



SISTEMA CLÚSTER DE ALTO RENDIMIENTO PARA LA RENDERIZACIÓN DE ESCENAS TRIDIMENSIONALES FOTOREALISTAS CON TRAZADO DE RAYOS ABANCAY 2015

HIGH PERFORMANCE CLUSTER SYSTEM FOR THE RENDERING OF THREE-PHOTOREAL PHOTOREALIST SCENES WITH RAY TRACING, ABANCAY 2015

Carlos Yinmel Castro Buleje^{1,*}, Alejandro Apaza Tarqui², Edgar Eloy Carpio Vargas², Confesor Milan Vargas Valverde², Hugo David Calderón Vilca², Iván Soria Solis²

¹Universidad Nacional José María Arguedas, Facultad de Ingeniería de Sistemas, Jr. Juan Francisco Ramos N° 380, Ciudad Universitaria, Andahuaylas, Apurímac, carlosycb@gmail.com

²Universidad Nacional del Altiplano, Escuela de Postgrado, Av. Floral N° 1153, Ciudad Universitaria, Puno, Perú.

RESUMEN

El presente artículo plasma el estudio realizado sobre la renderización de escenas tridimensionales fotorealistas con clústeres de computadoras de alto rendimiento (HPC - High Performace Cluster) un conjunto de computadores que trabajan conectados a través de red para resolver problemas de alta complejidad computacional, el proceso de renderización con la técnica de trazado de rayos no es ajena a la complejidad computacional, esto debido a la gran cantidad de cálculos que se realiza para simular un ambiente físico casi perfecto y lograr niveles altos de fotorealismo, en tal sentido el objetivo fundamental del estudio realizado es determinar la disminución en el tiempo de renderización de escenas tridimensionales fotorealistas con la utilización clústeres HPC, la metodología utilizada aborda la implementación de un clúster HPC así como la renderización de escenas tridimensionales en cinco grupos experimentales y un grupo control, las escenas renderizadas corresponden al modelo tridimensional de la plaza de armas de la ciudad de Talavera, se tomó los tiempos de renderización de las escenas de los cinco grupos experimentales con 16, 32, 48, 64 y 80 CPUs, asimismo se renderizó las escenas del grupo control con 8 CPUs y para la validación de datos se utilizó la prueba estadística del ANOVA de un factor y la comparativa entre los diversos grupos de estudio con la prueba de T3 de DUNNETT, los resultados obtenidos muestran que la aplicación de clúster de alto rendimiento mejoran los tiempos de renderización hasta en un 56.61% con respecto al grupo control.

Palabras clave: Renderización con clúster, renderización paralela, renderización distribuida.

ABSTRACT

This article presents the study of three-dimensional photorealistic rendering scenes with clusters of high performance computers (HPC - High Performace Cluster) a working set of computers connected via network to solve problems of high computational complexity, the process of rendering with the technique of ray tracing is no stranger to the computational complexity, this due to the large number of calculations performed to simulate an almost perfect physical environment and achieve high levels of photorealism in this regard the fundamental objective of the study conducted is to determine the decrease in time photorealistic rendering of three-dimensional scenes with the use HPC clusters, the methodology addresses the implementation of an HPC cluster and rendering of three-dimensional scenes into five experimental groups and a control group for the rendered scenes dimensional model of the main square of the city of Talavera, render times the scenes of the five experimental groups with 16, 32, 48, 64 and 80 CPUs took also the scenes of the control group was rendered with 8 CPUs and data validation statistical test one-way ANOVA and the comparison between different study groups to test Dunnett's T3 was used, the results show that the application of high performance cluster improve render times up to 56.61% over the control group.

Keywords: Rendering with cluster, parallel rendering, Rendering distributed.

*Autor para Correspondencia: carlosycb@gmail.com





INTRODUCCIÓN

La generación de gráficos por computadora es un área de investigación cuyo objetivo es la creación de algoritmos y métodos que permitan generar una imagen sintetizada artificialmente en un computador (Cohen y Wallace, 1993), por otro lado, (Matt *et al.*, 2016) afirman que el realismo en graficación por computadora es definido como la producción de una imagen indistinguible de una fotografía del correspondiente lugar. Para lograrlo, es necesario simular la interacción de la luz con las superficies de acuerdo con las leyes de la física de la luz y los materiales así como el modelo de iluminación que es el cálculo de la intensidad de cada punto de la escena (Gonzales, 2009), (Fernandez, 2003) define la síntesis de imágenes como la disciplina que se dedica al estudio y desarrollo de procesos que permitan sintetizar imágenes a partir de modelos tridimensionales. En ese sentido producir imágenes tridimensionales fotorealistas computacionalmente requiere días, semanas y hasta meses de trabajo y ardua labor solo para diseñar estas escenas, este tiempo se amplía aún más para procesar las escenas y obtener el producto final, todo esto no se lleva en un único ordenador, ya que si ese fuese el caso, estas producciones fotográficas y cinematográficas tomarían demasiados años por la gran cantidad de cálculos necesarios para determinar la iluminación global, el Ray Tracing (Whitted, 1980), es el algoritmo de render más popular, fue presentado por Turner Whitted, y hoy en día lo poseen muchos motores de render que están incluidos en aplicaciones populares de diseño tridimensional, Según lo antes abordado es una de las razones por las cuales el objetivo de generar imágenes de manera eficiente aún se mantiene vigente, para resolver estos problemas nacieron los denominados supercomputadores, los cuales cuentan con arreglos de microprocesadores que trabajan en sincronía empleando procesamiento paralelo (Mesa y Branch, 2008). Por otra parte, estos son considerados como sistemas débilmente acoplados (Deitel, 1987), la computación paralela o procesamiento en paralelo consiste en acelerar la ejecución de un programa mediante su descomposición en fragmentos que pueden ejecutarse de forma simultánea, cada uno en su propia unidad de proceso. Surge así de forma natural, la idea de la computación paralela, que genéricamente consiste en utilizar “n” computadoras para multiplicar, idealmente por n la velocidad computacional obtenida de un único computador. Por supuesto esta es una situación ideal que muy rara vez se consigue en la práctica (Dormido *et al.*, 2003). Con la significancia de los avances tecnológicos hoy en día es posible utilizar y construir sistemas de cómputo compuestos por un gran número de computadoras interconectadas por medio de una red de alta velocidad y con la utilización de software especial, estos sistemas reciben el nombre de Sistemas Distribuidos (Tanenbaum, 1996), por otra parte un sistema distribuido es una colección de procesadores que no comparten ni memoria ni una señal de reloj. En lugar de ello, cada procesador tiene su propia memoria local. Los procesadores se comunican entre sí a través de varios tipos de líneas de comunicaciones, como buses de alta velocidad o líneas telefónicas (Gorlatch *et al.*, 2008), Según (Hai *et al.*, 2002) un clúster es la interconexión de dos o más máquinas basadas en sistemas distribuidos que son débilmente acoplados a través de redes de alta velocidad, (Sun, 2006) “un clúster es un sistema distribuido por lo tanto posee sus características pero adicionalmente proporciona tres beneficios principales: alta disponibilidad, buena escalabilidad y bajo costo” y que aprovecha las redes y software libre existente (Torrealba y Martinez, 2002). Los clústeres de computadoras se clasifican en tolerancia a fallas, balanceo de carga y alto rendimiento (Castillo y Olivera, 2006), también un clúster es un conjunto de computadoras interconectadas con dispositivos de alta velocidad que actúan en conjunto usando el poder cómputo de varios CPUs en combinación para resolver ciertos problemas dados, a razón de ello los clústeres son una solución que muchos denominan “la supercomputadora de los pobres”. Generalmente la aplicación de los clústeres se ha hecho sobre las Aplicaciones de Grand Challenge (GCA), las aplicaciones Grand Challenge se definen como aplicaciones enfocadas a resolver los problemas fundamentales de la ciencia y la ingeniería (Barroso *et al.*, 2003), ya que (Silberschatz *et*





al., 2006) afirman que hay cuatro razones principales para construir sistemas distribuidos: compartición de recursos, aceleración de cálculos, fiabilidad y comunicación. La aceleración de cálculos evidentemente se puede lograr utilizando un clúster de alto rendimiento conocido como HPC (High Performance Clúster) que permite utilizar estos recursos computacionales como: memorias, microprocesadores. Ya que “Conforme la tecnología avanza los recursos de los ordenadores se incrementan también, tales como la cantidad de núcleos de procesamiento, memoria RAM y capacidad de almacenamiento, muchos de estos recursos no son aprovechados en su totalidad, a menudo, el recurso ocioso en los computadores, en promedio, es del 85% durante el día y del 95% durante la noche”, además, se presenta un uso de procesador menor al 5%”. (Gorlatch, 2008). Según lo antes mencionado estos recursos computacionales pueden ser aprovechados de mejor manera implementando un clúster de computadoras de alto rendimiento dedicado a la aceleración de cálculos, además de ello es importante conocer esta aceleración que trae consigo la reducción en los tiempos de renderización, esta cualidad se debe a que todos los cálculos necesarios para la interpretación del modelo son distribuidos entre todo el conglomerado de computadores, en palabras más sencillas se divide el trabajo para lograr la aceleración de cálculos entre todos los nodos conformantes del clúster, según lo antes abordado la cuestión es la siguiente: ¿en qué medida disminuye el tiempo de renderización de escenas tridimensionales fotorealistas con clústeres de computadoras de alto rendimiento?, para responder a esta interrogante se encontraron los siguientes antecedentes: (Fernández, 2006) presenta su investigación “Sistema GRID Para el Render de Escenas 3D distribuido: YAFRID” en el cual se crea un sistema vía web que permite la renderización de escenas tridimensionales utilizando múltiples clúster de computadoras reduciendo así drásticamente los tiempos de renderizado y creando a la vez un servicio en línea que permite el uso del motor de render yafray a través de una interface web, en este proyecto solo se limita a la validación del sistema GRID mas no al estudio de los tiempos de renderizado con el sistema, asimismo (Bedoya *et al.*, 2007) a través del Departamento de Sistemas y el departamento de Física y Astrofísica Computacional así como el Instituto de Física realizan la investigación “ZeBrA-Core Una Herramienta Para la Generación de Imágenes Fotorealistas Usando POV-Ray en ambientes de Computación Distribuida” en el cual estudia los tiempos de renderización de escenas tridimensionales basados en trazado de rayos, dichas mediciones se realizan sobre una sola imagen con un clúster con capacidad de 22 procesadores y con la división de la imagen en 1, 2, 5, 10, 15 y 20 franjas, la generación en secuencial de esta la imagen estudiada normalmente toma 10 horas 20 minutos, usando 20 procesos y el rendering utilizando la herramienta toma solamente 37 minutos con 20 procesos, finalmente (Olivares *et al.*, 2013) presentan un proyecto llamado “diseño de clústeres computarizados con algoritmos de renderizado para la construcción de escenas tridimensionales a través del uso de herramientas open source” dicha investigación utiliza el sistema clusterknoppix para desarrollar un algoritmo y hacer uso del clúster para renderizar escenas tridimensionales fotorealistas logrando una disminución del tiempo de renderizado hasta en un 359%.

Muchas universidades y centros de investigación hoy en día construyen su propio clúster de computadoras para dar solución a problemas de diversa índole así lo demuestra el ranking de las 500 mejores supercomputadoras del mundo, encabezando esta lista se halla el Tianhe-2, un superordenador desarrollado por la Universidad Nacional de Tecnología de Defensa de China, es el sistema N° 1 del mundo, con un rendimiento del 33,86 petaflop/s. La lista se anunció el 17 de junio durante la sesión inaugural de la Conferencia Internacional de Supercomputación 2013 en Leipzig, Alemania (Top 500, 2013).

En este sentido el objetivo de la investigación es el de evaluar la reducción en los tiempos de renderización de escenas tridimensionales fotorealistas con clústeres de computadoras de alto rendimiento.





la presente investigación es importante ya que permitirá aprovechar los recursos computacionales tales como uso de memorias y procesadores que son desperdician comúnmente en organizaciones, empresas y universidades en las cuales los trabajos realizados no requieren generalmente grandes capacidades de procesamiento ya que el mayor uso que se les da es para trabajo de oficina, en tal sentido esta puede ser una solución interesante para empresas, investigadores o alumnos cuyo trabajo esté orientado a la síntesis de imágenes fotorealistas utilizando la técnica de trazado de rayos, aprovechando de esta manera recursos computacionales remanentes de las organizaciones.

MATERIALES Y MÉTODOS

El diseño de investigación es de tipo experimental puro de subtipo postprueba únicamente y grupo control, se varió la cantidad de núcleos y así observar la reducción en los tiempos de renderización, como se observa en la tabla 1, son 6 los grupos de estudio. El grupo 6 es el grupo control con una sola computadora, recordemos que un clúster es la conexión de dos a más computadoras, los grupos restantes fueron los grupos experimentales en las que se varió la cantidad de núcleos de procesamiento.

Tabla 1. Cantidad de computadoras (nodos) por grupo de estudio

Grupo	Cantidad de computadoras	Cantidad de núcleos
Grupo 1	1	8
Grupo 2	2	16
Grupo 3	4	32
Grupo 4	6	48
Grupo 5	8	64
Grupo 6	10	80

La población es de carácter infinito ya que modelo tridimensional puede contar con infinitas posiciones de cámara, estas posiciones de las cámaras son conocidas como escenas, en ese sentido la muestra está conformada por 384 ocurrencias según muestreo probabilístico para poblaciones infinitas.

$$n = \frac{Z_{\alpha}^2 \cdot P \cdot (1 - P)}{i^2} \quad d \quad : \quad Z = 1.96 \quad i = 0.05 \quad P = 0.5$$

$$n = \frac{(1.96)^2 (0.5)(1 - 0.5)}{0.05^2} = 384.16$$

Para extraer estas escenas se diseñó el modelo de la plaza de armas de la ciudad de Talavera, este modelo tridimensional se muestra en la figura 1.





Figura 1. Modelo tridimensional de la plaza de armas de la ciudad de Talavera

A partir del modelo tridimensional (Figura 1), se generó posiciones aleatorias de las cámaras en Microsoft Excel. Seguidamente se realizó la instalación y configuración del clúster HPC en 10 computadores de última generación con procesador Intel Core I7, 8 GB de memoria RAM y 8 núcleos de procesamiento. Se renderizó cada una de las escenas de los grupos de estudio con el motor de renderizado de VRAY basado en trazado de rayos, se pudo observar que el proceso de renderización pasa por tres etapas, inicialmente en la primera etapa se mapea la escena a fin de trazar la estructura básica de los elementos conformantes de la escena así como de los efectos de la misma, en la segunda etapa se realiza el primer barrido de la escena, este barrido comienza a describir los pixeles que tendrá la imagen resultante. Finalmente, en la tercera etapa se realiza la segunda barrida de la escena cada núcleo es encargado de procesar los datos de la descripción de la geometría y simular la luz, así como los efectos que trae consigo la iluminación global. Durante la realización de estas fases se observó que la utilización de los núcleos de procesamiento llegó al 100%, asimismo a cada núcleo de procesamiento se le asignó un cuadro, Este fue el proceso que se realizó para las 384 escenas respectivamente de los grupos control y experimental y se tomaron los tiempos de renderización respectivos. Para determinar la disminución de los tiempos de renderización de escenas tridimensionales fotorealistas con el trazado de rayos se utilizó el método de ANOVA de un factor para determinar si hay diferencias significativas entre los diversos grupos de comparación, según (Bakieva y González, 2010) antes de realizar el ANOVA de un factor se debe realizar la comprobación de los supuestos que debe cumplir este método, es decir que los datos se distribuyan normalmente y las varianzas sean homogéneas, seguidamente se debe verificar si el ANOVA es significativo para la comparación inter-grupos, finalmente se debe realizar una de las pruebas POST-HOC, para nuestro caso se escogió la prueba de T3 de Dunnett para ver las diferencias significativas, en las siguientes tablas se puede apreciar los resultados obtenidos.

RESULTADOS Y DISCUSIÓN

Pruebas de normalidad

En la tabla 2 se puede observar los resultados de la prueba de normalidad realizados con el software de SPSS.



Tabla 2. Resultados de la prueba de normalidad

clúster	Kolmogorov-Smirnov ^a			
	Estadístico	gl	Sig.	
tiempo	Clúster con 8 CPUs	,071	64	,200*
	Clúster con 16 CPUs	,083	64	,200*
	Clúster con 32 CPUs	,070	64	,200*
	Clúster con 48 CPUs	,089	64	,200*
	Clúster con 64 CPUs	,095	64	,200*
	Clúster con 80 CPUs	,090	64	,200*

*. Este es un límite inferior de la significación verdadera.

a. Corrección de la significación de Lilliefors

Se puede visualizar que el valor de Sig. Kolmogorov-Smirnov es mayor a 0.05, por lo tanto, aceptamos que los datos provienen de una distribución normal (Tabla 2).

En la tabla 3 se puede observar los resultados de ANOVA de un factor

Tabla 3. -Resultados de ANOVA

	Suma de cuadrados	gl	Media cuadrática	F	Sig.
Inter-grupos	1,419,554,250	5	283,910,850	718,043	,000
Intra-grupos	149,459,375	378	395,395		
Total	1,569,013,625	383			

De similar manera se puede observar que el p-valor de comparación de inter-grupos es menor a 0.05, entonces se comprueba que los tres grupos de estudio son distintos (Tabla 3).

Igualdad de varianzas se puede evidenciar visualizando la prueba de homogeneidad de varianzas, utilizando el estadístico de Levene (Tabla 4).

Tabla 4. -Resultados de la igualdad de varianzas

Estadístico de Levene	gl1	gl2	Sig.
27,684	5	378	,000

Como se puede observar el valor de Sig. es menor a 0.05, por lo tanto, las varianzas no son iguales (Tabla 4).

Pruebas Post-Hoc T3 de Dunnett

Finalmente, para poder analizar las diferencias existentes entre los diversos grupos de comparación se utilizó la prueba T3 de Dunnett, según ello se calculó los porcentajes de mejora entre los diversos grupos de comparación (Tabla 5).





Tabla 5. Resultados de los porcentajes de mejora

Clúster	Clúster	Porcentajes de mejora
Clúster con 8 CPUs	Clúster con 16 CPUs	-34.94%
	Clúster con 32 CPUs	-52.36%
	Clúster con 48 CPUs	-56.61%
	Clúster con 64 CPUs	-55.05%
	Clúster con 80 CPUs	-49.41%
Clúster con 16 CPUs	Clúster con 8 CPUs	53.71%
	Clúster con 32 CPUs	-26.78%
	Clúster con 48 CPUs	-33.31%
	Clúster con 64 CPUs	-30.91%
	Clúster con 80 CPUs	-22.24%
Clúster con 32 CPUs	Clúster con 8 CPUs	109.94%
	Clúster con 16 CPUs	36.58%
	Clúster con 48 CPUs	-8.91%
	Clúster con 64 CPUs	-5.64%
	Clúster con 80 CPUs	6.20%
Clúster con 48 CPUs	Clúster con 8 CPUs	130.50%
	Clúster con 16 CPUs	49.95%
	Clúster con 32 CPUs	9.79%
	Clúster con 64 CPUs	3.60%
	Clúster con 80 CPUs	16.60%
Clúster con 64 CPUs	Clúster con 8 CPUs	122.49%
	Clúster con 16 CPUs	44.74%
	Clúster con 32 CPUs	5.97%
	Clúster con 48 CPUs	-3.47
	Clúster con 80 CPUs	12.55%
Clúster con 80 CPUs	Clúster con 8 CPUs	97.68%
	Clúster con 16 CPUs	28.60%
	Clúster con 32 CPUs	-5.83%
	Clúster con 48 CPUs	-14.23%
	Clúster con 64 CPUs	-11.15%

Los porcentajes de los tiempos de renderización en algunos casos aumentan y en otros disminuyen (Tabla 5), cuando los porcentajes tienen signos positivos se halla un incremento en los tiempos de renderización, y en caso de que muestren signo negativo es señal de que se ha reducido en dicho porcentaje, como antes se mencionó el objetivo principal de la investigación es el de determinar la disminución del tiempo de renderización de escenas tridimensionales fotorealistas con el clúster de alto rendimiento, para ello se tuvo que determinar inicialmente el tiempo de renderización de escenas tridimensionales fotorealistas sin intervención del clúster de alto rendimiento en el grupo control, seguidamente se procedió a la implementación del clúster de alto rendimiento para así determinar el tiempo de renderización de escenas tridimensionales fotorealistas en los grupos experimentales, finalmente se realizó una comparativa de los tiempos de renderización de escenas tridimensionales fotorealistas entre los grupos experimentales y el grupo control e identificar el menor tiempo. La cantidad de nodos asignados inicialmente a la renderización no fue lo suficiente para reducir



notablemente los tiempos de renderización, pero a medida que incrementó la cantidad de CPUs al ir agregando nodos al clúster se pudo observar que el porcentaje de reducción llega hasta un 56.61 %, esto es la reducción máxima que podemos observar según el grupo control conformado por 8 CPUs vs. La configuración del clúster con 48 CPUs, sin embargo cabe aclarar que inicialmente al ir agregando 1 nodo más al clúster la mejora es notable llegando al 34.94 %, seguidamente se agrega 2 nodos más al clúster y los tiempos de renderización se reducen muy notablemente llegando a un 52.36 % llegando a un valor óptimo con 6 nodos con una reducción del 56.61 %, a partir de este punto al ir incrementando 2 nodos la mejora en los tiempos de renderización obviamente se reduce a un 55.05% y si seguimos incrementando la cantidad de nodos en dos llegamos a una reducción del 49.41% (Tabla 5) esto quiere decir que al ir incrementando la cantidad de nodos sobre la configuración óptima se irá reduciendo el porcentaje de mejora llegando al punto en el que los tiempos de renderización se incrementarán, esto es debido a que el clúster toma más tiempo en la planificación y la asignación de tareas así como la reelección de los resultados que en el proceso de renderizado en sí.

Como antes se mencionó y desarrolló en los antecedentes de la investigación se pudieron encontrar dos investigaciones que se asemejan a la presente investigación, primeramente según (Bedoya, 2007) en su investigación titulada como “ZeBrA-Core: Una Herramienta para la Generación de Imágenes Fotorrealistas Usando POV-Ray en Ambientes de Computación Distribuida” se renderiza de la escena denominada “Chess2” utilizando el software de Pov Ray que está basado en la técnica de iluminación de trazado de rayos, para ello realizan la división en franjas así como en cuadrículas de la escena “Chess2” (Figura 2).



Figura 2. – Escena Chess2

Esta división en franjas permitió distribuir los trabajos a través de procesos entre diversos procesadores, el caso más extremo de reducción en el tiempo de ejecución se obtuvo para la escena chess2 a una resolución de 8640x6480. Mientras que la generación en secuencial de esta imagen toma 10 horas 20 minutos, usando 20 procesos el rendering toma solamente 37 minutos. Por otro lado, si calculamos este tiempo de mejora en porcentajes estaríamos haciendo referencia aproximadamente a una reducción del 94.03 %, este porcentaje es bastante alto, con respecto a nuestros resultados obtenidos, ya que la escena chess2 es mucho más sencilla con respecto al modelo de la plaza de Talavera en lo que refiere a reflexiones, refracciones, vegetación y reflejos acuáticos, además es importante mencionar que solo se renderiza una posición en la escena chess2, mientras que la presente investigación toma 384 escenas tomadas desde puntos tridimensionales aleatorios mucho más estrictas para aproximarnos a valores más reales sobre la reducción de los tiempos de renderización en un ambiente real, también los resultados obtenidos de esta investigación anterior no rebelan un punto óptimo para la máxima reducción de tiempos de renderización por lo que se complementa perfectamente en contraposición.



Por otro lado (Olivares, 2013) presentan un proyecto llamado “Diseño de clústeres computarizados con algoritmos de renderizado para la construcción de escenas tridimensionales a través del uso de herramientas open source” dicha investigación utiliza el sistema clusterknoppix para desarrollar un algoritmo y hacer uso del clúster para renderizar escenas tridimensionales fotorealistas logrando una disminución del tiempo, al igual que en el caso anterior se renderizan escenas pequeñas cuya complejidad escénica es poca y se llega a reducir el tiempo de renderizado hasta en un 359%, con lo que se nota una reducción significativa, en la Figura 3 se puede apreciar la escena renderizada.

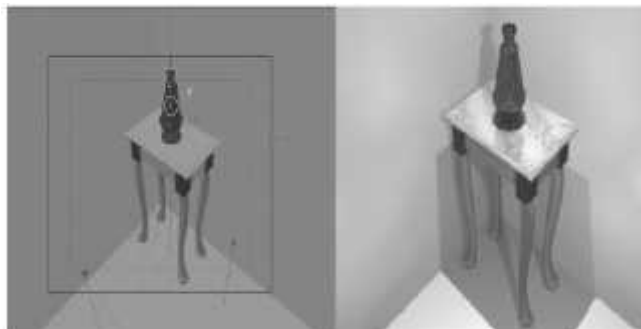


Figura 3. – Modelo Render mesa

Como se pudo apreciar en los antecedentes las escenas renderizadas suelen ser pequeñas y con pocos elementos con respecto a texturas y construcción geométrica, por ende, su complejidad en comparación con el modelo de la plaza de armas de Talavera suele ser mucho menor, objetivamente siendo las reducciones en los tiempos de renderización mayores, pero no tan generalizables con respecto a complejidad escénica, sin embargo, se nota claramente la reducción de tiempos beneficiosos al utilizar clústeres HPC.

CONCLUSIONES

Se evaluó los tiempos de renderización con la renderización de 384 escenas tridimensionales tomadas al azar del modelo de la plaza de armas de Talavera con la utilización del clúster de computadoras de alto rendimiento, asimismo se determinó los porcentajes de incidencia de mejora éntrelos diversos grupos experimentales de la investigación y se demostró que al igual que los antecedentes los tiempos de renderización se reducen favorablemente.

Los clúster de computadoras de alto rendimiento reducen los tiempos de renderización de modelos y escenas tridimensionales fotorealistas utilizando la técnica de trazado de rayos, la influencia de mejora se puede visualizar claramente cuando se utiliza un gran número de ordenadores, más por el contrario si se utiliza unos pocos ordenadores la incidencia de mejora es muy poca, pero a medida que se incrementa una buena cantidad de nodos al clúster se puede ver muy notablemente esta diferencia en los tiempos de renderización llegando a ser considerada como optima con una reducción del 56.61%, pero llega también el punto en el que no se puede reducir más el tiempo de renderización al ir incrementado muchos más nodos esto se debe a que la coordinación de los trabajos se vuelve más complicada y en contraposición se incrementan aún más estos tiempos.

A lo largo de todo este tiempo el hombre ha ido construyendo súper ordenadores que le ayuden a resolver problemas cuya complejidad computacional suele ser elevada, los clústeres de computadoras de alto rendimiento son prueba de ello, la solución en clúster aún se mantiene vigente y es utilizada en el mundo de la computación gráfica, así como en otros sectores para acelerar los cálculos.





LITERATURA CITADA

- Bakieva, y González. (2010). *Bakieva, M., González . J. y Jornet, J. SPSS: ANOVA de un Factor* (1 No. 1). Valencia.
- Barroso, Dean, y Holzle. (2003). Web Search for a Planet. *Web Search for a Planet, 1*, 22–28.
- Bedoya, Cadavid, y Zuluaga. (2007). *ZeBrA-Core: Una Herramienta para la Generación de Imágenes Fotorrealistas Usando POV-Ray y en Ambientes de Computación Distribuida*. Colombia. Retrieved from <http://www.saber.ula.ve/bitstream/123456789/16021/1/bedoya.pdf>
- Castillo, C., Olivera, A. (2006). *Implementación de un Cluster OpenMosix para Cómputo Científico*. Universidad Nacional Autónoma de México.
- Cohen, y Wallace. (1993). *Radiosity and Realistic Image Synthesis*. (Press, Ed.). United States.
- Deitel. (1987). *Introducción a los Sistemas Operativos*. Mexico: Addison-Wesley Iberoamericana.
- Dormido, Hernández, y Muñoz. (2003). *Procesamiento paralelo “Teoría y programación.”* (Torres, Ed.) (1ra Edición).
- Fernández. (2006). *Sistema GRID para el Render de Escenas 3D Distribuido: YAFRID*. Universidad de Castilla. Retrieved from http://www.esi.uclm.es/www/cglez/downloads/students/pfc/2006_jafernandez.pdf
- Gorlatch, Fragopoulou, y Priol. (2008). *Grid Computing: Archivevents and Prospects*. (Springer, Ed.), *Grid Computing: Archivevents and Prospects*. Canadá.
- Hai, Rajkumar, y Mark. (2002). *Cluster Computing Tools, Applications, and Australian Initiatives for Low Cost Supercomputing*. California.
- Matt, Wenzel, y Greg. (2016). *Physically Based Rendering From Theory to implementation*. (Pierce, Ed.) (3rd Edición). New York: Todd Green.
- Mesa, y Branch. (2008, December). Implementación de un cluster homogéneo para la resolución de problemas de alta complejidad computacional. *Avances En Sistemas E Informática*, 41–48.
- Olivares, Ferreira, Vega, Cendejas, y Rosano. (2013, November). Diseño de clústeres computarizados con algoritmos de renderizado para la construcción de modelos tridimensionales a través del uso de herramientas open source. *Multiciencias*, 88–96.
- Raya, G. (2009). *Técnicas de Iluminación Para Gráficos por Ordenador*. Mexico.
- Silberschatz, Galvin, y Gagne. (2006). *Fundamentos de sistemas operativos*. (Sánchez, Ed.) (Septima Ed). España: Mc Graw Hill.
- Sun. (2006). Sun Cluster Architecture. Retrieved January 22, 2006, from <http://www.sun.com./software/grid/SunCLusterArchitecture.pdf>
- Tanembaum. (1996). *Sistemas Operativos Distribuidos*. (Hall, Ed.) (1ra Edición). Mexico.
- Top 500. (2013). Top 500 Supercomputers. Retrieved December 10, 2013, from <https://www.top500.org/>
- Torrealba, M. (2002). Construcción de un Cluster Mosix con Pruebas con Simulación de Halo. Oxaca.
- Whitted. (1980). An Improved Illumination Model For Shaded Display. *Communications of the ACM* 23, 1, 343–349.

