

Mental Ray - Tutorial Comparativa luz natural

Hola a todos, como lo prometido es deuda os propongo un taller parecido al anterior que publiqué, interior mental ray luz artificial, donde en este caso voy a intentar profundizar en los parámetros de Final Gather, GI y Ambient Occlusion para que podáis entenderlos mejor y sepáis ajustarlos en función de la escena con la que estéis trabajando dependiendo de las condiciones de luz.

Se trata de ver como influye la cantidad de luz directa en una escena a la hora de configurar correctamente FG y GI. En este caso, intentaré llevar la explicación de forma paralela en dos escenas IDÉNTICAS, salvo que una tiene muy poca cantidad de luz directa (ventanas pequeñas) y la otra gran cantidad de luz directa entrando por grandes ventanales.

Se trata de dos escenas modeladas igual, con los mismos objetos y en la misma posición, solo las diferencia el tamaño de las ventanas. Por lo demás:

- Motor: Mental ray.
- Unidades Escala: cm.
- Luz: Sistema Daylight con MR Sun y MR Sky (misma hora en ambas escenas)
- Environment: MR Physical Sky.
- Material Arch & Design Mate blanco para empezar (sin AO).
- FG y GI desactivado.
- Control de exposición: MR photographic exposure ctrl.

De momento os cuelgo las dos escenas en un rar y os pongo un par de renders SOLO CON LUZ DIRECTA para establecer el punto de partida. Todos los aportes, dudas, comentarios, lógicamente, serán bienvenidos por todos.

Un saludo chicos.





Bueno, pues vamos al lío. Primero un poco de resumen y recordatorios.

Ya sabéis que los mejores resultado en Mental Ray para trabajar con la luz indirecta son (generalmente) combinar Final Gather y Global Illumination. En este ejercicio/tutorial/taller o como queráis llamarlo vamos a hacerlo así fijándonos en cómo la cantidad de luz directa que ya tenemos en la escena afecta a los parámetros de calidad de FG y GI y por lo tanto a los tiempos de render. Veremos que soluciones y trucos podemos usar cuando las cosas empiecen a complicarse.

El proceso normal en el workflow de los cálculos de luz indirecta suele ser:

1.- calculamos primero un mapa de fotones con GI

En este punto es importante recordar que cuando vamos a usar juntos GI y FG el mapa de fotones no es necesario que sea detallado y completamente suave.

2.- calculamos FG y guardamos el mapa.

Final Gather utiliza el mapa de fotones previamente guardado para sus cálculos de luz indirecta.

3.- Con los mapas guardados, aplicamos materiales y ajustamos los parámetros de calidad de render para la imagen final.

Al usar los dos métodos juntos aprovechamos las ventajas de ambos:

- FG ofrece resultados suaves en el sombreado de la iluminación indirecta con poco esfuerzo, pero le cuesta mucho trabajo "rebotar", llevar la luz indirecta a zonas alejadas de las entradas o fuentes de luz directa.

- GI transporta rápidamente la luz indirecta iluminando sin mucho esfuerzo la escena. Por contra, intentar conseguir un render final suavizado y con mucho detalle en la iluminación indirecta es bastante costoso.

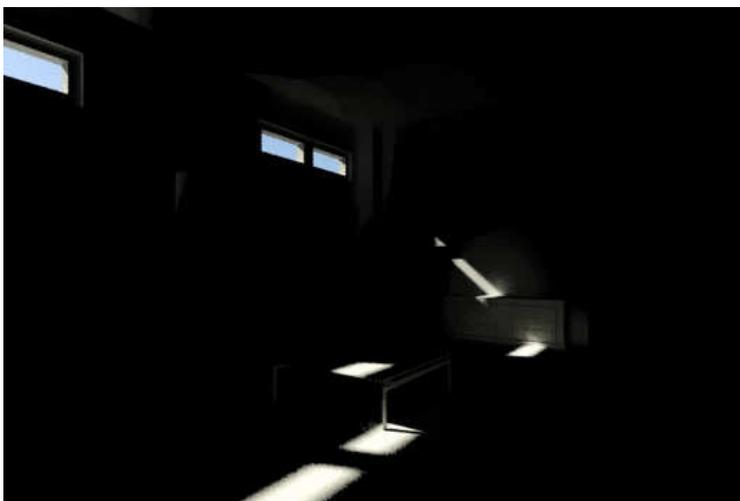
Por tanto, combinando la rapidez de GI con la mejor apariencia de FG se consiguen los resultados con mejor relación tiempo/calidad.

En este caso y para que podáis entender perfectamente todos los parámetros de FG (sobre todo) y GI dependiendo de la cantidad de luz, vamos primero a trabajar solo con FG y ver cuáles son los problemas que tenemos, luego solo con GI y problemas, y por último combinaremos ambos.

Así que vamos a empezar con las dos escenas a la vez, primero SOLO TRABAJANDO CON FINAL GATHER.

Para ambas:

- Panel Render Setup (F10) -> pestaña Indirect Illumination.
- Activamos la casilla "Enable Final Gather".
- Movemos la pestañita de "Preset" a la izquierda hasta ponerla en Draft.
- A continuación, nos vamos al menu Rendering->Exposure control.
- En MR photographic exposure control, ponemos el valor de exposición (EV) en 12.
- Renderizamos.



Bueno, pues aquí tenemos nuestro punto de partida con iluminación indirecta con FG. Vemos en la escena con mucha luz y los mismos valores de FG empiezan a verse el resto de objetos de la escena que antes no aparecían cuando solo teníamos luz directa.

En la escena con poca luz se atisban solo el aparador y la pared del fondo y se adivinan muy tenuemente algún otro objeto.

Recordad que, por defecto, Final Gather calcula solo la incidencia del primer rayo de luz indirecta. Si queremos más luz indirecta en la habitación debemos usar el primer parámetro importante, "Diffuse Bounces". Con este valor, le decimos a Final Gather que recopile (Final Gather significa "recolección Final")(y no es un golpe especial de un personaje de videojuego...) la cantidad de luz indirecta del primer, segundo, tercer... etc, rebote de la luz en la habitación.

Parece claro apriori que la escena con mucha luz va a necesitar pocos rebotes para conseguir una luz adecuada. Seguro que no es el caso de la escena con poca luz...

- Para mucha luz vamos a tirar dos renders con 1 y 2 en diffuse bounces.
- Para poca luz vamos a tirar tres renders con 1, 5 y 12 en diffuse bounces



Los renders salen muy muy feos pero para ver la cantidad de luz que recolecta FG es más que suficiente. Y ya tenemos la primera consecuencia.

En la escena muchaluz (vamos a llamarle así) con 1 rebote ya tenemos más luz que en pocaluz con 12. Es lógico, pues de partida había más luz directa, luego había más luz para repartir y tiene que rebotarla menos veces. Luego para obtener un resultado similar debemos rebotar muchas más veces la luz en una escena que en otra. Como el número

de rebotes influye directamente en el tiempo de cálculo, ya tenemos que el render de pocaluz 12 rebotes tarda ¡4 veces más! que el de muchaluz con 2 rebotes.

Ahora los tiempos son muy bajos por los valores de calidad que tenemos (4 sg una escena y 16 la otra) pero si seguimos la progresión cuando el render de una tarde 1min el de la otra serán 4 y si uno tarda 10min el otro... pues 40. Recuerdo que esto sería SOLO SI USARAMOS FG.

Para que tengáis una referencia de los tiempos os pongo el equipo que estoy usando:

- AMD Phenom Quad Core 9650
- 2 Gb Ram
- Gráfica Asus 9600GT

Dando por sentado que la cantidad de luz generada por los rebotes para ambas escenas es adecuada, es decir, me quedo con 12 rebotes para poca luz y 2 para mucha luz (parece lógico pensar que pocaluz se verá más oscura y tenue que mucha luz), el siguiente paso es empezar a ajustar los valores de calidad de FG para obtener un mejor render.

Continúo donde lo dejamos ayer. De momento ajustamos los rebotes (diffuse bounces) de FG para repartir la luz. Una vez fijados vamos con los parámetros principales de Final Gather para darle mayor calidad a la luz indirecta: Initial FG point density, Rays per FG point e Interpolation Over Num FG points.

El funcionamiento de Final Gather es el siguiente:

Los cálculos de luz indirecta responden a métodos estadísticos que "interpolan" la cantidad de luz de una serie de muestras a toda la escena. Me explico. Si realmente tuvieramos en cuenta cada rayo de luz o cada fotón para calcular la cantidad real de luz rebotada desde luego el resultado sería hiperrealista, pero los tiempos de cálculo serían imposibles para ordenadores normales.

Algo similar a esto último lo ofrecen algunos métodos de cálculo de distribución de energía (utilizados por ejemplo para la fabricación de altos hornos y aparatos similares) que intentan calcularlo "a lo bestia", como el Montecarlo Bruteforce. Si habéis trabajado alguna vez con Vray, este motor tiene (o tenía) un método de cálculo llamado Quasi Montecarlo, basado en el Bruteforce pero con interpolación. Todo esto os lo cuento de memoria porque Vray no es lo mio...

Por lo tanto y para ahorrar tiempo de cálculo se usan métodos estadísticos. Es como hacer una encuesta. Si yo pregunto a 20 vecinos si les parece justo que echaran a Karnele de Eurovisión, el resultado de la encuesta lo podría extrapolar a toda mi comunidad de 50 vecinos con resultado muy próximo a la realidad. Pero si, por ejemplo, al 50% les parece mal y a la otra mitad les parece bien, sería muy arriesgado decir que a la mitad de los españoles les molaría ver a Karnele en Eurovisión y a la otra mitad no. No puedo extrapolar el resultado de una muestra de 20 personas a un total de 40 millones.

Pues Final Gather funciona de forma similar. Para calcular la cantidad de luz indirecta

se toman una serie de muestras de la escena. Si tomamos muy pocas muestras, el resultado se ajustará poco a la realidad, pero el cálculo será rápido. En hacer 20 encuestas solo tardaría unos minutos... Por el contrario, si queremos más realismo debemos tomar más muestras. A mayor número de muestras, mas se parecerá el resultado de nuestra encuesta al resultado real. Pero más tiempo de cálculo. Imaginaros el tiempo que me llevaría encuestar a 1 millón de personas (y más para preguntar semejante chorrada...)

La cantidad de muestras se especifica con "Initial FG Point Density". Con el predeterminado "Draft" el valor es de 0,1, muy pocas muestras.

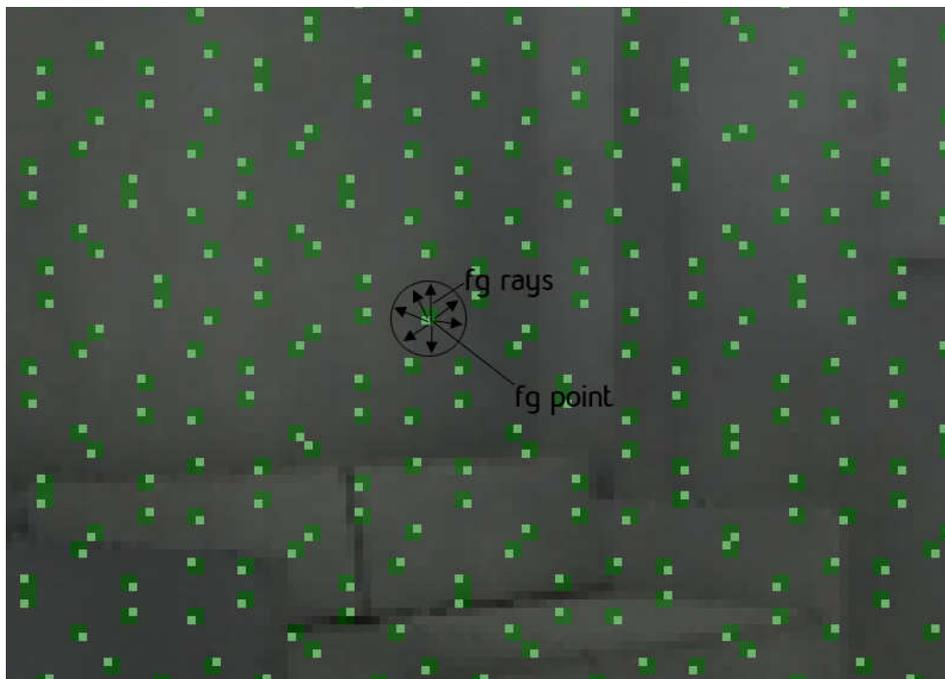
Vamos a ver las muestras que está tomando final gather en nuestras escenas.

En render setup (F10) vamos a la persiana "processing". En la persiana "Diagnostics" activamos "Enable" y marcamos la opción "Final Gather". Tiramos un render en las dos escenas.



Cada puntito verde es una muestra de FG, o FG point. Si os fijáis bien, la cantidad de FG points para el mismo valor de initial FG point Density es muy similar, independientemente de la cantidad de luz. Lo único que si podemos apreciar es que en las zonas donde hay luz directa se acumulan algo menos. Vemos además que la iluminación presenta manchas, es poco precisa, generando una luz plana e incorrecta.

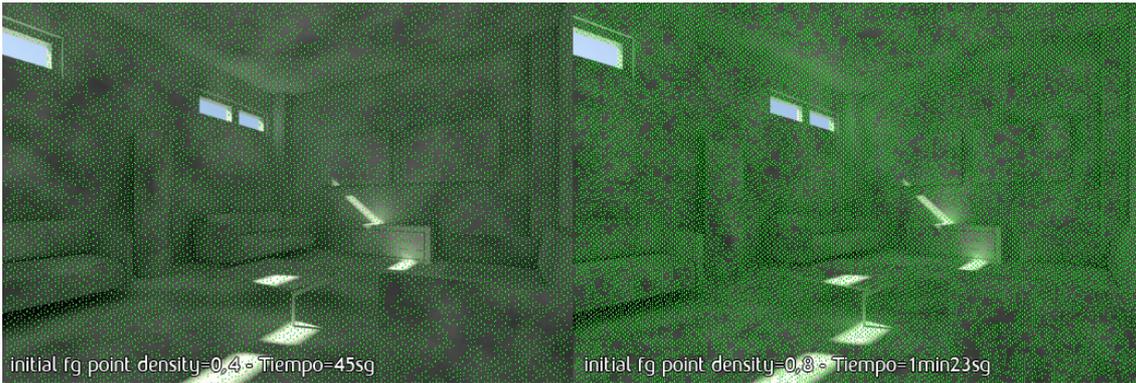
Para calcular la luz de la escena, FG toma puntos de FG de la geometría de la escena DESDE LA CÁMARA, es decir, solo se calcula la luz indirecta del encuadre de nuestra cámara. Lo que queda fuera del encuadre no se calcula. Esto es importante sobre todo si vamos a animar. A continuación, alrededor de cada punto de FG, se recolecta la cantidad de luz indirecta alrededor de ese punto. Esto lo hace FG disparando rayos. Cuando los rayos disparados desde cada punto impactan en la geometría próxima recogen, recolectan esa cantidad de luz y se sombrea la zona alrededor del punto de FG en cuestión.



Cuando hay muy pocos puntos de FG, las zonas de iluminación indirecta alrededor de esos puntos, cuyo color calculan los rayos de FG disparados desde ese punto, no se solapan y por eso aparecen imprecisiones y manchas. Además, como hay pocas muestras, pocos puntos, el resultado es pobre e impreciso.

Vamos a subir los valores de initial fg point density y vamos a comparar.

En ambas escenas tiramos dos render, initial fg point density a 0,4 y 0,8



Pues ya lo tenemos. Ahora hay más puntos de fg, y aunque no lo vemos aún por esos malditos puntos verdes, nuestras escenas tienen mejor apariencia aunque todavía presentan muchas manchas. Fijaros como los puntos de FG se concentran en las zonas "conflictivas": esquinas, zonas que quedan en sombra, es decir, FG trabaja para dar más detalle a las zonas que reciben solo luz indirecta.

Y fijaros en otra cosa: como arrastramos un número de diffuse bounces alto en la escena pocaluz, los tiempos de render como habíamos predicho se multiplican x4 con todo lo que eso conlleva. Chungo.

Voy a desactivar diagnostics para ver el render final con 0,8 fgpoint density en ambas escenas.

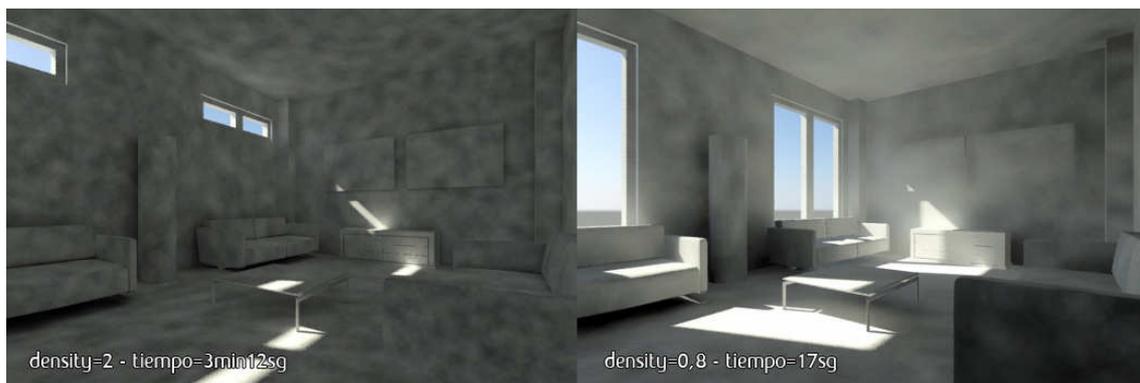




Sin los dichosos puntos verdes ahora vemos mejor las diferencias. Y nos damos cuenta de otra cosa:

- Ambas escenas tienen los mismos valores de densidad, es decir, calculamos con el mismo detalle la luz indirecta a través de FG. Vemos en ambos casos mayor detalle que con 0,1 y 0,4, unas luces y sombras mejor repartidas y más correctas. Sin embargo, y repito con la misma densidad, la imagen muchaluz presenta mejor aspecto que pocaluz. Menos manchas y más suavidad. ¿Debemos usar más FG points, o sea mas initial fg point density en pocaluz que en mucha luz o dependerá de otro valor?

Vamos a probar. En el archivo pocaluz le ponemos Initial FG point density 2. Más del doble que hemos usado antes, 0,8. Si la densidad influye en la suavidad y elimina las manchas debería notarse, ¿no?



Pues parece ser que no... la imagen de la izq, pocaluz, está renderizada con una densidad de FG de 2, mientras que la de la derecha es muchaluz con densidad 0,8, es decir, la misma imagen del post anterior. De los tiempos ni hablamos... 3 minutos más para la primera.

Observamos que las manchas no han desaparecido. La imagen muchaluz con una densidad inferior se ve más suave, tiene menos manchas. En lo que si hemos ganado al subir la densidad a 2 en poca luz es en DETALLE. Si nos fijamos bien, vemos como los

objetos, sobre todos los de la pared del fondo, aparecen mejor encajados y más definidos. Por ejemplo, los límites de los dos cuadros del fondo aparecen bien contorneados mientras que en la imagen muchaluz están muy difuminados y casi camuflados con la pared. Si comparamos esta imagen de poca luz con density 2 y la anterior también de poca luz con density 0,8 nos daremos cuenta de lo mismo.

Luego al aumentar la densidad de puntos de FG, lejos de suavizar la escena conseguimos realzar el detalle en la iluminación indirecta de los objetos y concentrar los puntos de FG en los límites de los objetos, zonas de sombra y pequeños huecos de nuestra escena, ganando detalle y definición.

¿Cuándo necesitamos una densidad alta? Si nuestra escena tiene muchos objetos, intersecciones entre superficies, huecos y recovecos escondidos a la luz, entonces necesitamos más densidad de puntos de FG para sacar el detalle en esas zonas conflictivas. Si tenemos una sala con 4 paredes y una esfera grande en el centro, con una densidad entre 0,1 y 0,4 será más que suficiente para conseguir el detalle necesario.

Entonces, ¿cómo eliminamos las manchas en esta escena?

Antes hemos dicho que FG toma una muestra de la geometría de nuestra escena. Eso es un FG point. La forma de calcular y representar la cantidad de luz alrededor de ese FG point es disparando rayos. FG dispara rayos desde el FG point y esos rayos recogen la cantidad de luz depositada alrededor del FG point, sombreando la zona. Cuanta más rica y definida sea la cantidad de luz recogida, cuanta más información de luminosidad sea procesada, mejor iluminada quedará esa zona alrededor del FG point.

Ahora bien, esa cantidad de luz recolectada depende de la cantidad de luz que haya en la escena. Puede suceder que si nosotros disparamos un número de rayos en una escena oscura, no recojamos la misma cantidad de información que disparando los mismos rayos en una escena bien iluminada.

Si la cantidad de luz recogida por los rayos de FG no es suficiente nos encontraremos con un área alrededor del punto de FG con muy poca información de luminosidad, rellena prácticamente con un color sólido!

Vamos a empezar a probar con muchaluz:

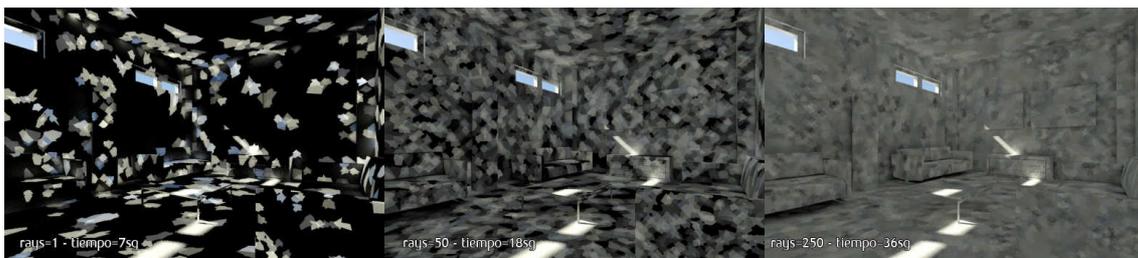
Initial fg point density: 0,1
Rays per FG point: 1 - 50 - 250
Interpolate over num fg points: 1

El valor de interpolación lo ponemos en 1 para ver nuestros puntos de FG de forma discreta, sin interpolar. Luego explicamos este valor.



He utilizado un valor de densidad bajo para que veáis perfectamente esas zonas alrededor de los FG points. Efectivamente, con un 1 rayo vemos unas manchas rellenas de color sólido, perfectamente definidas. Con 50 rayos cada punto recoge más cantidad de luminosidad y el sombreado es algo mejor, aunque se siguen apreciando perfectamente los contornos de las áreas iluminadas. Con 250 los contornos se empiezan a difuminar al aumentar la riqueza de la iluminación dentro del área alrededor de cada FG point.

Si usamos los mismos valores con pocaluz el resultado debería ser similar. Aunque ya hemos dicho que no va a ser así... Renderizamos pocaluz con density=0,1; rays=1-50-250 e Interpolate over...=1



Si nos fijamos solo en las tres imágenes de poca luz efectivamente se nota el resultado entre ellas; a mayor número de rayos mejor se sombrea esas zonas alrededor de los puntos de FG. En la de 250 rayos recogemos más luz y por tanto se empiezan a difuminar las áreas que en las otras dos imágenes se ven como manchas nítidas y bien contorneadas.

Pero si comparáis las tres imágenes de mucha luz con las de pocaluz, nos damos cuenta que pocaluz siempre sale perdiendo al mismo número de rayos. En mucha luz con 250 rayos la iluminación se ve mejor que en pocaluz.

La clave la hemos dado antes: la cantidad de luz en escena. En mucha luz hay más luminosidad, luego hace falta disparar menos rayos para conseguir mejores resultados. Como en pocaluz hay menos luz, debemos disparar más rayos para obtener EL MISMO resultado que en mucha luz.

Así que ya tenemos dos handicap que afectan al tiempo de render de pocaluz:

- 1.- Los diffuse bounces. Al necesitar más rebotes para iluminar mejor, aumenta el tiempo de render (en nuestro caso, x4).
- 2.- Los rayos. Al necesitar más FG rays para suavizar la escena en pocaluz, también aumenta el tiempo de render. Y también bastante.

Para manipular los rayos, por lo tanto, tenemos que tener en cuenta la cantidad de luz de la escena:

- En escenas con mucha luz artificial (un interior con varias fuentes de luz, potentes y bien repartidas) podemos usar entre 50-100 rayos.
- En escenas de contraste medio (como mucha luz) entre 500-2500 rayos.
- En escenas de bajo contraste, con luces artificiales muy tenues o entradas de luz natural muy pequeñas entre 5000-10000 rayos.

Como las dos escenas tienen los mismos objetos podemos usar la misma densidad de FG points en ambas. Pero el número de rayos no puede ser el mismo si queremos suavizar ambas por igual.

Con lo que hemos visto hasta ahora vamos a hacer una primera prueba de render con calidad alta. Aunque ya podemos deducir que si ponemos el mismo número de rayos en mucha luz y poca luz, ésta última se verá peor y tardará más en renderizarse por usar 12 rebotes de FG, vamos a probar a ver si esa diferencia se nota mucho o poco cuando usamos un número muy elevado de rayos. Y también si la diferencia en el tiempo de render se nota...

En ambas escenas:

- FG point density=2 (detalle medio-alto, pero el mismo para las dos escenas pues son iguales en cuanto a modelado)
- Rays per FG point 2000
- Interpolate Over num FG Point=30

CUIDADO: LA IMAGEN POCA LUZ CON ESTA CONFIGURACIÓN HA TARDADO 46 MINUTOS



Pues ahí lo tenéis, mientras que mucha luz ha tardado "solo" 8 minutitos en renderizarse tiene bastante mejor apariencia, muy suave aunque aún queda alguna manchita apreciable. Podríamos usar algunos rayos más para terminar de suavizar sin perder mucho tiempo de render. Pero poca luz... buf, 46 minutos y se siguen notando manchas apreciables aunque, claro está, mejor que en las pruebas anteriores con 250 rayos.

Voy a tirar la casa por la ventana. Render de mucha luz con 3000 rayos y de poca luz con 10.000 rayos!!!

ATENCIÓN: EL RENDER DE POCALUZ CON ESTA CONFIGURACIÓN HA TARDADO CASI 3 HORAS!!!



Pues ahí lo tenéis. Muchaluz se ve francamente bien, 12 minutos de render no es ninguna barbaridad teniendo en cuenta que solo estamos usando una luz y solo estamos trabajando con Final Gather. Pocaluz con 10.000 rayos también ha ganado muchísimo (estaría bueno) pero claro, a costa de un tiempo de render brutal. Tened en cuenta que a la diferencia de tiempo que suponía el número de rebotes adicionales hay que sumarle que estamos usando ahora 3 veces más rayos por FG point.

Nos falta por explicar el valor de Interpolate Over Num FG points. Quizá sea el concepto más fácil de entender de FG y el que menos influye en los tiempos de render.

Cuando calculamos la cantidad y calidad de la luz alrededor de cada punto de FG usamos más o menos rayos de FG. Pero incluso con un número elevado de rayos suavizar la iluminación calculada por FG sería muy complicado, pues siempre se notarían los límites del área alrededor de cada FG point.

Interpolate es un filtro de desenfoque o mediana que promedia la iluminación calculada no para un punto de FG, sino para un grupo de ellos. De esa forma se suavizan y desenfocan los contornos de las áreas alrededor de los puntos de FG. Es decir, si tomamos un punto de FG de referencia y ponemos un valor de Interpolate de 5, por ejemplo, se toman los 5 puntos de FG con sus respectivas áreas y se promedia la iluminación de esa zona, ahora más grande, con los datos de esos 5 puntos de FG, generando un desenfoque y suavizando toda esa zona. Y así con todos.

Usar valores altos de Interpolate sería parecido a usar un desenfoque gaussiano en Photoshop con un valor alto y la consecuencia sería una zona de iluminación desde luego muy suave, pero al mismo tiempo plana y sin detalle. Es decir, lo que ganamos aumentando la densidad lo perderíamos por interpolar en exceso. Si el valor es bajo, volverían a aparecer las manchas.

Para valores de densidad medios y bajos un valor de 30 es suficiente. Si usamos una densidad mayor podemos elevar un poco el valor de interpolate. Esto es porque si la densidad es muy alta y el valor de interpolate es 30, por ejemplo, como los puntos están muy juntos está claro que los 30 puntos más cercanos a un FG point cualquiera cubrirán una zona muy pequeña; mientras que si la densidad es baja esos mismos 30 puntos estarán más separados y la zona cubierta será mucho mayor.

Si tenéis alguna duda sobre interpolate over num FG points con esta explicación (no me gustaría enrollar mucho) podéis consultar el tutorial [Mental Ray interior](#) donde está también explicado este concepto.

Hasta aquí nuestra andadura solo con Final Gather. Hemos demostrado como las condiciones de luz afectan, y mucho, a los parámetros de configuración y por lo tanto a los tiempos de render, para obtener imágenes con iluminación indirecta de calidad.

Las escenas con muy poca luz directa necesitan un número de diffuse bounces muy elevado para repartir la luz convenientemente y un número de rayos también muy alto para conseguir una imagen suave y sin manchas. Los tiempos de render se disparan.

Sin embargo, para escenas con mucha luz podemos usar una densidad por ejemplo de 2 si no es una escena demasiado compleja, un solo rebote (o incluso 0 si hay mucha luz) y entre 100 y 250 rayos para conseguir una calidad muy elevada sin coste de procesamiento.

Pero no nos vamos a quedar aquí, claro. Queremos escenas con una buena calidad de render y tiempos contenidos. A pesar de que poca luz nos va a dar problemas queremos mejorar esa apariencia y sobre todo esos tiempos de render.

Tenemos dos opciones:

- Añadir más luz a la escena para ayudar a FG a procesar mejor y más rápido la iluminación.
- Usar Global Illumination, el otro método de Mental Ray para calcular la luz indirecta combinado con FG.

La primera opción la vamos a dejar para el final, pues mi intención con este tutorial es precisamente esa, no añadir más luz en principio para forzar a los dos métodos de cálculo (FG y GI) a tope a ver qué podemos conseguir. Así que vamos a trabajar primero con Global Illumination (GI) y nuestros amigos los fotones.

Por cierto, pequeño inciso. Para los novatos, **CUIDADO CON LOS PRESET DE FINAL GATHER**. Esta guay, aparentemente, poder usar unos valores preestablecidos y catalogados como "draft", "Low", "medium", "high" y "very high". Pero si movemos la pestañita deslizante y comprobamos los valores de densidad y rays, veremos que las distintas combinaciones pueden no ser adecuadas para mi escena en ningún caso. Tanto por quedarse cortas como por pasarse por arriba.

Por ejemplo, si tenemos una escena muy bien iluminada y con un modelado muy complejo, ningún preset hasta el "very high" puede ofrecernos una densidad adecuada. "High" coloca el valor de densidad en 1,5, el cual puede ser insuficiente. "Very high" lo pone en 4, bastante alto y puede que adecuado a mi escena, pero tira 10.000 rayos!! Si mi escena tiene estas características podría incluso pasar de 4 en densidad para ganar detalle y sin embargo usar solo 250 rayos para suavizar la escena.

Ahora si, vamos con GI.

Global Illumination usa fotones para calcular la cantidad de luz indirecta que se reparte

por la escena. Estos fotones rebotan en la geometría de la escena depositando energía que es recogida por GI para el cálculo de la luz.

La gran ventaja de GI es la rapidez con la que reparte la luz indirecta. Enseguida vais a ver que solo con GI el resultado en cuanto a cantidad de luz en ambas escenas es mucho más luminoso que usando Final Gather con varios rebotes, sobre todo en poca luz. Y lo mejor, muy rápido. Luego gracias a GI vamos a deshacernos de uno de los problemas de la escena poca luz: usar un número de rebotes alto para iluminar bien aumentando el tiempo de render.

La desventaja es que conseguir una escena suavizada con GI es complicado. Hay que usar un número elevado de fotones para conseguir la misma suavidad en la iluminación que con Final Gather.

Pero si combinamos ambos, tendremos lo mejor de cada casa.

Además GI se calcula PARA TODA LA ESCENA, independientemente del encuadre de la cámara.

Primera prueba. En el panel de Render Setup (F10) desactivamos "enable" en Final Gather. Luego lo volveremos a activar. Un poco más abajo, en la persiana "Caustics and Global Illumination" activamos la casilla "enable" en el panel de GI (no en caustics, cuidado).

Los valores de GI por defecto son:

- Maximum num Photons per sample: 500
 - Maximum sampling radius: desactivado
- Un poco más abajo:
- Average GI Photons per light: 20.000

Estos son los valores principales y más importantes. Tendremos que ajustar el control de exposición de mr pues va a cambiar la cantidad de luz. Tiramos un render de cada escena con estos valores de GI.





Estupendo! en apenas unos segundos tenemos gran cantidad de luz, pocaluz se ve un poco más oscura que lo que nos ha salido con FG y 12 rebotes. Ajustando el control de exposición conseguiremos los mismos resultados. Y los cálculos han sido muy rápidos.

Ahora vamos a darle un poco más de calidad al tema. Como vamos a combinar GI y FG NO ES NECESARIO OBTENER UN RENDER PERFECTO SOLO CON GI, sino simplemente correcto. FG ya se encargará de suavizar.

Otro de los problemas de GI es que ajustar los parámetros no es fácil. Aquí no se cumple la norma de "cuantos más fotones, mejor se va a ver todo", hay que encontrar la combinación adecuada entre los fotones emitidos, los fotones que queremos procesar y en qué radio queremos hacerlo.

Los valores que tenemos son:

Max. num. photons per sample=500
Max. sampling radius=desactivado
Average GI photons per light=20.000

Traducido al castellano (más o menos) esto quiere decir:

- el sol (la luz que tenemos en escena) dispara 20.000 fotones. De esos 20.000 muchos se colarán por las ventanas y rebotarán en los objetos de nuestra escena, depositando luz indirecta.

-Para recolectar esa luz, dividimos la escena en trozos del radio "max. sampling radius". Como este valor está desactivado, automáticamente mental ray utiliza un radio equivalente a 1/10 el radio de una esfera que supuestamente encerrara a nuestra habitación. En nuestro caso ese radio sería de unos 40 cm.

- Una vez dividida la escena en trozos de 40 cm, buscamos dentro de cada trozo 500 fotones rebotados (max num photons per sample) y procesamos la cantidad de luz.

Pero claro puede ocurrir:

Imaginaros que yo lanzo a la escena desde el sol 10 fotones solamente. De esos 10, solo 8 se cuelan por la ventana y se ponen a rebotar por la habitación. Divido la escena en

trozos de 40 cm y le pido que encuentre 500 fotones en cada trozo. ¿Los encontrará? Pues hombre, va a estar jodida la cosa... si solo hay 8 fotones en total rebotando difícil va a ser que encuentre 500... Quizá encuentre 2. O uno... pero no 500

Otro caso. Lanzo 100 millones de fotones. 90 millones se cuelan por la ventana. 90 millones rebotan por la habitación. Divido en trozos de 40 cm y le pido (max num photons per sample) que encuentre 2 fotones en ese radio. ¿Los encontrará? Estaría bueno! Pero claro, hay un montón de fotones que no estamos teniendo en cuenta, no se usan! Es más, el resultado visual del primer ejemplo y el segundo sería el mismo! encontramos los mismos fotones.

El problema es que en primer caso la emisión de 10 fotones tardaría una fracción de segundo, emitir 100 millones de fotones puede tardar horas.

Luego es importante encontrar el equilibrio adecuado entre el número de fotones buscados, el radio de búsqueda y los fotones totales emitidos.

Ahora mismo le estamos pidiendo 500 fotones en un radio de unos 40cm, lanzando 20.000. ¿Como sabemos si encuentra los 500 fotones? comparando.

Vamos a renderizar buscando 500, 250 y 100 fotones:

Max num photons per sample: 500-250-100

Maximum sampling radius: 45 cm

Average GI photons per light: 20.000



Solo os pongo los resultados con pocaluz. Tranquilos, a muchaluz le pasa lo mismo: las tres imágenes son IGUALES. ¿Que significa esto? que tirando 20.000 fotones y buscando en un radio de 45 cm encuentra LOS MISMOS FOTONES. Es decir, no encuentra 100 y por lo tanto no encontrará jamás 250 ni 500.

El valor de max num photons per sample TIENE que ser inferior a 100. Cuando empezamos a notar cambios (a peor) en la imagen entonces hemos encontrado cuántos fotones de verdad se localizarán en un radio de 45 cm disparando 20.000. Una vez localizado ese valor, todo será más fácil.

Vamos entonces a ver cuantos fotones encontramos como máximo en nuestro radio de 45cm.

max. num photons per sample=75-50-25

Max. sampling radius=45cm

Aver. GI photons per light=20.000



Ahora vemos cambios. La imagen donde hemos buscado 25 fotones presenta manchas muy definidas. Si vemos la imagen con 50 fotones vemos que esas manchas, a pesar de ser muy notables, están también bastante más suavizadas que en la imagen con 25 fotones.

¿Qué significa esto? Pues que hemos pedido 25 fotones y los ha encontrado. Pero le hemos pedido 50 y la imagen se ve mejor, luego había más de 25 fotones en ese radio de 45cm.

¿Habrán 75? pues si comparamos las imágenes con 50 y 75 también apreciamos un cambio, las manchas están ligeramente más difuminadas, no se nota tanto el cambio como entre 25 y 50 pero hay algo de mejoría.

El valor de max. num. photons per sample especifica cuántos fotones queremos tener en cuenta para calcular la iluminación indirecta en cada zona del radio especificado en max. sampling radius. Cuantos más fotones encontremos, más suave y mas calidad tendrá la solución de GI. El problema está, como habéis visto, que yo no puedo tirar un número muy bajo de fotones desde Aver. GI photons per light y luego pedirle 5000 fotones en Max. Nun photons per sample. No los va a encontrar. Tampoco los encontraría si buscara en un radio muy pequeño, cuanto menor sea el radio obviamente menos fotones caben.

Luego, para nuestra escena, dado el radio y el nº de fotones que lanzamos, de momento podemos decir que como mínimo encontramos 75 fotones.

En el post anterior comparamos entre 100-250 y 500 y vimos que las 3 imágenes eran iguales. Luego el número de fotones que buscamos debe estar entre 75 y 100.

Vamos a comparar las dos imágenes; las tenemos, no hace falta renderizar.



Como véis son muy muy similares. No apreciamos cambios a simple vista. Si notáramos algún cambio, ¿que querría decir? pues que encontramos más de 75 fotones. Habría que volver a probar con 80, 90, 95.. por ahí estaría el valor. En nuestro caso el número de fotones estará próximo a 75. Pueden ser 70 o 73, pero más de 75 no. Así que nos vamos a quedar con este número de fotones, fotón arriba fotón abajo.

Luego para poca luz, si tiramos 20.000 fotones y buscamos en un radio de 45cm, encontramos 75 fotones como máximo. El tiempo de render es de 7sg.

Para mucha luz, probamos igualmente hasta encontrar ese valor. Buscamos 25-50-75 fotones.



Pues... lo mismo. Encontramos los mismos fotones en el mismo radio en ambas escenas, así que nos quedamos también con ese valor de 75 fotones encontrados para un radio de 45cm, lanzando 20.000 fotones desde el sol.

¿Y ahora qué? Pues antes hemos comentado que cuantos más fotones encontremos en max. num. photons per sample, mejor apariencia tendrá la escena. Pero ahora mismo encontramos como máximo 75. ¿Cómo podemos encontrar más fotones? Pues esta claro que para encontrar más fotones solo podemos hacer 2 cosas:

- Lanzar más fotones. Si desde la luz emitimos un mayor número de fotones, habrá más fotones rebotando por la habitación y podremos encontrar más de 75 para nuestro radio de 45cm ¿no?

- Ampliar el radio. Está claro que si en 45cm encontramos 75 fotones, en 100cm habrá más... El problema de ampliar el radio es que si cada vez buscamos en zonas más grandes perderemos detalle en la iluminación, llegando a quedar plana si nos pasamos. En ese sentido os recuerdo cómo calcula automáticamente Mental Ray el radio idóneo para la escena: 1/10 el radio de una esfera que envolviera nuestra escena. Luego cada escena tiene su propio radio, y es conveniente dejarlo como está en la mayoría de las ocasiones.

Así que para suavizar la escena y conseguir mayor calidad, dejamos el radio y lanzamos más fotones. Pero, ¿cuántos más? ¿y cómo se ahora cuántos fotones voy a encontrar en max. num photons per sample? Es decir, se trata de mantener siempre el número de fotones encontrados al máximo sin pasarme ni quedarme corto, si lanzamos el doble de fotones en average GI photons per light, ¿cuántos fotones más vamos a encontrar?

Fácil. Si buscamos el doble de fotones, debemos tirar 4 veces más fotones en average GI photons per light.

Voy a empezar con muchaluz. Teníamos 75 fotones en la búsqueda, lanzando 20.000. Vamos a pedirle 150 lanzando 80.000.



Bien bien... se nota. Poco, pero se nota. Al encontrar más fotones, la imagen se empieza a ver mejor, menos manchada y más homogénea. ¿Cómo se que la regla que os he dado antes (fotones buscados x2 y fotones lanzados x4) es correcta? Probadlo. Si lanzando 80.000 fotones, le pido 100 en lugar de 150 y la imagen se ve peor, significa que si encuentra esos 150 fotones (o al menos más de 100) Y si lanzando 80.000 fotones le pido 200 en lugar de 150 y la imagen se ve igual, entonces no encuentra más de 150. Si se cumplen las dos pruebas, entonces la norma es correcta.

De verdad, probadlo vosotros, es la única forma de aprender.

Seguimos nuestra progresión con muchaluz. Vamos a hacer 3 render más:

- Max. num photons per sample: 300-600-1200
- Avrg. GI photons per light: 320.000-1.280.000-5.120.000



Cada vez mejor. Y los tiempos de render son muy rápidos, fijaros en emitir 5 millones de fotones hemos tardado 53sg. Todavía quedan manchas pero veis que a medida que encontramos más fotones los detalles se van concentrando más y la imagen es cada vez más homogénea y suave.

El problema de GI es que le cuesta mucho suavizar la imagen. Para obtener un render óptimo habría que seguir lanzando fotones y encontrando más y mas fotones. El siguiente paso sería lanzar 20 millones (los 5 millones x4). Si no queda bien habría que tirar 80millones... buf.

De todas formas, hemos dicho que íbamos a combinar GI con Final Gather para obtener resultados óptimos, ¿no? Asi que no tenemos que dejar la imagen perfecta solo con GI, no es necesario. Necesitamos una solución de fotones que nos proporcione el suficiente

detalle y que no presente graves imperfecciones.

Y no solo eso. Si a Final Gather le damos un mapa de fotones demasiado detallado lo pasa mal. Tened cuenta que cuando trabajemos con FG y GI, primero calculamos la GI y luego FG se calcula SEGÚN EL MAPA DE FOTONES que hayamos calculado previamente. Luego para los mismos valores de FG nos podemos encontrar con resultados distintos en función de cómo hayamos calculado la GI.

La pregunta clave es: ¿como de detallado (cuantos fotones) tiene que ser GI para obtener buenos resultados con FG?

Pues vamos a probar.

En la ventana de render setup (F10), en la pestaña "indirect illumination" vamos a activar "enable" en Final Gather. Es importante usar los mismos valores de calidad de FG para poder comparar. En este caso uso valores medio-bajos, pues solo quiero comprobar si la imagen mejora cuando usamos los mismo valores de FG combinados con tres mapas de GI distintos:

FG: density=0,4 - Rays=250 - Interpolation=30

Con estos valores de FG tiramos 3 render con los siguientes valores de GI:

Max num photons per sample: 75 - 300 - 600
Average GI photons: 20.000 - 320.000 - 1.280.000

A ver que pasa....



Vaya, resulta que la primera imagen, la del mapa de fotones más pobre presenta varios artefactos, zonas luminosas que no deberían estar ahí. Pero las otras dos... son casi iguales. Lo que no es igual son los tiempos: la segunda ha tardado casi 3 veces más en renderizar. Si los valores de FG son iguales y el tiempo de emisión de fotones apenas se lleva 20sg más en la tercera que en la segunda, como habréis podido comprobar a FG le cuesta más trabajo procesar un mapa de fotones más complejo. Pero el resultado visual es el mismo.

Eso significa no solo que más fotones en la búsqueda sea igual sino que si lo vamos a combinar con FG es PEOR, a mas fotones más tiempo de render para obtener el mismo resultado.

Luego por lo tanto, no es necesario un mapa de fotones grande y suave, basta con que sea CORRECTO. El primer mapa es demasiado pobre y arrastra a FG a cálculos

incorrectos.

Y otra cosa. Ahora nuestra imagen es mejor. Mejor que solo con FG y mejor que solo con GI. Si recordáis, solo con FG una imagen aceptable (pero con manchas) se había ido a 12 minutos de render. Ahora tenemos una imagen suave en 1:35 si nos quedamos con la opción 2.

Con FG tuvimos que tirar 3000 rayos para conseguir una imagen más o menos suave. Y darle 2 rebotes. Con GI los rebotes de FG no tienen efecto, y al haber más luz (gracias a GI) no hace falta usar tantos rayos para suavizar.

Lo que si hemos perdido es detalle. Los bordes de los objetos se difuminan. Recordad que estamos usando una densidad baja en FG, 0,4, poco detalle. Vamos a darle un poco más de caña a ver si sacamos más detalle. Si no, tendremos que usar Ambient Occlusion.

FG: Density=2 Rays=250 Interpolation=30

GI: Max num photons per sample=600 - Average GI photons=1.280.000

Tiempo de render: 4:25



Al aumentar la densidad de FG points, aumenta el detalle. Claro, también ha aumentado el tiempo de render, pero no es ninguna barbaridad, 4:37. Todavía podemos sacar más detalle, y para ello podemos usar dos métodos:

- Seguir aumentando la densidad de FG points a costa de mayor tiempo de render.
- Usar Ambient Occlusion a nivel de material para controlar las sombras de contacto sin aumentar demasiado los tiempos.

He tirado una primera prueba con density 4. El tiempo de render ha sido de 6:52, pensé que iba a tardar más al subir la densidad! Tenemos algo más de detalle, sobre todo en los cuadros y en la columna de la derecha, arriba.

Así podríamos seguir aumentando la densidad a costa del tiempo de render. Sería interesante ver que nos ofrece Ambient Occlusion



Antes de pasar a ver Ambient Occlusion, debemos dejar la escena suave y con la mejor calidad posible. Ahora hemos conseguido gracias a GI disminuir los tiempos de render (lo vamos a notar sobre todo con pocaluz, pues usaba 12 rebotes y muchísimos FG rays) pero todavía aparecen algunas manchas, sobre todo en el techo.

Incluso con una densidad de 4 (valor bastante alto) no conseguimos eliminar esas manchas. De hecho, initial FG point density, como ya hemos comentado, sirve para generar detalle, no suavidad. Podemos:

- Aumentar el número de rayos. En este caso no llegaríamos a los valores que hemos usado solo con FG (2000 en muchaluz y 10000 en pocaluz) pues con GI tenemos más luz en la escena y no serían necesarios.
- Aumentar la interpolación. Recordar que aumentar en exceso la interpolación nos dejaría la escena muy suave, pero plana. Si subimos un poco podemos suavizar sacrificando algo de detalle, que luego volveríamos a sacar con Ambient Occlusion.

Además de los 3 valores principales de FG (densidad, rayos e interpolación) tenemos otros controles que quizá puedan ayudarnos. De ellos destacamos el parámetro NOISE FILTERING.

Noise filtering- Desestima rayos de forma selectiva de la solución de FG eliminando aquellos que son mucho más brillantes que los que están a su alrededor. Por ejemplo, si en una zona de nuestra escena la luz indirecta tiene una luminosidad media y de repente aparecen rayos bastante más brillantes, esta claro que ahí tendremos un problema de manchas. Con noise filtering podemos eliminar esas zonas más brillantes igualando la luminosidad y haciendo desaparecer las manchas, a costa de algo más de tiempo de render. Al eliminar esos rayos es posible que la escena quede algo más oscura, lo que corregiremos con el control exposición.

Use Falloff (Limit rays)- Nos permite especificar desde que distancia y hasta que distancia se calculan los rayos emitidos en cada punto de FG. Obviamente modifica el comportamiento real de la luz, pero para escenas conflictivas nos puede ayudar a minimizar la aparición de manchas reduciendo el alcance de FG.

La mejor solución es añadir más luz a la escena para que a FG le cueste menos procesar y suavizar. En ese caso usaríamos una MR Sky portal, pero eso lo veremos al final del todo para comparar. Recordar que nuestro objetivo es exprimir a FG y GI con una sola luz.

Podéis pensar que un mapa de fotones más detallado puede ayudar. Ya os digo que no, pero si queréis probar...

Voy a usar un noise filter alto y a limitar los rayos a la altura de la habitación. Os pongo los valores:

- FG Point density: 2
- Rays per FG point: 1000
- Interpolate: 40
- Noise filtering: Very high
- Activamos "use falloff": Start=0 Stop=375cm

El render son 12 minutos



En la imagen anterior usamos noise filtering para intentar eliminar las manchas y suavizar la escena, junto con un número de rayos alto (1000) y falloff (limit rays) para reducir el alcance de los rayos de FG. Vamos, de todo un poco a lo bestia. Usando una densidad de 4 la escena no pintaba mal del todo y el render era algo más de 6 minutos. En este caso el render se nos ha ido a 12, queda mejor, más suave (y la luz ha cambiado un poco, ahora se cuele más azul del entorno) pero el render está en 12 minutos.

Si limitamos los rayos y usamos noise filtering, ¿hacen falta 1000 rayos para suavizar la escena? Vamos a probar.

- Initial FG point density=4 (si tiramos un render antes y tardó 6 minutos... ¿por qué no probar?)
- Rays=250
- Interpolate=50 (recordar que al usar densidades altas conviene subir un poco la

interpolación)

- Noise filtering = High
- Use falloff (Limit rays)= start=0 stop=375

Tiempo de render: 5:25



Os pongo una comparativa de las últimas tres imágenes que hemos sacado. La primera, con un noise filter standar y sin atenuar los rayos (use falloff desactivado). La segunda, aumentando todos los valores para suavizar, el tiempo se fue a 12 minutos. La tercera, la última que hemos renderizado se ve prácticamente igual que la segunda en menos de la mitad de tiempo, 5:25.

Fijaros que hemos conseguido una escena muy suave e iluminada solo con la luz del sol y el cielo. Hemos tenido que trastear mucho con FG hasta conseguir el resultado que buscábamos sin unos tiempos de render exagerados. Pero todavía no hemos terminado, los contornos, las sombras de contacto de algunos objetos no están bien definidas. Lo podemos apreciar sobre todo en los cuadros, no se ve cual es el límite de su geometría sobre la pared, se confunde la sombra de los cuadros con la iluminación de la pared.

Para conseguir mayor definición en esas zonas donde la luz queda oculta entre los objetos que se tocan o están muy juntos, podemos:

- Aumentar mucho más la densidad. Nos tendríamos que ir a valores realmente altos, es decir computación pura y dura a costa de tiempo de render.
- Usar Ambient Occlusion.

Antes de usar ambient occlusion vamos a facilitarnos las cosas.



Lógicamente, mientras estamos probando y cambiando valores de FG tenemos que volver a calcularlo en cada render. GI también la estamos calculando en cada render, pero los valores ya los tenemos fijos. Ahora que ya hemos encontrado los valores de FG que queremos, ¿no podríamos guardar esos mapas para no volver a calcularlos mas? Pues si, de hecho lo vamos a hacer.

Podemos reutilizar los mapas de FG y GI siempre que:

- no cambiemos sus valores
- No movamos ni añadamos objetos en la escena
- No cambiemos las luces.

Y solo para FG

- No cambiemos el encuadre de la cámara.

En el panel de render setup (F10) nos vamos a la pestaña indirect illumination y bajamos a la persiana "Reuse (FG and GI disk caching)".

En MAX 2010 se han añadido nuevas herramientas para el cálculo de FG en animaciones que también afectan a cómo se guarda el mapa de FG. Lo tenemos en el primer desplegable. En nuestro caso usamos "single" pues estamos con un render estático.

En la sección "Final Gather Map" elegimos "Incrementally add FG points to map files". Esto hace que se guarde el mapa de FG. Debajo, pulsamos el botón de los tres puntitos (...) y elegimos donde y con qué nombre queremos guardar el mapa de FG.

Debajo, en "Caustics and Global Illumination Photon Map" elegimos "Read/Write Photons to map files". Debajo, pulsamos el botón de los tres puntitos (...) y elegimos donde y con qué nombre queremos guardar el mapa de fotones de GI.

Tiramos un render (también podéis pulsar el botón "Generate....now" bajo FG y GI) para calcular y guardar los mapas.

Logicamente, veremos lo mismo que en el último render pero habremos guardado los mapas.

Una vez guardados los mapas vamos a evitar que vuelvan a ser calculados. Bajo "final gather map" cambiamos la opción "incrementally add...to map files" por "Read FG points only from existing map files", bajo "Caustics and Global Illumination Photon map" cambiamos "Read/Write...." por "Read Photons only from existing map files".

Tiramos un render. Sorpresa, 2 sg de render.

Y ahora si, vamos con Ambient Occlusion (AO).

AO tiene que ver con la iluminación ambiental pero no se calcula ni en FG ni en GI, es una propiedad de los materiales y se calcula en el proceso de render. Lo que hace es, en aquellas partes de los objetos (que tengan un material con AO activado, claro) que están en contacto con otros objetos, genera una sombra de contacto para reforzar la que calcula FG y GI. Esto hace que los objetos queden mejor integrados y contorneados, eliminando la sensación de que un objeto parezca que esté "flotando" sobre otro o que no esté en contacto con él.

En el editor de materiales tenemos el único material que estamos usando, el de la primera ranura se llama blanco mate. Buscamos la pestaña "Special Effects" y ahí nos encontramos con AO.

- Activamos "Enable"
- Ponemos "max distance" en 25cm
- Renderizamos.



El resultado salta a la vista, en las zonas de contacto de unos objetos con otros se sombrea y contornea mejor todos los objetos. Sin embargo, a veces la sombra de AO puede ser excesiva y dar el cante. Podemos controlar, básicamente la distancia y el color de esa sombra.

-Con la distancia (max distance) reducimos el tamaño de la sombra de contacto a esa zona en concreto. Si aumentamos la distancia máxima entonces la sombra se extenderá, generando un degradado o sombra suave, desde la zona de contacto hacia el interior del objeto.

-Si cambiamos Shadow color podemos aclarar u oscurecer la sombra y hacer que "cante" menos.

Vamos a hacer un par de pruebas cambiando los valores de AO.

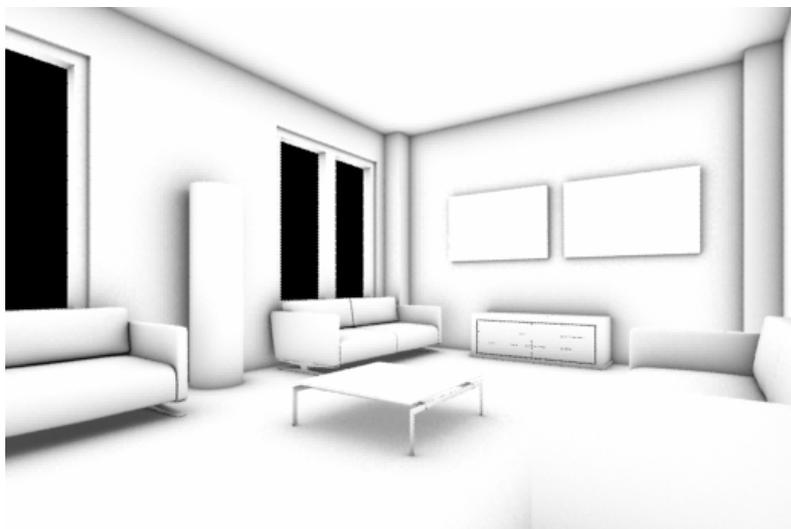
- Max distance= 5 cm - Shadow color=0,5(value)
- Max distance= 100 cm - Shadow color=0 (value)



Hemos puesto dos casos extremos de AO. El primero tenemos unas sombras pequeñas y concentradas en las zonas de contacto, con un color gris medio (value=0,5) y como véis apenas se nota AO. En el segundo caso el color de sombra es negro (value=0) y la distancia es de 100cm, es decir la sombra se extiende como máximo 1 metro desde la zona de contacto hasta el interior de los objetos (en los objetos pequeños lógicamente la sombra es más corta). En este caso, al usar una distancia grande, vemos ruido en el sombreado generado por AO. Para eliminar ese ruido aumentaremos el valor de "samples" (muestras) para que el degradado de sombra sea suave y sin ruido. A mayor distancia, más samples.

Luego iríamos graduando distancia y color hasta obtener el resultado deseado. Y renderizando. ¿no estaría mejor tirar un solo render y graduar AO de otra forma, por ejemplo en photoshop? Vamos a sacar un pase de render de AO y aplicar postproducción en photoshop.

- Primero vamos a tirar un render de nuestra escena sin AO.
- Luego activamos AO con valores altos de distancia y color (distance=100 y shadow color=0, samples=64).
- En el panel de render setup nos vamos a la pestaña "Render Elements".
- Pulsamos el botón "Add" y elegimos de la lista "mr A&D raw: Ambient Occlusion"
- Un poco más abajo tenemos un botón con tres puntitos (...) para elegir en que carpeta queremos guardar nuestro pase de AO.
- Renderizamos.



Con el render sin AO y esta imagen vamos en photoshop a graduar la cantidad de AO que queremos en la imagen final sin tener que renderizar. Lo que vamos a hacer (hay mil formas) es usar esta imagen en escala de grises para seleccionar en la imagen sin AO cuanto AO queremos usando, por ejemplo, el comando niveles en photoshop o cualquier otro que nos permita trabajar con la luminosidad de la escena.

- En photoshop abrimos el render sin AO y el pase de AO que hemos sacado antes.

Ok, abrimos Photoshop. Vamos a ver como usar el pase de AO sobre nuestro render para graduar la intensidad de las sombras de contacto. Se puede hacer de muchas formas, si controláis Photoshop seguro que se os ocurren unas cuantas.

Abrimos "muchaluz-noao.jpg" y "AOMap.jpg" en photoshop. En la imagen "muchaluz-noao.jpg" nos vamos a la paleta "Canales" y pulsamos el botón de "nuevo canal" (primera imagen). Se creará el canal Alfa 1 en la paleta. En ese canal vamos a pegar la imagen con el pase de AO: vamos a la imagen AOMap.jpg, seleccionamos toda la imagen (Ctrl+A) y copiamos (Ctrl+C). Volvemos a "muchaluz-noao.jpg" y en la paleta canales hacemos click en el canal "alfa 1". Debemos ver todo negro. Pulsamos pegar (Ctrl+V) para pegar en el canal alfa nuestro pase de AO.

Ahora vamos a seleccionar las zonas oscuras del canal con el pase de AO para usarlo sobre la imagen del render. Pulsamos la tecla control y hacemos click SOBRE LA MINIATURA DEL CANAL ALFA 1 (ver imagen 4). Veremos como se seleccionan las áreas blancas del canal alfa (imagen 5). A continuación vamos a usar esa selección, pero al revés (invertida para seleccionar las zonas oscuras) sobre nuestro render. Hacemos click sobre el canal RGB (IMPORTANTE: No activar el "ojo" (visibilidad del canal), hacer click sobre él -> IMAGEN 6) y nos vamos a la paleta "capas". Ahora debemos ver la selección sobre la imagen del render. Necesitamos invertirla para seleccionar las áreas de AO. Pulsamos Ctrl+May+I (o menú selección-> invertir) Ver imagen 7.

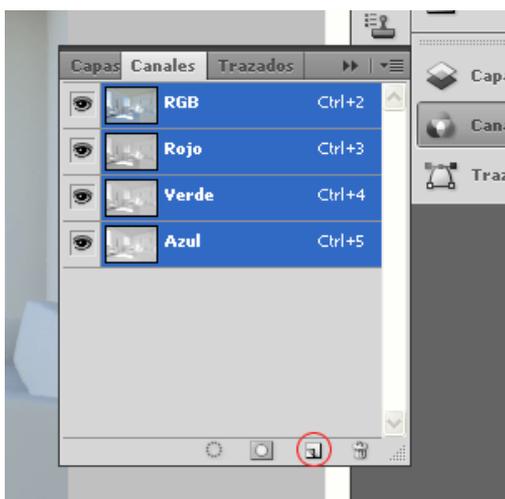


Imagen 1

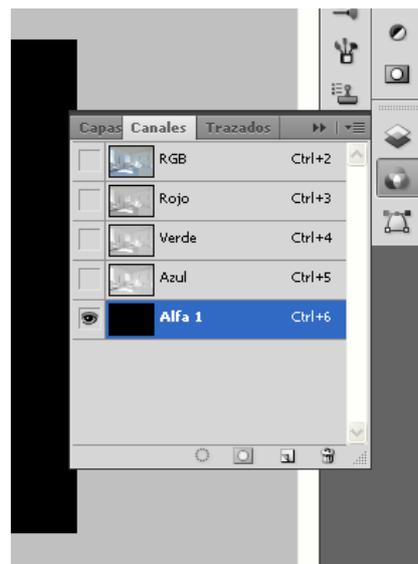


Imagen 2

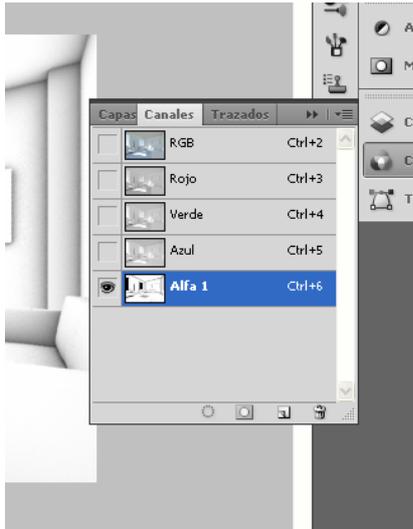


Imagen 3

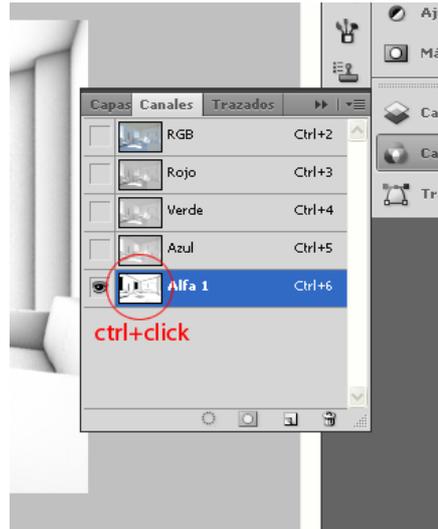


Imagen 4

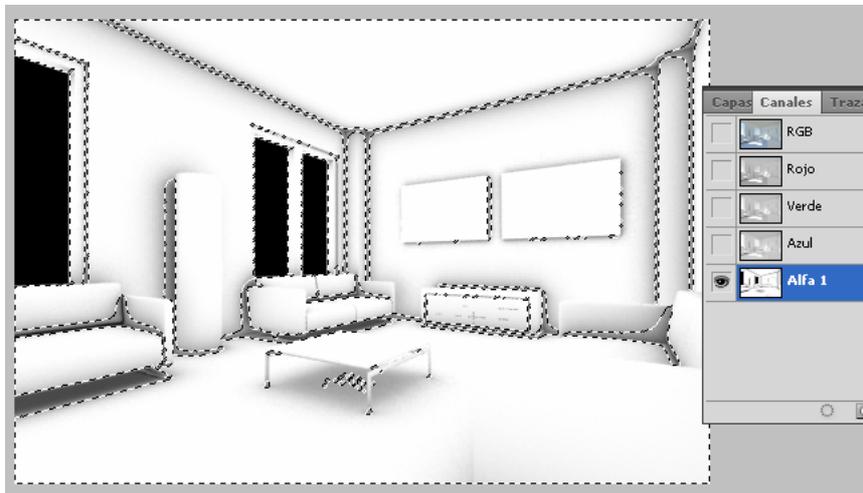


Imagen 5

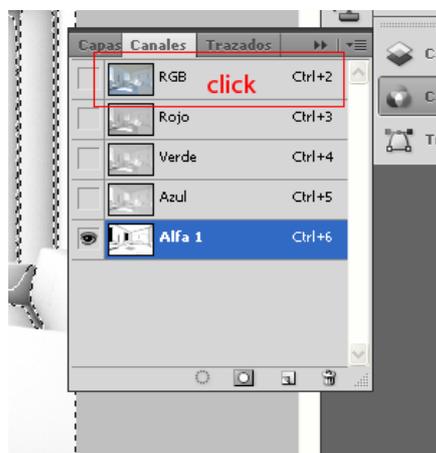


Imagen 6

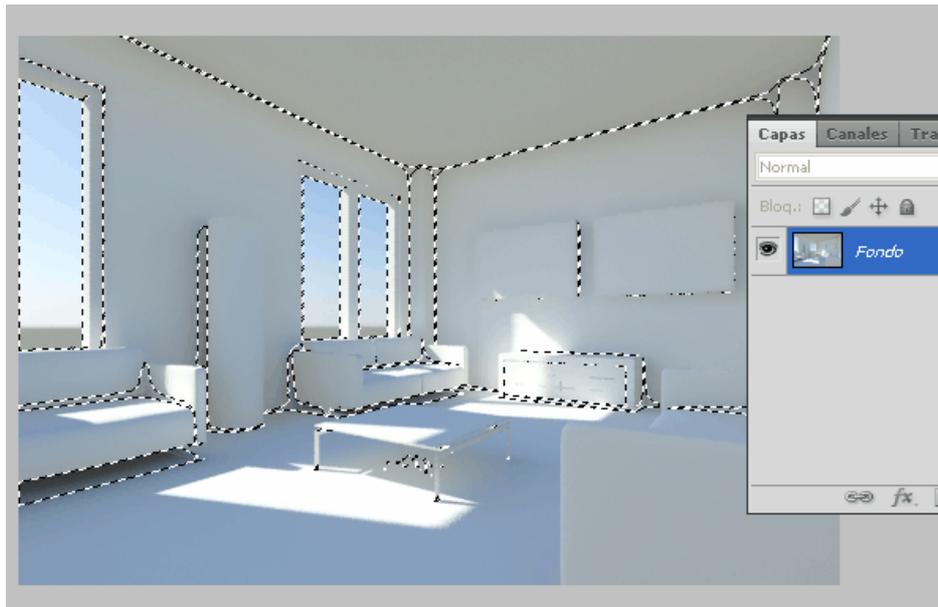


Imagen 7

Pues casi está. Una de las formas más sencillas de hacerlo es usar algún comando que me permita ajustar la luminosidad, como niveles, curvas o similar. Yo voy a usar el comando niveles. Para que no nos molesten los bordes de la selección pulsamos Ctrl+H y desaparecerán. Cuidado, aunque no los vemos sigue estando seleccionada una parte de la imagen. A continuación pulsamos Ctrl+L (o menú imagen->ajustes->niveles).

Ok, bajo el histograma (distribución de píxeles según su luminosidad) tenéis un triangulito gris en el centro. Si vamos moviendo el triangulito hacia la derecha vemos como las zonas afectadas por AO Y SOLO ELLAS se van oscureciendo. De esta forma podemos regular la intensidad del pase de AO con postproducción de forma sencilla.

Os pongo 3 ejemplos con 0,85, 0,75 y 0,5 en punto medio de gris.

