

## PENGARUH BANYAKNYA *CHUNKS* DAN *PRIORITY RENDER ANIMATION 3D* TERHADAP WAKTU *RENDER* MENGGUNAKAN *ANOVA ONE WAY* DENGAN MENERAPKAN *PARALLEL COMPUTING RENDER ENGINE*

**Much. Rifqi Maulana, Ichwan Kurniawan**

*Program Studi Teknik Informatika, STMIK Widya Pratama*

*Jl. Patriot 25 Pekalongan*

*Telp (0285)427816*

*email : [ichwan.ana10@gmail.com](mailto:ichwan.ana10@gmail.com), [rifqi\\_13@yahoo.com](mailto:rifqi_13@yahoo.com)*

### ABSTRAK

Dalam proses *rendering* perangkat *hardware* yang paling penting adalah *CPU (Central Processing Unit)*, *VGA (Video Graphic Adaptor)* dan *RAM (Random Access Memory)*. Pemilihan *render engine* dilakukan pada *software* yang digunakan, dan mengatur parameter *render* dapat mempengaruhi hasil akhir dari *rendering model 3D*. Untuk memperoleh hasil *render* yang maksimal dengan waktu *render* yang singkat dapat memanfaatkan beberapa sumberdaya komputer secara bersama-sama dalam sebuah jaringan. Dalam penelitian ini *render engine* yang digunakan adalah *blender network render*, dengan menggunakan 12 unit perangkat komputer, yaitu 10 unit *slave mode*, 1 unit *master* dan 1 unit *client*. Uji coba *rendering* dilakukan 5 kali penambahan *frame render*, yaitu 12, 24, 38, 60, 120 *frame render*. Kemudian nilai pada setiap uji coba menghasilkan nilai penggunaan sumberdaya (*slave/node*) dalam tiap unit, nilai inilah yang nantinya akan diuji lagi menggunakan teknik uji komparasi statistik *Anova one way*. Hasil dari uji komparasi statistik *Anova one way* adalah  $18.11 > 2.53$  untuk probabilitas 5% dan  $3.71 > 3.35$  maka  $H_a$  diterima dan  $H_0$  ditolak. Jadi Terdapat perbedaan penggunaan Waktu/Second *RENDER* dengan adanya penambahan *Chunks* dan *Priority Render* (penambahan *Chunks* dan *Priority Render* dapat mengurangi penggunaan Waktu/Second *Render*) Dengan Memanfaatkan *Parallel Computingrender Engine* Pada Platform *Open-Source Software*. Kemudian untuk menghitung korelasi antar sampel terdapat 3 pasang sampel  $X_1:X_2$ ,  $X_1:X_3$  dan  $X_2:X_3$ . Dari 3 korelasi tersebut terdapat 1 korelasi yang Hipotesis Alternatif diterima, yaitu korelasi  $X_1:X_3$  dengan harga  $T$  hitung (2.98) lebih kecil dari harga  $T$  tabel ( $2.23 < 2.10$ ). Dengan demikian  $H_a$  diterima, dan  $H_0$  ditolak. Kesimpulannya Terdapat Perbedaan Waktu/Second *RENDER* *Render* Sebelum Menambah dari *Chunks 5* dan *Priority 1 Render* Menjadi *Chunks 10* dan *Priority 10*. Hal ini dikarenakan sampel korelasi yang diuji terdapat selisih *Chunks* dan *Priority render* yang cukup besar. Sedangkan korelasi yang Hipotesis Alternatifnya ditolak yaitu  $X_1$  dengan  $X_2$  dan  $X_2$  dengan  $X_3$  karena selisih *Chunks* dan *Priority render* yang diuji hanya sedikit. Dari hasil pengujian yang telah disimpulkan bahwa, dengan menambah *frame*, *chunks* atau *priority render* dapat mempengaruhi banyaknya sumber daya *node/slave* yang digunakan pada saat *render*, dan mempengaruhi waktu/*second rendering*.

**Keywords:** *Parallel Computing, Render Engine, 3D Animation, open-source software*

## 1 PENDAHULUAN

### 1.1 Latar Belakang

Pemanfaatan *render fram* secara tradisional memerlukan beberapa kumpulan komputer untuk mengerjakan proses *render*, yang hasilnya dapat memuaskan dalam hal daya komputasi dan waktu. Namun dalam beberapa layanan *render fram* menanggung biaya dalam proses *rendering*. [2]. Sebagai media *render* komputer dapat dimanfaatkan secara bersamaan dan dalam waktu yang sama, hal ini dapat memungkinkan

proses *render* dapat dilakukan dengan waktu yang lebih cepat dibandingkan dengan hanya memanfaatkan satu sumber daya komputer.

Penggunaan internet sebagai media *rendering* yang terdistribusi menjadi kebutuhan bagi produsen gambar dan animasi, hal ini dapat mempercepat proses *rendering* dan waktu *rendering* yang singkat tanpa mempengaruhi kualitas hasil *render* yang diperoleh [3]. Integrasi antara beberapa perangkat komputer dalam jaringan internet dapat dimanfaatkan

sebagai media *rendering*, hal ini dapat mempersingkat proses *rendering* tanpa mengurangi kualitas hasil *render*.

Konsep jaringan distribusi *Local Area Network (LAN)*, *end user* dapat menggunakan *network* sepanjang media *network* tidak digunakan komputer lain. Untuk mengatasi permasalahan *collision*, *Ethernet* telah dilengkapi dengan *collision detection*, jika terjadi *collision* komputer pengirim akan diberikesempatan untuk mengirim ulang data [4]. Dengan menggunakan Jaringan *Local Ethernet* sumberdaya komputer dapat digunakan secara bersama-sama dalam satu kegiatan. Artinya sumberdaya komputer, seperti *processor*, *video grafick adapter (VGA)* dan *random access memory (RAM)* dapat digunakan bersama-sama dalam satu waktu.

Dalam Penelitian sebelumnya [5], dengan memanfaatkan *Parallel computing render engine* untuk *rendering 3D model* pada *open-source software*. Dalam penelitian tersebut *render engine* yang digunakan adalah *blender network render*, dengan menggunakan 12 unit perangkat komputer, yaitu *10 unit slave mode*, *1 unit master* dan *1 unit client*. Uji coba *rendering* dilakukan 5 kali penambahan *frame render*, yaitu 12, 24, 38, 60, 120 *frame render*. Pada penelitian ini pengukuran komparasi menggunakan teknik uji komparasi statistik *Anova one way*.

Hasil dari uji komparis statistic *Anova one way* adalah  $18.11 > 2.53$  untuk probability 5% dan  $18.11 > 3.77$  untuk probality 1%.  $F$  hitung  $> F$  tabel maka  $H_a$  diterima dan  $H_o$  ditolak yaitu Terdapat Perbedaan Banyaknya Sumberdaya Komputer (Slave/Node) Yang Digunakan Sebelum Penambahan Frame Render Dari 12, 24, 36, 60, 120 Frame Dengan Memanfaatkan Parallel Computingrender Engine Pada Platform Open-Source Software. Terdapat 2 korelas yang Hipotesisi Alternatif diterima, yaitu korelas  $X_3: X_4$  dengan harga  $T$  hitung (2.98) lebih kecil dari hatga  $T$  tabel ( $2.979 < 2.101$ ) [5].

Dengan demikian  $H_a$  diterima, dan  $H_o$  ditolak yaitu Terdapat Perbedaan Penggunaan Slave/Node Render Sebelum Menambah 36 Frame Menjadi 60 Frame, dan korelas  $X_4: X_5$  dengan harga  $T$  hitung (4.81) lebih kecil dari hatga  $T$  tabel ( $4.81 < 2.101$ ). Dengan demikian

$H_a$  diterima, dan  $H_o$  ditolak yaitu Terdapat Perbedaan Penggunaan Slave/Node Render Sebelum Menambah 60 Frame Menjadi 120 Frame [5]. Dalam pengujian tersebut dapat dilihat bahwa dengan menggunakan *Parallel computing render engine* untuk *rendering 3D model* pada *open-source software* bahwa terdapat perbedaan penggunaan *slave/node render* sebelum menambahkan 60 frame menjadi 120 frame pada *render 3D Model*. Namun pada penelitian sebelumnya belum diuji apakah ada perbedaan penggunaan *slave/node* jika objek *render*-nya adalah *3D Animation*.

Dari pemaparan di atas, dalam penelitian ini akan menerapkan *parallel computing* untuk *rendering 3d animation* pada *open-space software* dengan menguji apakah ada perbedaan penambahan *Chunks* dan *Priority* yang digunakan pada proses *rendering* animasi. Objek animasi 3D yang digunakan sebagai bahan ujicoba memanfaatkan 520 frame render, di dalamnya menggunakan 9 gerakan yang disusun secara berurutan. Percobaan dilakukan dengan beberapa tahap dengan memanfaatkan 10 *slave/node*, dengan spesifikasi komputer yang sama.

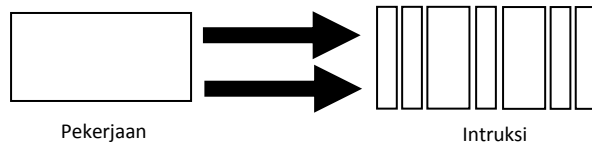
Dalam penelitian ini, teknik statistik yang digunakan unutk menguji hipotesis menggunakan teknik *Analisis of Variants (Anova) one way*. *Analisis of Variants (Anova) one way* merupakan teknik uji statistik yang digunakan untuk membandingkan dua atau lebih variable penelitian [6]. Menurut G. Keller (2014) *Anova* umumnya dapat digunakan sebagai pengujian signifikansi dari pengamatan *experimental* atau perbedaannya [7]. Dengan menggunakan teknik statistik hipotesis *Anova*, banyaknya *Chunks* dan *Priority render* akan dibandingkan satu sama lain apakah akan mempengaruhi waktu/*second render* yang digunakan.

Manfaat dari penelitian ini adalah ingin mengetahui penerapan *Parallel Computing* untuk *rendering 3D Animation* pada *Open-Source Software* dengan menambah *Chunks* dan *Priority* dapat mempengaruhi waktu *rendering* dengan penggunaan *slave/node render* yang sama.

## 1.2 Landasan Teori

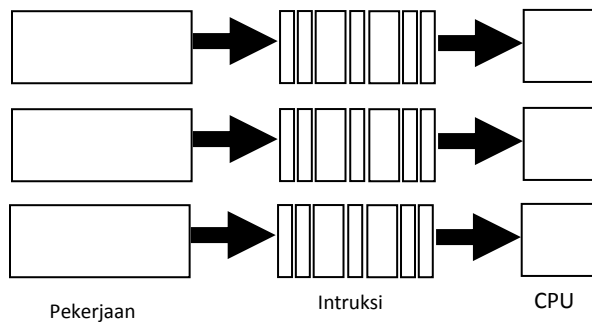
### 1.2.1 Parallel Computing

*Parallel computing* adalah penggunaan lebih dari satu sumber daya komputasi secara simultan untuk memecahkan persoalan komputasi [6]. Proses komputasi umumnya dikerjakan secara serial yang dikerjakan oleh prosesor tunggal, artinya intruksi yang diberikan kepada prosesor berurutan dan dalam saat yang sama intruksi tidak boleh dikerjakan sebelum intruk yang masih dikerjakan selesai.



Gambar 1.1 Serial Computing

Dengan memanfaatkan *parallel Computing* proses komputasi akan dipecah menjadi beberapa bagian yang dapat dikerjakan secara bersama-sama dalam satu waktu [6]. Artinya proses komputasi tidak dikerjakan oleh satu prosesor saja, namun proses komputasi akan dibagi menjadi beberapa bagian komputasi yang akan dikerjakan oleh beberapa prosesor secara bersama-sama dalam waktu yang sama.



Gambar 1.2 Parallel Computing

Alasan utama penggunaan *parallel computing* adalah [6]:

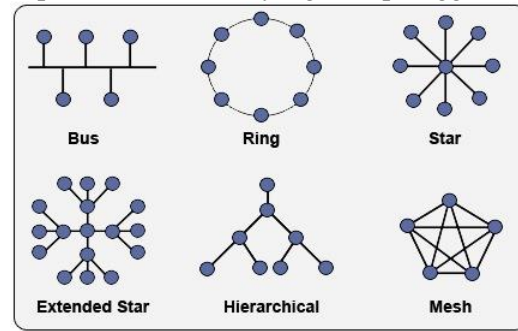
1. Dapat menghemat waktu
2. Dapat memecah persoalan yang lebih besar
3. Dapat menghemat biaya
4. Dapat mengatasi keterbatasan fisik dari komputasi serial

Alasan lain yang cukup penting adalah *fault tolerant*, jika salah satu prosesor mengalami

kegagalan prosesor lain dapat menggantikannya, meskipun dengan perform yang menurun.

### 1.2.2 Local Area Network (LAN)

LAN merupakan koneksi jaringan yang berukuran relative kecil, oleh karena itu dapat dikembangkan dengan mudah dan mendukung kecepatan transfer data yang cukup tinggi.



Gambar 1.3 Topology LAN

Ada 4 topology LAN yaitu [4]:

- a. Topology *Bus*, menggunakan kabel *Backbone* dan semua host terhubung secara langsung pada kabel tersebut.
- b. Topology *Star*, menghubungkan semua komputer pada sentral atau konsentrator. Biasanya konsentrator adalah sebuah *Hub* atau *Switch*.
- c. Topology *Ring*, menghubungkan host dengan host lainnya hingga membentuk ring (lingkaran tertutup).
- d. Topology *Mesh* atau *Fully-Mesh*, menghubungkan setiap komputer secara point-to-point. Artinya semua komputer akan saling terhubung satu-satu sehingga tidak dijumpai ada *link* yang putus.
- e. Topology *ExtendedStar*, merupakan topology star yang telah dikembangkan, yaitu menggabungkan beberapa topology star menjadi satu kesatuan. Alat yang digunakan untuk menghubungkan masing-masing topology star adalah *hub* dan *switch*.
- f. Topology *Hierarchical*, hampir sama dengan topology *extended star*. Perbedaannya terletak pada alat penghubung masing-masing topology start, tidak menggunakan *hub* dan *switch* namun menggunakan komputer sebagai kendali *traffic* pada topologi lain.

### 1.2.3 *Rendering*

Rendering adalah suatu proses untuk mengubah model geometri menjadi suatu gambar. Proses untuk membangun sebuah gambar membutuhkan beberapa fase seperti *modelling*, pengaturan *material* dan *texture*, penempatan virtual *light*, dan proses render [7]. Dalam proses *rendering* ini penggunaan *hardware* yang mencukupi akan berpengaruh pada waktu *rendering* model 3D

Rendering memainkan peran yang sangat vital pada proses penciptaan animasi dan gambar. Rendering dapat digunakan untuk meniru objek visual yang nyata (*photorealistic rendering*), atau *stylistic fashion (non-photorealistic renderings)* dengan baik. Dengan mengesampingkan *style*, *rendering* dapat diselesaikan dengan menggunakan algoritma yang memang dikhususkan untuk proses render. Algoritma ini bisa menjadi sangat kompleks. Semakin kompleks algoritma yang digunakan maka semakin lama waktu yang dibutuhkan untuk melakukan proses kalkulasi. Dengan menggunakan *hardware* yang moderen proses render dari film sederhana dapat menghabiskan waktu antara beberapa menit sampai dengan dua ratus menit untuk setiap frame-nya [6]. Dapat disimpulkan bahwa waktu yang dibutuhkan dalam proses render animasi dapat dilihat dari banyak frame yang dimiliki dalam sebuah adegan. Dapat gambarkan jika dalam satu frame perkiraan waktu render adalah dua menit maka untuk mengetahui waktu keseluruhan render, dapat dikalikan dengan banyak frame dalam satu adegan animasi.

Pada umumnya proses render yang dilakukan pada saat ini masih bekerja secara single dengan menggunakan sebuah mesin yang memiliki sumber daya yang besar. Kerugian dari proses render yang bekerja secara single adalah Waktu yang diperlukan untuk melakukan proses ini masih dirasakan cukup lama walaupun mesin yang digunakan memiliki sumber daya yang cukup besar [7]. Hal ini dikarenakan satu sumber daya komputer menangani semua proses render, hal ini mengakibatkan proses render membutuhkan waktu yang lama. Jika proses render dikerjakan

menggunakan banyak sumber daya computer, proses render akan dibagi menjadi beberapa bagian dan akan didistribusikan kesemua sumber daya yang ada. Dengan cara seperti itu proses render akan menjadi lebih singkat dan lebih cepat, karena semua sumber daya computer bekerja bersama dalam satu waktu.

### 1.2.4 *Blender Network Render*

Blender adalah aplikasi *Opens Source pembentuk* grafis 3D, aplikasi ini mendukung 3D *Pipeline-modeling, rigging, animation, simulation, rendering, compositing, motion tracking, even video editing* dan pembuatan *game*. Pengguna tingkat lanjut dapat menggunakan *API Blender* dengan menggunakan *script python* [8]. Dalam aplikasi *blender* karena bersifat *open source* dapat meminimalkan beban biaya dan kemudian dengan aplikasi *blender* fitur-fitur yang dimiliki lebih variatif.

*Network Render* tujuannya bukan untuk menggantikan *render farm*, melainkan dirancang untuk secara interaktif merender sebuah gambar melalui jaringan komputer dengan umpan balik langsung di *Blender*. Dalam *network render* memerlukan desain yang berbeda dengan pembuatan *render farm*, meskipun dalam kasus sederhana dapat menjadi pengganti [9]. Cara *network render* hampir sama dengan *render farm*, namun dengan menggunakan *network render* pekerjaan render secara paralel dapat dilakukan dalam aplikasi *blender* sendiri.

### 1.2.5 *Analisis of Variants (Anova) one way*

Analisis varians dapat digunakan untuk menguji hipotesis komparasi rata-rata k sample bila data yang diolah berbentuk interval atau rasio. Satu sampel dalam k kejadian atau pengukuran berarti sampel tersebut berpasangan, model *befor-after*. Satu sampel diberi perlakuan sampai 5 kali, hal ini artinya 5 sampel tersebut berpasangan. Sedangkan k sampel dalam satu kejadian berarti sampel *independen*. Terdapat dua jenis *Anova*, *Anova single classification* dan *Anova multiple classification* [10].

*Analisis single classification* yang sering disebut *Anova one way* dapat digunakan untuk menguji hipotesis komparasi rata-rata k sampel, bila pada setiap sampel hanya terdiri atas satu kategori. Dalam teknik ini setiap sampel akan mempunyai

Mean (rata-rata) dan Varians (simpangan baku kuadrat), dimana  $n$  = jumlah sample,  $M$  = mean/rata-rata sedangkan  $s^2$  = varians [10].

Dalam *Anova* terdapat [10]:

1. Deviasi Total, yaitu jarak antara nilai individual yang ada dalam seluruh sampel dengan Mean Total. Dalam hal ini misal ( $X - M_{tot}$ ).
2. Deviasi antara kelompok (*between*), yaitu jarak antara Mean setiap kelompok dengan Mean Total. Dalam hal ini misal ( $M_4 - M_{tot}$ ).
3. Deviasi dalam kelompok (*within*), yaitu jarak nilai seluruh individu dalam satu kelompok dengan Mean kelompok itu. Dalam hal ini misal ( $X - M_4$ ).

Kemudian jarak suatu nilai dalam kelompok terhadap rata-rata ( $X_i - M$ ) dikuadratkan menjadi  $(X_i - M)^2$ . Kuadrat ini selanjutnya disingkat dengan JK dan merupakan varian dari kelompok tersebut. Karena dalam pengujian hipotesis melibatkan lebih dari dua kelompok sampel, maka akan terdapat beberapa macam JK [10], yaitu:

1. Jumlah Kuadrat Total ( $JK_{tot}$ ) merupakan penjumlahan kuadrat deviasi nilai individual dengan  $M_{tot}$

$$JK_{tot} = \sum X_{tot}^2 - \frac{(\sum X_{tot})^2}{N}$$

Rumus 1.1 Jumlah Kuadrat Total

$N$  = jumlah seluruh anggota sampel

2. Jumlah Kuadrat Antara ( $JK_{ant}$ ) merupakan jumlah selisih kuadrat Mean Total ( $M_{tot}$ ) dengan Mean setiap kelompok ( $M_1$ ), dikalikan dengan jumlah sampel setiap kelompok.

$$JK_{ant} = \sum \frac{(\sum X_k)^2}{n_k} - \frac{(\sum X_{tot})^2}{N}$$

Rumus 1.2 Jumlah Kuadrat Antara

3. Jumlah Kuadrat dalam Kelompok ( $JK_{dal}$ )

$$JK_{dal} = JK_{tot} - JK_{ant}$$

Rumus 1.3 Jumlah Kuadrat dalam Kelompok

Setiap sumber variasi didamping dengan  $dk$ , dan  $dk$  untuk setiap sumber variasi tidak sama.

Untuk Antar Kelompok  $dk = m - 1$

Untuk Dalam Kelompok  $dk = N - m$

Total  $dk = N - 1$

$M$  = Jumlah kelompok sampel

$N$  = Jumlah seluruh anggota sampel

Untuk dapat menghitung harga  $F$  hitung, maka beberapa sumber variasi harus dihitung mean kelompoknya, yang meliputi Mean Antara Kelompok dan Mean Dalam Kelompok.

Untuk Antar Kelompok:

$$MK_{ant} = \frac{JK_{ant}}{m - 1}$$

Rumus 1.4 Mean Antar Kelompok

Untuk Dalam Kelompok :

$$MK_{dal} = \frac{JK_{dal}}{N - m}$$

Rumus 1.5 Mean dalam Kelompok

$$F \text{ hitung} = \frac{MK_{ant}}{MK_{dal}}$$

Rumus 1.6  $F$  Hitung

Jadi untuk pengujian hipotesis dalam *Anova one way* diperlukan langkah-langkah sebagai berikut [10]:

1. Menghitung Jumlah Kuadrat Total ( $JK_{tot}$ ).
2. Menghitung Jumlah Kuadrat Antar Kelompok ( $JK_{ant}$ ).
3. Menghitung Jumlah Kuadrat dalam Kelompok ( $JK_{dal}$ ).
4. Menghitung Mean Kuadrat Antar Kelompok ( $MK_{ant}$ ).
5. Menghitung Mean Kuadrat Dalam Kelompok ( $MK_{dal}$ ).
6. Menghitung  $F$  Hitung ( $F_{hit}$ ).
7. Menbandingkan harga  $F$  hitung dengan  $F$  table dengan  $dk$  pembilang ( $m - 1$ ) dan  $dk$  penyebut ( $N - 1$ ). Harga  $F$  hasil perhitungan tersebut selanjutnya disebut  $F$  hitung ( $F_{hit}$ ), yang berdistribusi  $F$  dengan  $dk$  pembilang ( $m - 1$ ) dan  $dk$  penyebut ( $N - 1$ ) tertentu. Ketentuan pengujian sipotesis: *Bila  $F$  hitung lebih kecil atau sama dengan harga  $F$  table ( $F_h \leq F_t$ ) maka  $H_0$  diterima dan  $H_a$  ditolak, sebaliknya bila  $F_h > F_t$ , maka  $H_a$  diterima dan  $H_0$  ditolak.*
8. Membuat kesimpulan pengujian hipotesis:  $H_0$  diterima  $H_0$  ditolak.

Setelah mengetahui apakah hipotesis  $H_a$  diterima atau tidak, selanjutnya untuk membuktikan apakah perbedaan dua sampel

yang diterima atau tidak dengan menggunakan t-test.

$$t = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{S_1^2}{n_1} + \frac{S_2^2}{n_2} - 2.r\left(\frac{S_1}{\sqrt{n_1}}\right)\left(\frac{S_2}{\sqrt{n_2}}\right)}}$$

Rumus 1.7 *t-test related/berpasangan*  
 r = Korelasi antara dua sampel  
 $\bar{X}$  = Rata-rata sampel

**2 METODE PENELITIAN**

**2.1 Metode Penelitian**

- 1 Membuat Model 3D dan Animasi 3D dalam hal ini membuat gerakan Sholat, kemudian menambahkan dan mengatur modifier, lighting, camera, material dan texture, yang nantinya akan digunakan sebagai bahan penelitian.
- 2 Analisis dan perancangan konfigurasi topologi jaringan komputer dengan menggunakan *switch*, dalam hal ini terdapat 12 perangkat komputer dan 1 perangkat *switch*.
- 3 Mengatur konfigurasi perangkat komputer dalam *network render*, dalam hal ini terdapat 10 perangkat komputer yang berfungsi sebagai *slave mode*, 1 perangkat komputer yang berfungsi sebagai *master mode* dan 1 perangkat komputer yang berfungsi sebagai *client mode*.
- 4 Implementasi *rendering Animation 3D* dengan menggunakan *blender network render*, dalam hal ini experiment dilakukan dengan melakukan uji coba dengan *render, 1 Priority dan 5 Chunks render, 1 Priority dan 10 Chunks render, 10 Priority dan 10 Chunks render*.
- 5 Evaluasi dengan melakukan komparasi banyaknya *Chunks dan Priority render* apakah akan mempengaruhi waktu render yang digunakan. Dalam hal ini dalam tahap uji komparasi akan menggunakan teknik statistik *Anova one way* dengan menentukan hipotesis  $H_a$  dan  $H_o$  terlebih dahulu, kemudian akan mengukur tingkat korelasi antara antar sampel dengan sampel yang lain

untuk mengetahui perubahan penggunaan sumberdaya (*slave/node*) yang signifikan.

**2.2 Pengumpulan Data**

1. *Model 3D dan Animation 3D*




Dalam pembuatan model 3d terdapat 6 model 3d yaitu Monitor LCD, Meja, Keyboard, Mouse, Lampu Meja dan Mouse Path, detail dari properties model 3d sebagai berikut.




Table 2.1 *Properties Model 3D*

No	Model	Modifier	Material and Texture
1	Man	Mirror Armature	Diffuse: FFFFFFFF; intensity: 0.80 Specular: FFFFFFFF; intensity: 0.50 Texture: Uvmap; Image or Movie
2	Armature	Bone	
3	Sajadah		Diffuse: FFFFFFFF; intensity: 0.80 Specular: FFFFFFFF; intensity: 0.50 Texture: Uvmap; Image or Movie

Pada kasus ini, karena *engine render* menggunakan *network render* dengan *engine blender*, material dan texture memanfaatkan *property* untuk mengolahnya.

Table 2.2 *Properties Animation 3D*

No	Action	Key Frame	Frame	Hasil
1	Berdiri	1, 24	Start: 1 End: 24	
2	Takbir	1, 12, 24	Start: 1 End: 24	
3	Sedekap	1, 12, 24	Start: 1 End: 24	

No	Action	Key Frame	Frame	Hasil
4	Ruku'	1, 11, 23, 35, 48	Start: 1 End: 48	
5	Sujud	1, 6, 9, 11, 2, 27, 48	Start: 1 End: 48	
6	Itidal	1, 12, 24, 36, 48	Start: 1 End: 48	

Gambar 2.4 Animasi 3D yang belum dalam tahap rendering



2. Lighting

Kemudian setelah model 3d telah terbentuk, selanjutnya adalah mengatur pencahayaan pada lingkungan render. Ada 3 type lighting yang dimanfaatkan yaitu: *Sun Lighting*.

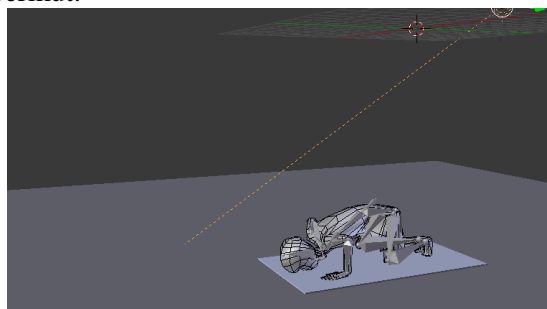
Table 2.3 Properties Lighting

No	Type	Properties
1	Sun	Lighting Color: FFFFFFFF; Energy: 3.000; Sky: Turbidity: 4.00 Horizon: Brightnes: 10.000 Sun: Brightnes: 1.000 Shadow: Sampling: 1; Soft Size: 0.100

Kemudian setelah mengatur posisi objek dan *lighting*, dan menentukan intensitas cahaya yang akan dihasilkan oleh light, mengatur tampilan *shadow* atau bayangan model 3d yang dihasilkan oleh *lighting*. Selanjutnya adalah mengatur posisi kamera, yang nantinya dari tampilan kamera ini hasil render akan ditampilkan.

3. Camera

Dengan mengatur posisi, resolusi, preview render sebelum dan sesudah dirender sebagai berikut.



Gambar 2.5 Animasi 3D yang sudah dalam tahap rendering

2.3 Analisis dan Perencanaan

2.3.1 Analisis Kebutuhan Hardware

Pemanfaatan *Parallel Computing Render Engine* pada *platform Open-Source Software*, membutuhkan banyak sumberdaya komputer yang nantinya akan digunakan sebagai mesin *rendering* yang berkerja secara bersama-sama atau *parallel*. Dalam penelitian ini terdapat 12 perangkat komputer dan 1 perangkat *switch*, detail analisis kebutuhan system sebagai berikut.

Table 2.4 Analisis Kebutuhan Hardware

Perangkat Komputer

No	Processor	Motherboard	Ram	Vga	Cardlan
1	Intel(R) Core(TM) i3 @3.20 GHz	Gigabyte	4 GB	ATI AMD Radeon HD 5450 (1024 MB)	Realtek PCIe GBE Family Controller
2	Intel(R) Core(TM) i3 @3.30 GHz	Gigabyte	4 GB	ATI AMD Radeon HD 5450 (1024 MB)	Realtek PCIe GBE Family Controller
3	Intel(R) Core(TM) i3 @3.30 GHz	Gigabyte	4 GB	ATI AMD Radeon HD 5450 (1024 MB)	Realtek PCIe GBE Family Controller
4	Intel(R) Core(TM) i3 @3.30 GHz	Gigabyte	4 GB	ATI AMD Radeon HD 5450 (1024 MB)	Realtek PCIe GBE Family Controller
5	Intel(R) Core(TM) i3 @3.30 GHz	Gigabyte	4 GB	ATI AMD Radeon HD 5450 (1024 MB)	Qualcomm Atheros AR8151 PCI-E Gigabit

No	Processor	Motherboard	Ram	Vga	Cardlan
					Ethernet Controller
6	Intel(R) Core(TM) i3 @3.30 GHz	Gigabyte	4 GB	ATI AMD Radeon HD 5450 (1024 MB)	Realtek PCIe GBE Family Controller
7	Intel(R) Core(TM) i3 @3.30 GHz	Gigabyte	4 GB	ATI AMD Radeon HD 5450 (1024 MB)	Realtek PCIe GBE Family Controller
8	Intel(R) Core(TM) i3 @3.30 GHz	Gigabyte	4 GB	ATI AMD Radeon HD 5450 (1024 MB)	Realtek PCIe GBE Family Controller
9	Intel(R) Core(TM) i3 @3.30 GHz	Gigabyte	4 GB	ATI AMD Radeon HD 5450 (1024 MB)	Realtek PCIe GBE Family Controller
10	Intel(R) Core(TM) i3 @3.30 GHz	Gigabyte	4 GB	ATI AMD Radeon HD 5450 (1024 MB)	Qualcomm Atheros AR8151 PCI-E Gigabit Ethernet Controller
11	Intel(R) Core(TM) i3 @3.30 GHz	Gigabyte	4 GB	ATI AMD Radeon HD 5450 (1024 MB)	Realtek PCIe GBE Family Controller
12	Intel(R) Core(TM) i3 @3.30 GHz	Gigabyte	4 GB	ATI Radeon HD 4300/4500	Qualcomm Atheros AR8151 PCI-E Gigabit Ethernet Controller

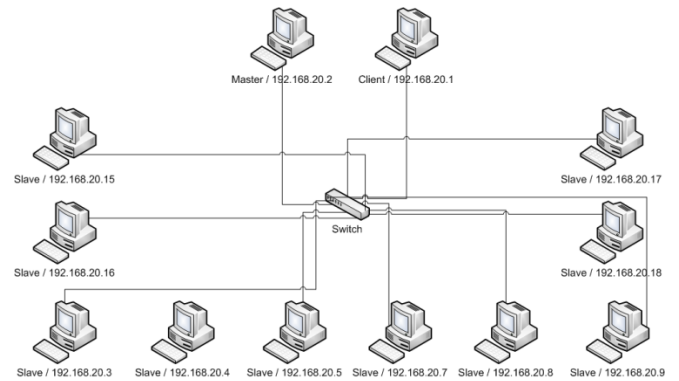
### 2.3.2 Perancangan Konfigurasi Jaringan

Setelah mempersiapkan kebutuhan *hardware*, kemudian 12 perangkat komputer dan 1 switch akan dikonfigurasi menjadi satu jaringan *Local Area Network (LAN)* dengan menggunakan topologi jaringan *Bus*

Table 2.5 Konfigurasi Software

No	IP Address	Mode Operation	Name	OS
1	192.168.20.2	MASTER	STMIK-3-2	Win 7 32bit
2	192.168.20.1	CLIENT	STMIK-3-1	Win 7 32bit
3	192.168.20.3	SLAVE	STMIK-3-3	Win 7 32bit
4	192.168.20.4	SLAVE	STMIK-3-4	Win 7 32bit
5	192.168.20.5	SLAVE	STMIK-3-5	Win 7 32bit
6	192.168.20.7	SLAVE	STMIK-3-7	Win 7 32bit
7	192.168.20.8	SLAVE	STMIK-3-8	Win 7 32bit
8	192.168.20.9	SLAVE	STMIK-3-9	Win 7 32bit
9	192.168.20.15	SLAVE	STMIK-3-15	Win 7 32bit
10	192.168.20.16	SLAVE	STMIK-3-16	Win 7 32bit
11	192.168.20.17	SLAVE	STMIK-3-17	Win 7 32bit
12	192.168.20.18	SLAVE	STMIK-3-18	Win 7 32bit

Sedangkan skema konfigurasi topologi jaringan *LAN* sebagai berikut.



Gambar 2.6 Skema Topologi jaringan

Setelah mengatur dan melakukan konfigurasi *LAN*, selanjutnya akan dilakukan konfigurasi pada *blender network render*. Dalam hal ini setiap sumberdaya komputer akan diidentifikasi sesuai fungsinya masing.

### 2.3.3 Konfigurasi Network Render

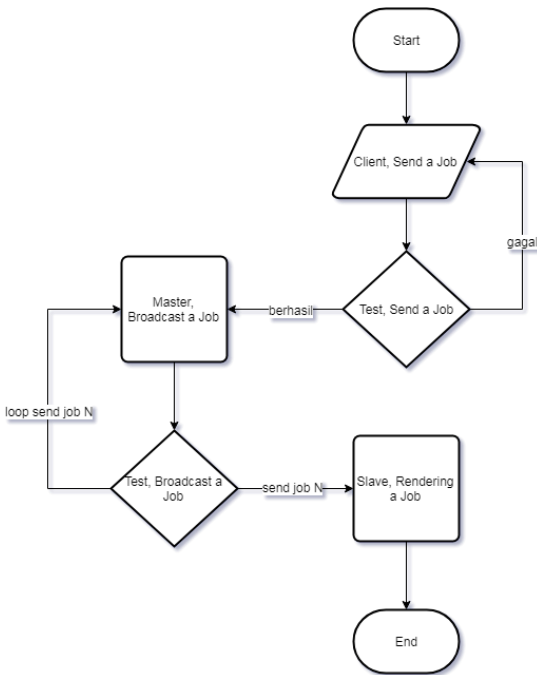
Seperti yang telah dijelaskan dalam pembahasan sebelumnya bahwa terdapat 1 komputer dengan mode *Master* yang berfungsi sebagai pengontrol dan mem-*broadcast* pekerjaan kesemua bagian yang lain, 1 komputer dengan mode *Client* yang berfungsi sebagai *client* yang mengirimkan pekerjaan ke komputer *master* selanjutnya mem-*broadcast* ke komputer *slave* dan 10 komputer dengan mode *Slave* yang berfungsi sebagai semua komputer yang menerima pekerjaan dari komputer *master*. Konfigurasi pada *properties render* tiap mode sebagai berikut.

Table 2.6 Konfigurasi blender Network Render

No	Mode	Properties
1	Master	Network setting (type: Master mode; address: default; port:8000; Start Service: active) Master setting (broadcast)
2	Client	Network setting (type: Client mode; address: 192.168.20.2; port:8000) Job setting (type: blender; name: default; engine: cycles; send job)
3	Slave	Network setting (type: Slave mode; address: 192.168.20.2; port:8000; Start Service: active) Slave setting (tag: non; render on slave: on; bake on slave: on; clear on exit: on; output rener on console:on; threads:auto-detect)

Gambaran skema alur kerja *blender network render* sebagai berikut.





Gambar 2.7 Slema Alur blender network render

Sedangkan Setelah melakukan konfigurasi pada *blender network render*, selanjutnya adalah tahap implementasi *rendering 3D Model*. Pada tahap berikutnya ini *Model 3D* yang telah dibuat akan di-render dengan *frame render* yang berbeda-beda.

**2.4 Implementasi Rendering 3D Model**

**1. Rendering Priority 1 and Chunks 5 Render**

Dari proses *rendering* tersebut dapat di lihat bahwa *start time: 11:34:25 AM – finished time: 11:35:26 AM = line time: 0 Jam 1 Menit 1 Detik*.

Pada saat proses *Rendering Priority 1 and Chunks 5 Render* sumberdaya komputer (*slave/node*) yang dimanfaatkan 10 komputer sekaligus dengan *ip address 192.168.20.3, 192.168.20.4, 192.168.20.5, 192.168.20.7, 192.168.20.15* dengan 2 *job 5* komputer, sedangkan *192.168.20.8, 192.168.20.9, 192.168.20.18, 192.168.20.17* dengan 1 *job 5* komputer

**2. Rendering Priority 1 and Chunks 10 Render**

Dari proses *rendering* tersebut dapat di lihat bahwa *start time: 11:36:57 AM – finished time: 11:37:56 AM = line time: 0 Jam 0 Menit 59 Detik*.

Pada saat proses *Rendering Priority 1 and Chunks 10 Render* sumberdaya komputer (*slave/node*) yang dimanfaatkan 10 komputer sekaligus dengan *ip address 192.168.20.18, 192.168.20.17* dengan 2 *job 2* komputer, sedangkan *192.168.20.3, 192.168.20.4, 192.168.20.5, 192.168.20.7, 192.168.20.8, 192.168.20.9, 192.168.20.16, 192.168.20.15* dengan 1 *job 8* komputer.

**3. Rendering Priority 10 and Chunks 10 Render**

Dari proses *rendering* tersebut dapat di lihat bahwa *start time: 11:39:34 AM – finished time: 11:40:33 AM = line time: 0 Jam 0 Menit 59 Detik*.

Pada saat proses *Rendering Priority 10 and Chunks 10 Render* sumberdaya komputer (*slave/node*) yang dimanfaatkan 10 komputer sekaligus dengan *ip address 192.168.20.8* dengan 2 *job 1* komputer, sedangkan *192.168.20.3, 192.168.20.4, 192.168.20.5, 192.168.20.7, 192.168.20.9, 192.168.20.18, 192.168.20.17, 192.168.20.16, 192.168.20.15* dengan 1 *job 9* komputer

Dari hasil 3 tahap implementasi *rendering 3D Animation*, dapat dilihat bahwa penggunaan *Chunks* dan *Priority Render* yang digunakan semaik cepat pula proses *rendering* yang dilakukan. Detail penggunaan *Chunks* dan *Priority Render* sebagai berikut

Table 2.7 Job Slave Render tiap frame render

Ip Address	Total Second		
	Prio 1 and Cnks 5	Prio 1 and Chks 10	Prio 10 and Chks 10
192.168.20.3	65	28	28
192.168.20.4	60	28	28
192.168.20.5	50	18.8	20
192.168.20.7	45	20	20
192.168.20.8	22	20	45
192.168.20.9	22	20	18
192.168.20.18	18	47.3	20
192.168.20.17	20	35	20
192.168.20.16	24	20	20
192.168.20.15	55	16	16
Total	381	253.1	235

Dari data *Chunks* dan *Priority Render* ini adakan diuji perbandingan tiap *frame* menggunakan teknik pengujian perbandingan *Anova one way*

**3 EVALUASI DAN HASIL**

**3.1 Perbandingan dengan Anova One Way**

Penentuan hipotesis awal:

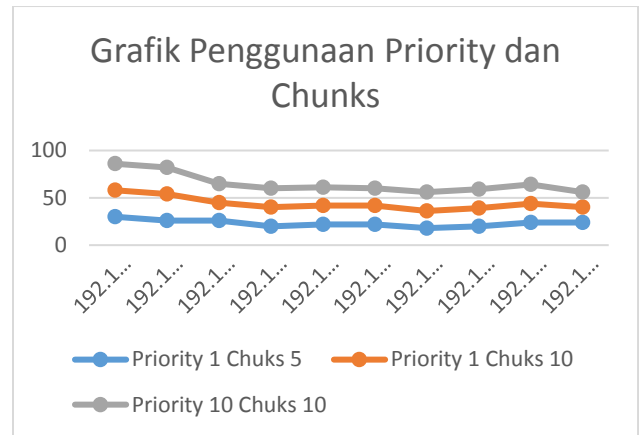
Ho: Tidak terdapat perbedaan penggunaan Waktu/*Second RENDER* dengan adanya penambahan *Chunks* dan *Priority Render* (penambahan *Chunks* dan *Priority Render* tidak berpengaruh terhadap penggunaan Waktu/*Second RENDER*).

Ha: Terdapat perbedaan penggunaan Waktu/*Second RENDER* dengan adanya penambahan *Chunks* dan *Priority Render* (penambahan *Chunks* dan *Priority Render* dapat mengurangi penggunaan Waktu/*Second RENDER*).

Table 3.1 *Job Slave Render* tiap *frame* render ( $X_1, X_2, X_3$ )

IP Address	Priority and Chunks						Jumlah Total	
	P: 1 C: 5 X <sub>1</sub>	X1 ^2	P: 1 C: 10 X <sub>2</sub>	X2 ^2	P: 10 C: 10 X <sub>3</sub>	X3 ^2	X <sub>tot</sub>	X <sub>tot</sub> ^2
192.168.20.3	65	4225	28	784	28	784	86	2468
192.168.20.4	60	3600	28	784	28	784	82	2244
192.168.20.5	50	2500	18	324	20	400	64.8	1429.44
192.168.20.7	45	2025	20	400	20	400	60	1200
192.168.20.8	22	484	20	400	45	2025	61	1245
192.168.20.9	22	484	20	400	18	324	60	1208
192.168.20.18	18	324	47	2209	20	400	56.1	1051.61
192.168.20.17	20	400	35	1225	20	400	59	1161
192.168.20.16	24	576	20	400	20	400	64	1376
192.168.20.15	55	3025	16	256	16	256	56	1088
Jml	381	17643	253.1	7239.73	235	6173	648.9	14471.05
$\bar{X}$	23.20		20.79		20.90		n	10

$\sum X^2$	5496		4466.05		4509	x	3
s	3.55		4.00		3.96	N	10x3=30
s <sup>2</sup>	12.62		15.98		15.66		



Gambar 3.8 *Grafik Penggunaan Priority and Chunks*

Diketahui:

- $\bar{X}$  : Rata-rata x
  - $\sum X^2$  : Mean X kuadrat
  - s : Simpangan baku
  - s<sup>2</sup> : Simpangan baku kuadrat
- sebelum analisis varian dilakukan perlu homogenitas varians terlebih dahulu dengan menggunakan uji F.

$$F \text{ hitung} = \frac{\text{Varianterbesar}}{\text{Varianterkecil}}$$

Dapat dilihat bahwa varian terbesar = 15.98 dan varian terkecil = 12.62 sehingga F hitung yang diperoleh:

$$F \text{ hitung} = \frac{15.98}{12.62} = 1.27$$

Selanjutnya F hitung akan dibandingkan dengan F table dengan df (degr freedom) pembilang n<sub>7</sub> – 1 dan df penyebut n<sub>1</sub> – 1, dalam hal ini nilai n<sub>7</sub> dan n<sub>1</sub> bernilai sama yaitu 10 dan setelah dikurang 1 nilainya 9. Berdasarkan Tabel F, maka F tabel untuk 5% = 3.18 sedangkan F table untuk 1% = 5.35. Harga F hitung lebih kecil dari F

table (1.27<3.18). Dengan demikian dapat disimpulkan bahwa varians data yang akan dianalisis homogen, sehingga perhitungan *anova* dapat dilanjutkan.

1. Jumlah Kuadrat Total ( $JK_{tot}$ )

$$JK_{tot} = \sum X_{tot}^2 - \frac{(\sum X_{tot})^2}{N}$$

$$= 31055.73 - \frac{869.1^2}{30}$$

$$JK_{tot} = 5877.9$$

2. Jumlah Kuadrat Antara ( $JK_{ant}$ )

$$JK_{ant} = \sum \frac{(\sum X_k)^2}{n_k} - \frac{(\sum X_{tot})^2}{N}$$

$$JK_{ant} = \frac{(\sum X_1)^2}{n_1} + \frac{(\sum X_2)^2}{n_2} + \dots + \frac{(\sum X_m)^2}{n_m} - \frac{(\sum X_{tot})^2}{N}$$

$$= \frac{381^2}{10} + \frac{253.1^2}{10} + \frac{235^2}{10} - \frac{869.1^2}{30}$$

$$= 14516.1 + 6405.96 + 5522.5 - 25177.83$$

$$= 1266.73$$

3. Jumlah Kuadrat dalam Kelompok ( $JK_{dal}$ )

$$JK_{dal} = JK_{tot} - JK_{ant}$$

$$JK_{dal} = 5877.9 - 1266.73$$

$$JK_{dal} = 4611.2$$

4. Menghitung Mean Kuadrat Antar Kelompok ( $MK_{ant}$ ).

$$MK_{ant} = \frac{JK_{ant}}{m-1}$$

$$= \frac{1266.73}{3-1}$$

$$= 633.37$$

5. Menghitung Mean Kuadrat Dalam Kelompok ( $MK_{dal}$ ).

$$MK_{dal} = \frac{JK_{dal}}{N-m}$$

$$= \frac{4611.2}{30-3}$$

$$= 170.78$$

6. Menghitung F Hitung ( $F_{hit}$ ).

$$F_{hitung} = \frac{MK_{ant}}{MK_{dal}}$$

$$= \frac{633.37}{170.78}$$

$$= 3.71$$

Table 3.2 Rencana Target Capaian Perhitungan

Sumber Validasi	df	Jumlah Kuadrat	MK	Fh	Ft	Kep
Total	30-1 = 29	5877.9				
Antar Kelompok	3-1=2	1266.73	633.37	3.71	5% 3.35	Fh>Ft Jadi Ha diterima
Dalam Kelompok	30-3=27	4611.2	170.78			

7. Menbandingkan harga F hitung dengan F table dengan df pembilang  $\{m(3) - 1 = 2\}$  dan df penyebut  $\{N(30) - m(3)=27\}$ , hasilnya adalah untuk probability 5% = 3.35, Harga F hitung ( $F_{hit}$ ) = **3.71** lebih besar dari F table ( $F_{tab}$ ) ( $F_h > F_t$ ) tertentu. Jadi bila F hitung lebih besar harga F table ( $F_h > F_t$ ) maka  $H_a$  diterima dan  $H_o$  ditolak.

8. Jadi **Terdapat perbedaan penggunaan Waktu/Second RENDER dengan adanya penambahan Chunks dan Priority Render (penambahan Chunks dan Priority Render dapat mengurangi penggunaan Waktu/Second RENDER Dengan Memanfaatkan Parallel Computingrender Engine Pada Platform Open-Source Software.**

### 3.2 Menentukan Korelasi Antar Sampel

1. Untuk mengetahui perbedaan sampel satu dengan sampel yang lain menggunakan t-test berpasangan.

2. Pasangan sampel yang akan diuji adalah:

a. Perbedaan Waktu/Second RENDER yang digunakan dari *Chunks 5 dan Priority 1 Render* dengan *Chunks 10 dan Priority 1* ( $X_1:X_2$ ).

b. Perbedaan Waktu/Second RENDER yang digunakan dari *Chunks 5 dan Priority 1 Render* dengan *Chunks 10 dan Priority 10* ( $X_1:X_3$ ).

c. Perbedaan Waktu/Second RENDER yang digunakan dari *Chunks 10 dan Priority 1 Render* dengan *Chunks 10 dan Priority 10* ( $X_2:X_3$ ).

3. Sedangkan Hipotesis yang diajukan adalah:

$X_1:X_2$

$H_o$ : Tidak Terdapat Perbedaan Waktu/Second RENDER Render Sebelum Menambah dari

*Chunks 5 dan Priority 1 Render Menjadi Chunks 10 dan Priority 1.*

Ha: Terdapat Perbedaan Waktu/Second RENDER Render Sebelum Menambah dari *Chunks 5 dan Priority 1 Render Menjadi Chunks 10 dan Priority 1.*

$X_1:X_3$

Ho: Tidak Terdapat Perbedaan Waktu/Second RENDER Render Sebelum Menambah dari *Chunks 5 dan Priority 1 Render Menjadi Chunks 10 dan Priority 1.*

Ha: Terdapat Perbedaan Waktu/Second RENDER Render Sebelum Menambah dari *Chunks 5 dan Priority 1 Render Menjadi Chunks 10 dan Priority 1.*

$X_2:X_3$

Ho: Tidak Terdapat Perbedaan Waktu/Second RENDER Render Sebelum Menambah dari *Chunks 10 dan Priority 1 Render Menjadi Chunks 10 dan Priority 10.*

Ha: Terdapat Perbedaan Waktu/Second RENDER Render Sebelum Menambah dari *Chunks 10 dan Priority 1 Render Menjadi Chunks 10 dan Priority 10.*

4. Mengetahui korelasi dua sampel sebagai berikut:

Korelasi  $X_1:X_2$  dari perhitungan ditemukan: -0.031692

Korelasi  $X_1:X_3$  dari perhitungan ditemukan: -0.0333

Korelasi  $X_2:X_3$  dari perhitungan ditemukan: -0.05683

5. Pengujian Hipotesis Pertama ( $X_1:X_2$ )

$$t = \frac{\bar{X}_1 - \bar{X}_2}{\sqrt{\frac{S_1^2}{n_1} + \frac{S_2^2}{n_2} - 2r\left(\frac{S_1}{\sqrt{n_1}}\right)\left(\frac{S_2}{\sqrt{n_2}}\right)}}$$

$$= \frac{38.10 - 25.31}{\sqrt{\frac{347.43}{10} + \frac{92.64}{10} - 2 \cdot (-0.3692) \left(\frac{18.64}{\sqrt{10}}\right) \left(\frac{9.63}{\sqrt{10}}\right)}}$$

$$= \frac{12.79}{\sqrt{34.743 + 9.264 - 0.634(5.894 \times 3.044)}}$$

$$= \mathbf{1.72}$$

Harga T hitung **1.72** uji dua pihak berbarti harga mutlak, selanjutnya harga T hitung

dibandingkan dengan T tabel ( $df = n_1 + n_2 - 2 = 10 + 10 - 2 = 18$ ) berdasarkan  $df = 18$ , untuk kesalahan 5%, maka harga T tabel = **2.10** (uji dua pihak). Ternyata harga T hitung (1.72) lebih kecil dari harga T tabel ( $1.72 < 2.10$ ). Dengan demikian Ha ditolak, dan Ho diterima. Kesimpulannya **Tidak Terdapat Perbedaan Waktu/Second RENDER Render Sebelum Menambah dari Chunks 5 dan Priority 1 Render Menjadi Chunks 10 dan Priority 1.**

6. Pengujian Hipotesis Kedua ( $X_2:X_3$ )

$$t = \frac{\bar{X}_1 - \bar{X}_3}{\sqrt{\frac{S_1^2}{n_1} + \frac{S_3^2}{n_3} - 2r\left(\frac{S_1}{\sqrt{n_1}}\right)\left(\frac{S_3}{\sqrt{n_3}}\right)}}$$

$$= \frac{38.10 - 23.50}{\sqrt{\frac{347.43}{10} + \frac{72.23}{10} - 2 \cdot (-0.0333) \left(\frac{18.64}{\sqrt{10}}\right) \left(\frac{8.50}{\sqrt{10}}\right)}}$$

$$= \frac{14.60}{\sqrt{34.743 + 7.228 - 0.067(5.894 \times 2.688)}}$$

$$= \mathbf{2.23}$$

Harga T hitung **2.23** uji dua pihak berbarti harga mutlak, selanjutnya harga T hitung dibandingkan dengan T tabel ( $df = n_1 + n_2 - 2 = 10 + 10 - 2 = 18$ ) berdasarkan  $df = 18$ , untuk kesalahan 5%, maka harga T tabel = **2.10** (uji dua pihak). Ternyata harga T hitung (2.23) lebih besar dari harga T tabel ( $2.23 > 2.10$ ). Dengan demikian Ha diterima, dan Ho ditolak. Kesimpulannya **Terdapat Perbedaan Waktu/Second RENDER Render Sebelum Menambah dari Chunks 5 dan Priority 1 Render Menjadi Chunks 10 dan Priority 10.**

7. Pengujian Hipotesis Ketiga ( $X_2:X_3$ )

$$t = \frac{\bar{X}_2 - \bar{X}_3}{\sqrt{\frac{S_2^2}{n_2} + \frac{S_3^2}{n_3} - 2r\left(\frac{S_2}{\sqrt{n_2}}\right)\left(\frac{S_3}{\sqrt{n_3}}\right)}}$$

$$= \frac{25.31 - 23.50}{\sqrt{\frac{92.64}{10} + \frac{72.28}{10} - 2 \cdot (-0.05683) \left(\frac{9.63}{\sqrt{10}}\right) \left(\frac{8.50}{\sqrt{10}}\right)}}$$

$$= \frac{1.81}{\sqrt{9.264 + 7.228 - 0.114(3.044 \times 2.688)}}$$

= **0.43**

Harga T hitung **0.43** uji dua pihak berbarti harga mutlak, selanjutnya harga T hitung dibandingkan dengan T tabel ( $df = n_1 + n_2 - 2 = 10 + 10 - 2 = 18$ ) berdasarkan  $df = 18$ , untuk kesalahan 5%, maka harga T tabel = **2.10** (uji dua pihak). Ternyata *harga T hitung (0.43) lebih kecil dari harga T tabel (0.43 < 2.10)*. Dengan demikian *Ha ditolak, dan Ho diterima. Kesimpulannya Tidak Terdapat Perbedaan Waktu/Second RENDER Render Sebelum Menambah dari Chunks 10 dan Priority 1 Render Menjadi Chunks 10 dan Priority 10.*

### 3.3 Hasil Pengujian Hipotesis

Dapat disimpulkan pada tahap ini, bahwa dengan menggunakan F uji Hipotesis Alternatif dapat diterima. Bahwa *Terdapat perbedaan penggunaan Waktu/Second RENDER dengan adanya penambahan Chunks dan Priority Render (penambahan Chunks dan Priority Render dapat mengurangi penggunaan Waktu/Second RENDER).*

Sedangkan pada perhitungan korelasi antara dua sampel yang berbeda, dengan menggunakan t uji terdapat satu korelasi yang Hipotesis Alternatif dapat diterima yaitu korelasi antara  $X_1$  dengan  $X_3$  bahwa *harga T hitung (2.23) lebih besar dari harga T tabel (2.23 < 2.10) Terdapat Perbedaan Waktu/Second RENDER Render Sebelum Menambah dari Chunks 5 dan Priority 1 Render Menjadi Chunks 10 dan Priority 10.*

Hal ini dikarenakan sampel korelasi yang diuji terdapat selisih *Chunks* dan *Priority render* yang cukup besar. Sedangkan korelasi yang Hipotesis Alternatifnya ditolak yaitu  $X_1$  dengan  $X_2$  dan  $X_2$  dengan  $X_3$  karena selisih *Chunks* dan *Priority render* yang diuji hanya sedikit.

Dari hasil pengujian yang telah disimpulkan bahwa, pada penelitian sebelumnya dengan *rendering* pada *3D Model* dan penelitian ini dengan *rendering 3D Animation*, mempunyai hasil yang sama. Bahwa dengan menambah *frame*, *chunks* atau *priority render* dapat mempengaruhi banyaknya sumber daya *node/slave* yang digunakan pada saat render, dan mempengaruhi waktu/*second rendering*.

## 4 KESIMPULAN

Dari hasil yang telah dicapai terkait dengan komparasi banyaknya *Chunks* dan *Priority Render* apakah akan mempengaruhi *Waktu/Second Render* yang digunakan dengan menggunakan teknik statistik *Anova one way*, Hipotesis Alternatif dapat diterima. Karena nilai F hitung lebih besar dibandingkan dengan F table ( $F_h > F_t$ ) dengan nilai **3.71** > 3.35, Jadi *Terdapat perbedaan penggunaan Waktu/Second RENDER dengan adanya penambahan Chunks dan Priority Render (penambahan Chunks dan Priority Render dapat mengurangi penggunaan Waktu/Second Render) Dengan Memanfaatkan Parallel Computingrender Engine Pada Platform Open-Source Software.*

Sedangkan pada perhitungan korelasi antara dua sampel yang berbeda, dengan menggunakan t uji terdapat satu korelasi yang Hipotesis Alternatif dapat diterima yaitu korelasi antara  $X_1$  dengan  $X_3$  bahwa *harga T hitung (2.23) lebih besar dari harga T tabel (2.23 < 2.10) Terdapat Perbedaan Waktu/Second RENDER Render Sebelum Menambah dari Chunks 5 dan Priority 1 Render Menjadi Chunks 10 dan Priority 10.*

Hal ini dikarenakan sampel korelasi yang diuji terdapat selisih *Chunks* dan *Priority render* yang cukup besar. Sedangkan korelasi yang Hipotesis Alternatifnya ditolak yaitu  $X_1$  dengan  $X_2$  dan  $X_2$  dengan  $X_3$  karena selisih *Chunks* dan *Priority render* yang diuji hanya sedikit.

Dari hasil pengujian yang telah disimpulkan bahwa, pada penelitian sebelumnya dengan *rendering* pada *3D Model* dan penelitian ini dengan *rendering 3D Animation*, mempunyai hasil yang sama. Bahwa dengan menambah *frame*, *chunks* atau *priority render* dapat mempengaruhi banyaknya sumber daya *node/slave* yang digunakan pada saat render, dan mempengaruhi waktu/*second rendering*.

## DAFTAR PUSTAKA

- [1 M. Z. Patoli, M. Gkion and A. Al-Barakati, ] "An open source Grid based render farm for Blender 3D," in *Power Systems Conference and Exposition, 2009. PSCE '09. IEEE/PES,*

Seattle, WA, USA, 2009.

[2 Z. Zhurong, D. Wei and Q. Yuhui, "A Grid Based Graphics Rendering Design," in *Information Technology and Applications, 2009. IFITA '09. International Forum on*, Chengdu, China, 2009.

[3 I. Sofian, CISCO CCNA & Jaringan Komputer, Bandung: Informatika, 2010.

[4 I. Kurniawan and M. R. Maulana, ] PERBANDINGAN BANYAKNYA FRAME RENDER DENGAN PENGGUNAAN SUMBERDAYA YANG DIGUNAKAN MENGGUNAKAN ANOVA ONE WAY DENGAN MEMANFAATKAN PARALLEL COMPUTING RENDER ENGINE, vol. 13, Pekalongan: ICTech, 2018.

[5 S. Verma and K. Patel, "Association between shopping habit and demographics of m-commerce user's in India using two way ANOVA," in *Convergence in Technology (I2CT), 2017 2nd International Conference for*, Mumbai, India, 2017.

[6 P. Bokoro and K. Malandala, "Condition assessment of SF6 circuit breakers using analysis of variance (ANOVA)," in *AFRICON, 2017 IEEE*, Cape Town, South Africa, 2017.

[7 A. Rumagit, "IMPLEMENTASI GRID

] COMPUTING UNTUK HIGH THROUGHPUT COMPUTING," *JURNAL TEKNIK INFORMATIKA UNIVERSITAS SAM RATULANGI*, vol. 2, no. 1, Manado 2013.

[8 M. Lysy C, M. R. Arthur and A. S. Brave, ] "Implementasi Cluster Computing Untuk Render Animasi," *E-journal Teknik Elektro dan Komputer*, vol. 2, no. 3, pp. 1-8, 2013.

[9 Blender, "About," Blender.org, [Online]. ] Available: <https://www.blender.org/about/>. [Accessed 9 Mei 2017].

[1 Blender, "Network Render," Blender, 0] [Online]. Available: [https://wiki.blender.org/index.php/Dev:Source/Render/Cycles/Network\\_Render](https://wiki.blender.org/index.php/Dev:Source/Render/Cycles/Network_Render). [Accessed 9 Mei 2017].

[1 P. D. Sugiyono, Statistik untuk Penelitian, 1] Bandung: Penerbit Alfabeta, 2017.