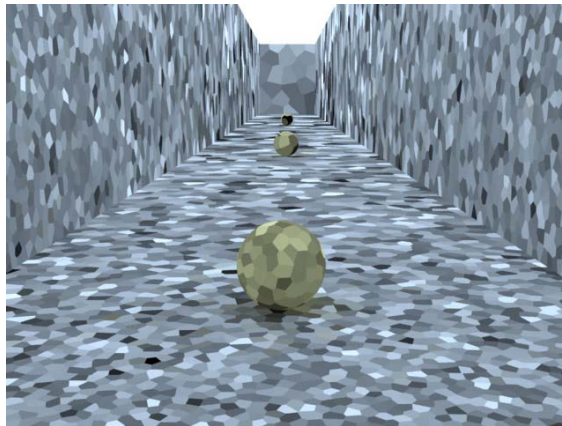
In the Light cache rollout is another setting that determines final sample size throughout a scene. When the Scale setting is to Screen, the Sample size value is a fraction of the screen size. When set to the default value of 0.02, the Sample size value wants to limit each sample to be no larger than 2% of the screen's size. This does not mean that the sample size will necessarily be 2% or smaller because if there aren't enough samples to divide up the image into, then the actual sample size could be much larger. Let's see this.

11. Decrease the Sample size from 0.02 to 0.01.
12. Render Camera01. Notice that very few samples were actually affected because there were only a few samples larger than 2% of the screen size.
13. Change the Sample size from 0.01 to 0.1. Now you are telling V-Ray to make the samples no larger than 10% of the screen size.
14. Render Camera01. Now the Subdivs value is no longer the final determining factor of the sample size. Since there are enough samples, the sample sizes can now be as large as dictated by the Sample size value, as shown below.



Now let's look closer at what the Scale setting means. If you look at the last rendered image (top of previous page), you'll see that all the samples are the same size relative to the camera's view. But look closely and you'll realize that a sample in the foreground represents a larger area than the same sample size on the back wall of the corridor. This means that the samples are less accurate when they are taken farther away from the camera. Although this is not a problem from the current perspective, if you move down the hall and render the image again with this saved light cache, your image will likely not have as much quality. Let's see this.

15. Change the Subdivs value to 100.
16. Change the Sample size to 0.02.
17. Render Camera01. The image should look similar to the one below. Notice how large the samples are against the back wall. They are the same size on the screen but in world-space coordinates they represent a much larger area than those samples in the foreground.



18. Change the Scale to World.
19. Change the Sample size to 1'0".
20. Render Camera01. As you can see, all of the samples are the same size throughout the scene – 1'0", as shown below.

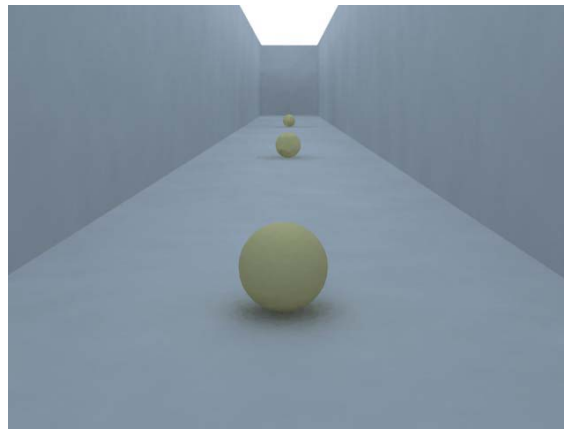


Now let's restore the image to a higher quality with some of the remaining light cache settings.

21. Increase the Subdivs value to 1000. You might think that this would return the image to a high level of quality because of the 1 million samples being taken, but it will still be far from desirable.
22. Render Camera01. As you can see in the image below, the rendering still has a noticeable cellular appearance. This is because of the sample size constraint still in effect – 0.02.



23. Increase the Interp. samples from 1 to 10. Just like other areas in V-Ray where interpolation is used, this tells V-Ray to spend a little time after the samples are taken to estimate the light cache in the areas between samples. Notice that this setting is in the Reconstruction section of the rollout, indicating that it is in fact performed after the light cache samples are taken.
24. Render Camera01. Significant improvement in the light cache can be seen, as shown below. The Interp. samples value can be left at the default value of 10 for most situations. Increasing the value will blur the light cache which will help smooth the appearance but will also cause some details to be lost. Further blurring can be accomplished with the Pre-filter option but this is not recommended for most situations. Using this feature, just like adjusting the Interp. samples, will only hide a problem most likely caused by other poor settings. The end result of too much blurring is not only a loss in detail but a lack of a GI quality.



Summary

This article focused on what are arguably the 3 most critical and fundamental features in V-Ray – irradiance maps, light cache, and image samplers. High quality images simply cannot be achieved without a good GI and image sampling configuration. The irradiance map and light cache have become the most widely used GI solutions and without knowing the fundamentals behind them, users usually spend countless hours struggling with their settings. After a proper GI solution, there is no other more important and complex component to a good rendering than image sampling and antialiasing. Nearly every 3D scene is victim to the limitations and side effects of these two features but by knowing how they work, you can quite easily control their influence.

If you want to read more on the subject of sampling with irradiance maps, a great reference is a paper entitled Irradiance Gradients, by Greg Ward.

<http://radsite.lbl.gov/radiance/papers/erw92/paper.pdf>