

## PEMANFAATAN *SCRIPT PYTHON* UNTUK MEMBUAT *3D ANIMATION* PADA *OPEN-SOURCE SOFTWARE*

Ichwan Kurniawan, Much. Rifqi Maulana, Dicke JSH Siregar

Program Studi Teknik Informatika, STMIK Widya Pratama

Jl. Patriot 25 Pekalongan

Telp (0285)427816

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

### ABSTRAKS

Dalam pembuatan animasi komputer, semua aktifitas objek disimpan dalam sebuah *keyframe* kemudian setiap *keyframe* disimpan dalam suatu *frame* (Pranatio and Kosala 2010). *Animation* dibuat berdasarkan rentang waktu animasi dalam satu detik, hal ini disebut dengan *frame rate/frame per second(FPS)*. Animasi, baik 2D atau 3D dibuat berdasarkan rentang waktu animasi dalam satu detik, hal ini disebut dengan *frame rate/frame per second(FPS)*. Secara umum metode untuk menghemat waktu *rendering* adalah dengan mengurangi kompleksitas gerakan, namun hal ini akan sedikit mempeharuhi hasil akhir animasi (V. Patil and L. Deshpande 2016). Dengan memanfaatkan komputer dalam produksi animasi, membutuhkan spesifikasi *hardware* yang mencukupi untuk menghasilkan hasil akhir yang memuaskan. pembuatan animasi komputer, selain menggunakan teknik *keyframe*, animasi dapat dibuat dengan memanfaatkan script program. Oleh karena itu dengan memanfaatkan *python script* dalam pembuatan animasi dapat meminimalkan waktu *render* tanpa mengurangi kualitas *render*, dan dapat membuat animasi dengan presisi nilai yang sesuai dengan kenyataannya. Tujuan dari penelitian ini adalah Membandingkan pembuatan *3D Animation Rotation, Location* dan *Scale* dengan teknik *keyframe* dengan teknik *script python* pada *software blender* apakah akan mempengaruhi waktu *rendering* dengan Teknik *Anova one way*.

Hasil pengujian *3d animation rotation*, bahwa  $F$  hitung lebih kecil harga  $F$  table ( $F_h < F_t$ ) maka  $H_o$  diterima dan  $H_a$  ditolak. Jadi **Tidak terdapat penurunan *time rendering* (lama waktu) terhadap penggunaan *Script Python 3D Animation ROTATION* dengan *Keyframe 3D Animation ROTATION*.**

Hasil pengujian *3d animation location*, bahwa  $F$  hitung lebih kecil harga  $F$  table ( $F_h < F_t$ ) maka  $H_o$  diterima dan  $H_a$  ditolak. Jadi **Tidak terdapat penurunan *time rendering* (lama waktu) terhadap penggunaan *Script Python 3D Animation LOCATION* dengan *Keyframe 3D Animation LOCATION*.**

Sedangkan untuk pengujian *3d animation scale*, bahwa  $F$  hitung lebih kecil harga  $F$  table ( $F_h < F_t$ ) maka  $H_o$  diterima dan  $H_a$  ditolak. Jadi **Tidak terdapat penurunan *time rendering* (lama waktu) terhadap penggunaan *Script Python 3D Animation SCALE* dengan *Keyframe 3D Animation SCALE*.**

Hasil Evaluasi *3d animation rotation, location* dan *scale* dengan menggunakan teknik *Anova* adalah  $H_o$  dapat diterima, karena nilai  $F$  hitung lebih kecil dari  $F$  table dan  $H_a$  ditolak. Hal ini membuktikan bahwa dengan menggunakan Teknik *Anova*, **Tidak terdapat penurunan *time rendering* (lama waktu) terdapat penggunaan *script python* dengan *keyframe***, hal ini dikarenakan selisih *time rendering* (lama waktu) antara penggunaan *script python* dan *keyframe* berubah tidak signifikan.

**Keywords:** *Script Python, Keyframe, Render 3D Model, open-source software*

## 1 PENDAHULUAN

### 1.1 Latar Belakang

Penggunaan animasi merupakan bagian yang penting dalam interface pada saat ini (Kramer, et al. 2016). Pemanfaatan animasi tidak hanya diduni huburan saja, namun sekarang penggunaan animasi telah berkembang dalam bidang-bidang yang lain. Penempatan gerak dimasi pada objek

statis yang tidak bergerak dalam deretan waktu merupakan mekanisme proses animasi (V. Patil and L. Deshpande 2016). Merangkai gerakan objek statis dalam runtutan waktu merupakan pembuatan animasi tradisional. Terdapat beberapa teknik dalam membuat animasi secara tradisional, yang sering digunakan oleh animator diantaranya adalah teknik *stop motion, frame by frame, cut out*

dan sebagainya. Dengan menggunakan teknik tradisional ini produksi animasi membutuhkan waktu yang lama. Dengan menggunakan komputer proses produksi animasi menjadi lebih mudah.

Setiap gerakan dalam animasi akan disimpan dalam sebuah *keyframe* yang diletakkan dalam sebuah urutan *frame* (Pranatio and Kosala 2010). Animasi, baik 2D atau 3D dibuat berdasarkan rentang waktu animasi dalam satu detik, hal ini disebut dengan *frame rate/frame per second(FPS)*. Animasi komputer terbentuk karena komputer akan menciptakan gerakan diantara *keyframe*, kemudian runtutan gerakan di dalam *frame* akan dikonversi melalui proses *rendering* menjadi format video. Namun dalam proses *rendering* dapat memakan waktu lebih lama, karena komputer akan menterjemahkan semua gambar yang terdapat pada *frame render*. Semakin detail gambar dan gerakan dalam animasi akan mempengaruhi lama waktu *rendering* sebuah animasi.

Secara umum metode untuk menghemat waktu *rendering* adalah dengan mengurangi kompleksitas gerakan, namun hal ini akan sedikit memengaruhi hasil akhir animasi (V. Patil and L. Deshpande 2016). Dengan memanfaatkan komputer dalam produksi animasi, membutuhkan spesifikasi *hardware* yang mencukupi untuk menghasilkan hasil akhir yang memuaskan. Selain mengurangi kompleksitas gerakan terdapat metode lain untuk menghemat waktu render, yaitu menggunakan metode *expression* atau menggunakan pemrograman komputer.

Perbandingan dua metode *frame by frame* dan *expression* adalah untuk *rendering process*, metode *expression (script)* lebih unggul 13 menit lebih cepat dibandingkan dengan metode *frame by frame* pada komputer dengan spesifikasi menengah, sedangkan untuk spesifikasi tinggi sama-sama 17 menit (Pramono, Suyanto and Sofyan 2017). Dalam penelitian tersebut belum diuji pada animasi tiga dimensi (3D), dengan menggunakan pemrograman komputer proses animasi dapat dijalankan melalui sintak program di setiap *frame* (Kramer, et al. 2016). Hal ini dapat memungkinkan proses *rendering* dapat dijalankan dengan lebih cepat, karena gerakan dan *keyframe* dijalankan melalui intruksi program komputer. Selain itu dengan memanfaatkan script

program memungkinkan meminimalkan kerja komputer dalam proses rendering.

*Python script* dapat digunakan untuk membangun perangkat lunak simulasi system transportasi vertikal, penggunaan *python script* salah satunya dimanfaatkan pada proses penganimasian (Chaosangket, et al. 2018). Animasi dapat diciptakan dengan memanfaatkan script pemrograman, hal ini bertujuan agar animasi dapat tercipta dengan presisi nilai yang sesuai dengan kenyataan.

Dari pemaparan di atas bahwa dalam pembuatan animasi komputer, selain menggunakan teknik *keyframe*, animasi dapat dibuat dengan memanfaatkan script program. Oleh karena itu dengan memanfaatkan *python script* dalam pembuatan animasi dapat meminimalkan waktu *render* tanpa mengurangi kualitas *render*, dan dapat membuat animasi dengan presisi nilai yang sesuai dengan kenyataannya.

Dalam penelitian ini, akan memanfaatkan *Python Script* untuk pembuatan *3D Animation* pada *open-space software* dengan membandingkan pembuatan animasi dengan teknik *keyframe* dengan teknik *script python* pada *software blender* apakah akan mempengaruhi waktu *rendering*. Dalam penelitian ini, teknik statistik yang digunakan untuk menguji hipotesis menggunakan teknik *Analisis of Variants (Anova) one way*.

*Analisis of Variants (Anova)* merupakan teknik uji statistik yang digunakan untuk membandingkan dua atau lebih variable penelitian (Verma and Patel 2017). Menurut G. Keller (2014) *Anova* umumnya dapat digunakan sebagai pengujian signifikansi dari pengamatan experimental atau perbedaannya (Bokoro and Malandala 2017). Dengan menggunakan teknik statistik hipotesis *Anova*, pembuatan animasi dengan teknik *keyframe* dengan teknik *script python* pada *software blender* akan dibandingkan apakah akan mempengaruhi waktu *rendering*.

## 1.2 Landasan Teori

### 1.2.1 Script Python

Python adalah bahasa pemrograman yang *interpreted, interactive* dan bahasa pemrograman berbasis *object oriented programming*. Hal ini mencakup *modules, exceptions, dynamic typing, very high-level dynamic data type*, dan *class* (Foundation 2018). Bahasa pemrograman *python* merupakan bahasa pemrograman berorientasi

objek yang mudah digunakan dan telah mendukung tingkat bahasa pemrograman level tinggi.

*Script Python* adalah cara baik dan mudah untuk mengeksplorasi fungsi-fungsi yang terdapat pada *blender*. Sebagian besar fungsi pada *blender* dapat dituliskan dengan skrip ini, termasuk dalam membuat animasi, *rendering*, import dan export, pembuatan objek dan penulisan perintah-perintah pemrograman (Foundation 2018). Penggunaan fungsi-fungsi yang terdapat di *blender*, selain menggunakan propertie yang telah disediakan, pengguna dapat menambahkan atau menggunakan fungsi-fungsi melalui *script python*. Dengan *script python* fungsi-fungsi tambahan dapat memaksimalkan kerja pada *blender*.

### 1.2.2 *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 (Lysy C, Arthur and Brave 2013). 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. 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 (Rumagit 2013). Dapat disimpulkan bahwa waktu yang dibutuhkan dalam proses render animasi dapat dilihat dari banyak frame yang dimiliki dalam sebuah adegan.

Pada umumnya proses render yang dilakukan pada saat ini masih bekerja secara single dengan menggunakan sebuah mesin yang memiliki sumber daya yang besar. (Lysy C, Arthur and Brave 2013). Hal ini dikarenakan satu sumberdaya komputer menangani semua proses render, jika proses render dikerjakan

menggunakan banyak sumberdaya, proses render akan dibagi menjadi beberapa bagian dan akan didistribusikan kesemua sumberdaya yang ada. Dengan cara seperti itu proses render akan menjadi lebih singkat dan lebih cepat.

### 1.2.3 *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* (*Blender n.d.*). Dalam aplikasi *blender* karena bersifat *open source* dapat meminimalkan beban biaya dan kemudian dengan aplikasi *blender* fitur-fitur yang dimiliki lebih variatif.

### 1.2.4 *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 smpa 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* (Sugiyono 2017).

*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  $n = \text{jumlah sample}$ ,  $M = \text{mean/rata-rata}$  sedangkan  $s^2 = \text{varians}$  (Sugiyono 2017).

Dalam *Anova* terdapat (Sugiyono 2017):

1. Deviasi Total, yaitu jarak antara nilai individual yang ada dalam seluruh sampel dengan Mean Total. Dalam hal ini misal  $(X - M_{\text{tot}})$ .
2. Deviasi antara kelompok (*between*), yaitu jarak antara Mean setiap kelompok dengan Mean Total. Dalam hal ini misal  $(M_4 - M_{\text{tot}})$ .
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 (Sugiyono 2017), 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 didampingi 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 (Sugiyono 2017):

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.

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

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 dengan penambahan pengaturan modifier, lighting, camera, material dan texture, yang nantinya akan digunakan sebagai bahan penelitian.
2. Membuat Animasi dengan menambahkan *Keyframe* melalui *Properties*, dalam hal ini animasi *rotation*, *location* dan *scaling*.

- 3 Membuat Animasi dengan menambahkan *Script Python* melalui *Text Editor*, dalam hal ini animasi *rotation, location* dan *scaling*.
- 4 Mengatur konfigurasi *cycles rendering* pada *blender*
- 5 Implementasi *rendering 3D Animation* dengan menggunakan *blender cycles render*, dalam hal ini *experiment* dilakukan dengan melakukan uji coba dengan *render 3D Animation* dengan *script python* dan *keyframe*, dalam hal ini animasi *rotation, location* dan *scaling*.
- 6 Evaluasi dengan melakukan komparasi *rendering 3D Animation* dengan *script python* dan *keyframe* apakah akan mempengaruhi waktu *rendering*. Dalam hal ini dalam tahap uji komparasi akan menggunakan teknik statistik *Anova one way* dengan menentukan hipotesis *Ha* dan *Ho* terlebih dahulu, kemudian akan mengukur tingkat korelasi antara antar sampel dengan sampel yang lain untuk mengetahui perbandingan waktu *rendering* antara *render 3D Animation* dengan *script python* dan *keyframe*.

**2.2 Pengumpulan Data**

**1. Model 3D**

Dalam pembuatan model 3d hanya terdapat satu objek balok detail dari properties model 3d sebagai berikut.

Table 2.1 *Properties Model 3D*

No	Model	Properties	Material
1	Balok :	Location: x (0.0), y (0.0), z (0.0) Rotation: x (0.0), y (0.0), z (0.0) Scaling: x (0.104), y (0.104), z (0.104)	Glossy Diffuse BSDF:GGX; color: 000000;

**2. Lighting**

Kemudian setelah model 3d telah terbentuk, selanjutnya adalah mengatur pencahayaan pada lingkungan render. Ada 3 type lighting yang dimanfaatkan yaitu: *Spot Light, Point Light* dan *Emission Light*.

Table 2.2 *Properties Lighting*

No	Type	Properties
1	Point Light	Size: 0.100; Max Bounces: 1024; color: FFFFFFFF; Cast Shadow; Multiple Importance

**3. Camera**

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

**4. Membuat Animasi dengan *Script Python***

Pembuatan *3d animastion* dengan memanfaatkan *script python* dengan menambahkan skrip program pada *text editor blender*, dalam hal ini membuat 3 (tiga) gerakan dasar pada objek balok, *Rotation, Location* dan *Scalling*.

**5. Membuat Animasi dengan *Properties Keyframe***

Pembuatan *3d animastion* dengan memanfaatkan *properties keyframe* dengan menambahkan *keyframe* pada *properties blender*, dalam hal ini membuat 3 (tiga) gerakan dasar pada objek balok, *Rotation, Location* dan *Scalling*. Artinya semua aktifitas pada objek akan disimpan kedalam *key* yang terdapat pada nomer *frame*.

**6. Konfigurasi *Cycles Render Blender***

Seperti yang telah dijelaskan dalam pembahasan sebelumnya bahwa terdapat 30 *frame rendering*, dimana setiap *frame rendering* akan dianalisis berapa lama waktu *render* yang dibutuhkan. Adapun konfigurasi untuk *Cycle Render Blender* sebagai berikut:

Table 2.3 Konfigurasi *Cycle Render Blender*

No	Properties	Value
1	Resolution	x: 1920px y: 1080px scale: 50%
2	Frame Range	Start Frame: 1 End Frame: 30 Frame Step: 1
3	Frame Rate	24 fps
4	Output	File format: AVI JPEG Quality: 90%
5	Sampling	Render: 128 Preview: 128

**2.3 Analisis dan Perencanaan**

Spesifikasi *Hardware* dan *Software* yang digunakan untuk *3d modeling* dan *animation* adalah sebagai berikut:

Table 2.4 *Spesifikasi Kebutuhan Hardware Perangkat Komputer*

No	Processor	Mothe rboard	Ram	Vga	Software
1	Intel(R) Core(TM) i7-7700 @3.60 GHz	Dell	8 GB	AMD Radeon RX 550 400Mhz	Windows 10 Enterprise 64-bit Blender 2.79b 64-bit

**2.4 Implementasi *Rendering 3D Model***

Pada pengujian ini dilakukan 2 tahap *rendering* dengan jumlah *frame render* 30 *frame*. Rentang *frame* dari *start frame 1* dan *end frame 30* dengan *frame per second/frame rate 24 fps*. Jika dikonversi dalam satuan detik, durasi animasi

yang dihasilkan adalah 30 detik dibagi 24 fps sama dengan 1.25 detik.

Pertama 3d animation dengan script python, kemudian pada saat proses rendering akan dilakukan pengamatan berapa lama waktu yang dibutuhkan tiap frame rendering. Selanjutnya 3d animation dengan keyframe properties, sama dengan tahap yang pertama pada saat proses rendering akan dilakukan pengamatan berapa lama waktu yang dibutuhkan tiap frame rendering. Dalam hal ini terdapat 3 animasi yang akan di-render, yaitu animasi dengan rotation, location dan scale.

1. Rendering Rotation Animation dengan Script Python
2. Rendering Location Animation dengan Script Python
3. Rendering Scale Animation dengan Script Python
4. Rendering Rotation Animation dengan Properties Keyframe
5. Rendering Location Animation dengan Properties Keyframe
6. Rendering Scale Animation dengan Properties Keyframe

Dari hasil 6 tahap implementasi rendering 3d animation rotation, location dan scale untuk animasi dengan script python maupun keyframe properties, dapat dilihat bahwa terdapat perbedaan lama waktu rendering antara 3d animation dengan script python maupun keyframe properties. Detail penggunaan sumberdaya tiap frame render sebagai berikut.

Table 2.5 Job Slave Render tiap frame render

Dari data frame rendering tiap 3d animation, baik rotation, location dan scale ini adakan diuji komparasi teknik pembuatan 3d animation dengan script python dan keyframe propertie menggunakan teknik pengujian perbandingan Anova one way.

### 3 EVALUASI DAN HASIL

#### 3.1 Evaluasi 3D Animation Rotatoion

Penentuan hipotesis awal:

Ho : Tidak terdapat penurunan time rendering (lama waktu) terhadap penggunaan Script Python 3D Animation ROTATION dengan Keyframe 3D Animation ROTATION.

Ha : Terdapat penurunan time rendering (lama waktu) terhadap penggunaan Script Python 3D

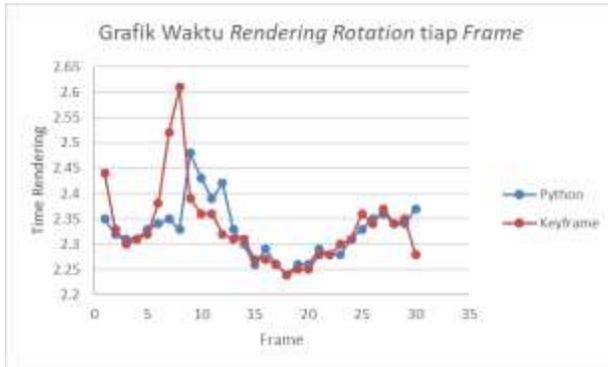
Animation ROTATION dengan Keyframe 3D Animation ROTATION

Pengujian Homogenitas Varians

Table 3.1 Time Rendering tiap frame render ( $X_1, X_2$ )

Frame	Time Rendering (s)				N	
	$X_1$	$X_1^2$	$X_2$	$X_2^2$	$X_{tot}$	$X_{tot}^2$
1	2.35	5.52	2.44	5.95	4.79	11.48
2	2.32	5.38	2.33	5.43	4.65	10.81
3	2.31	5.34	2.3	5.29	4.61	10.63
4	2.31	5.34	2.31	5.34	4.62	10.67
5	2.33	5.43	2.32	5.38	4.65	10.81
6	2.34	5.48	2.38	5.66	4.72	11.14
7	2.35	5.52	2.52	6.35	4.87	11.87
8	2.33	5.43	2.61	6.81	4.94	12.24
9	2.48	6.15	2.39	5.71	4.87	11.86
10	2.43	5.90	2.36	5.57	4.79	11.47
11	2.39	5.71	2.36	5.57	4.75	11.28
12	2.42	5.86	2.32	5.38	4.74	11.24
13	2.33	5.43	2.31	5.34	4.64	10.77
14	2.3	5.29	2.31	5.34	4.61	10.63
15	2.26	5.11	2.27	5.15	4.53	10.26
16	2.29	5.24	2.27	5.15	4.56	10.40
17	2.26	5.11	2.26	5.11	4.52	10.22
18	2.24	5.02	2.24	5.02	4.48	10.04
19	2.26	5.11	2.25	5.06	4.51	10.17
20	2.26	5.11	2.25	5.06	4.51	10.17
21	2.29	5.24	2.28	5.20	4.57	10.44
22	2.28	5.20	2.28	5.20	4.56	10.40
23	2.28	5.20	2.3	5.29	4.58	10.49
24	2.31	5.34	2.31	5.34	4.62	10.67
25	2.33	5.43	2.36	5.57	4.69	11.00
26	2.35	5.52	2.34	5.48	4.69	11.00
27	2.36	5.57	2.37	5.62	4.73	11.19
28	2.34	5.48	2.34	5.48	4.68	10.95
29	2.34	5.48	2.35	5.52	4.69	11.00
30	2.37	5.62	2.28	5.20	4.65	10.82

Frame	Time Rendering (s)				N	
	X <sub>1</sub>	X <sub>1</sub> <sup>2</sup>	X <sub>2</sub>	X <sub>2</sub> <sup>2</sup>	X <sub>tot</sub>	X <sub>tot</sub> <sup>2</sup>
JML	69.81	162.53	70.01	163.56	139.82	326.10
$\bar{X}$	2.33		2.33		n	30
$\sum X^2$	162.53		163.56		x	2
s <sup>2</sup>	0.00		0.01		N	30*2=60
s	0.05		0.08			



Gambar 3.1 Grafik Waktu Rendering tiap Frame Animation Rotation

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 = 0.0063 dan varian terkecil = 0.0030 sehingga F hitung yang diperoleh:

$$F \text{ hitung} = \frac{0.0063}{0.0030} = \mathbf{2.11}$$

Selanjutnya F hitung akan dibandingkan dengan F table dengan df (degr freedom) pembilang m – 1 dan df penyebut N – 1. Demgan demikian dk pembilang = 2-1 = 1 dan dk penyebut = 60-2 = 58. Berdasarkan Tabel F, maka F tabel untuk 5% = 4.01 sedangkan F table untuk 1% = 7.09. Harga F hitung lebih kecil dari F table (**2.11<4.01**). Dengan demikian dapat disimpulkan bahwa varians data yang akan dianalisis homogen, sehingga perhitungan *anova* dapat dilanjutkan.

1. Jumlah Kuadrat Total (JK<sub>tot</sub>)

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

$$= 326.10 - \frac{139.82^2}{60}$$

$$JK_{tot} = \mathbf{0.268}$$

2. Jumlah Kuadrat Antara (JK<sub>ant</sub>)

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

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

$$= \frac{69.81^2}{30} + \frac{70.01^2}{30} - \frac{139.82^2}{60}$$

$$= 162.45 + 163.38 - 325.83$$

$$= \mathbf{0.001}$$

3. Jumlah Kuadrat dalam Kelompok (JK<sub>dal</sub>)

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

$$JK_{dal} = 0.268 - 0.001$$

$$JK_{dal} = \mathbf{0.267}$$

4. Menghitung Mean Kuadrat Antar Kelompok (MK<sub>ant</sub>).

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

$$= \frac{0.001}{2 - 1}$$

$$= \mathbf{0.001}$$

5. Menghitung Mean Kuadrat Dalam Kelompok (MK<sub>dal</sub>).

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

$$= \frac{0.267}{60 - 2}$$

$$= \mathbf{0.005}$$

6. Menghitung F Hitung (F<sub>hit</sub>).

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

$$= \frac{0.001}{0.005}$$

$$= \mathbf{0.145}$$

Table 3.2 Rencana Target Capaian Perhitungan

Sumber Validasi	df	Jumlah Kuadrat	MK	Fh	Ft	Kep
Total	60-1 = 59	0.268		0.145	5% 4.01	Fh < Ft

Sumber Validasi	df	Jumlah Kuadrat	MK	Fh	Ft	Kep
Antar Kelompok	2-1=1	0.001	0.001		1% 7.09	Jadi Ha diterima
Dalam Kelompok	60-2=58	0.267	0.005			

- Menbandingkan harga F hitung dengan F table dengan df pembilang {m(2) - 1 = 1} dan df penyebut {N(60) - m(2)=58}, hasilnya adalah 4.01. F hitung ( $F_{hit}$ ) = 0.145, jadi  $0.145 < 4.01$  untuk probability 5% dan  $0.145 < 7.09$  untuk probality 1% , Harga F hitung ( $F_{hit}$ ) lebih kecil dari F table ( $F_{tab}$ ) ( $F_h < F_t$ ) tertentu. Jadi bila F hitung lebih kecil harga F table ( $F_h < F_t$ ) maka  $H_0$  diterima dan  $H_a$  ditolak.
- Jadi **Tidak terdapat penurunan time rendering (lama waktu) terhadap penggunaan Script Python 3D Animation ROTATION dengan Keyframe 3D Animation ROTATION.**

### 3.2 Evaluasi 3D Animation Location

Penentuan hipotesis awal:

$H_0$  : Tidak terdapat penurunan *time rendering* (lama waktu) terhadap penggunaan *Script Python 3D Animation LOCATION* dengan *Keyframe 3D Animation LOCATION*.

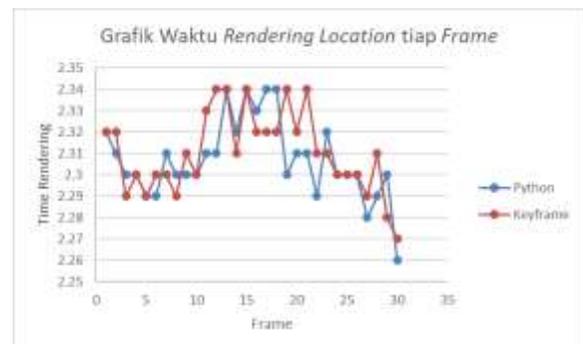
$H_a$  : Terdapat penurunan *time rendering* (lama waktu) terhadap penggunaan *Script Python 3D Animation LOCATION* dengan *Keyframe 3D Animation LOCATION*

Pengujian Homogenitas Varians

Table 3.3 *Time Rendering tiap frame render* ( $X_1, X_2$ )

Frame	Time Rendering (s)				N	
	$X_1$	$X_1^2$	$X_2$	$X_2^2$	$X_{tot}$	$X_{tot}^2$
1	2.32	5.38	2.32	5.38	4.64	10.76
2	2.31	5.34	2.32	5.38	4.63	10.72
3	2.3	5.29	2.29	5.24	4.59	10.53
4	2.3	5.29	2.3	5.29	4.6	10.58
5	2.29	5.24	2.29	5.24	4.58	10.49
6	2.29	5.24	2.3	5.29	4.59	10.53
7	2.31	5.34	2.3	5.29	4.61	10.63
8	2.3	5.29	2.29	5.24	4.59	10.53
9	2.3	5.29	2.31	5.34	4.61	10.63

Frame	Time Rendering (s)				N	
	$X_1$	$X_1^2$	$X_2$	$X_2^2$	$X_{tot}$	$X_{tot}^2$
10	2.3	5.29	2.3	5.29	4.6	10.58
11	2.31	5.34	2.33	5.43	4.64	10.77
12	2.31	5.34	2.34	5.48	4.65	10.81
13	2.34	5.48	2.34	5.48	4.68	10.95
14	2.32	5.38	2.31	5.34	4.63	10.72
15	2.34	5.48	2.34	5.48	4.68	10.95
16	2.33	5.43	2.32	5.38	4.65	10.81
17	2.34	5.48	2.32	5.38	4.66	10.86
18	2.34	5.48	2.32	5.38	4.66	10.86
19	2.3	5.29	2.34	5.48	4.64	10.77
20	2.31	5.34	2.32	5.38	4.63	10.72
21	2.31	5.34	2.34	5.48	4.65	10.81
22	2.29	5.24	2.31	5.34	4.6	10.58
23	2.32	5.38	2.31	5.34	4.63	10.72
24	2.3	5.29	2.3	5.29	4.6	10.58
25	2.3	5.29	2.3	5.29	4.6	10.58
26	2.3	5.29	2.3	5.29	4.6	10.58
27	2.28	5.20	2.29	5.24	4.57	10.44
28	2.29	5.24	2.31	5.34	4.6	10.58
29	2.3	5.29	2.28	5.20	4.58	10.49
30	2.26	5.11	2.27	5.15	4.53	10.26
JML	69.21	159.68	69.31	160.14	138.52	319.82
$\bar{X}$	2.31		2.31		n	30
$\sum X^2$	159.68		160.14		x	2
$s^2$	0.00035		0.00036		N	30*2=60
s	0.01860		0.01884			



Gambar 3.2 Grafik Waktu Rendering tiap Frame Animation Location

Diketahui:

- $\bar{X}$  : Rata-rata x
- $\sum X^2$  : Mean X kuadrat
- s : Simpangan baku
- $s^2$  : 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 = 0.0036 dan varian terkecil = 0.0035 sehingga F hitung yang diperoleh:

$$F \text{ hitung} = \frac{0.0036}{0.0035} = \mathbf{1.03}$$

Selanjutnya F hitung akan dibandingkan dengan F table dengan df (degr freedom) pembilang m – 1 dan df penyebut N – 1. Demgan demikian dk pembilang = 2-1 = 1 dan dk penyebut = 60-2 = 58. Berdasarkan Tabel F, maka F tabel untuk 5% = 4.01 sedangkan F table untuk 1% = 7.09. Harga F hitung lebih kecil dari F table (**1.03<4.01**). Dengan demikian dapat disimpulkan bahwa varians data yang akan dianalisis homogen, sehingga perhitungan *anova* dapat dilanjutkan.

1. Jumlah Kuadrat Total (JK<sub>tot</sub>)

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

$$= 319.82 - \frac{138,52^2}{60}$$

$$JK_{tot} = \mathbf{0.0205}$$

2. Jumlah Kuadrat Antara (JK<sub>ant</sub>)

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

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

$$= \frac{69.81^2}{30} + \frac{70.01^2}{30} - \frac{138.52^2}{60}$$

$$= 159.67 + 160.13 - 319.80$$

$$= \mathbf{0.0002}$$

3. Jumlah Kuadrat dalam Kelompok (JK<sub>dal</sub>)

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

$$JK_{dal} = 0.0205 - 0.0002$$

$$JK_{dal} = \mathbf{0.0203}$$

4. Menghitung Mean Kuadrat Antar Kelompok (MK<sub>ant</sub>).

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

$$= \frac{0.0002}{2 - 1}$$

$$= \mathbf{0.0002}$$

5. Menghitung Mean Kuadrat Dalam Kelompok (MK<sub>dal</sub>).

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

$$= \frac{0.0203}{60 - 2}$$

$$= \mathbf{0.0004}$$

6. Menghitung F Hitung (F<sub>hit</sub>).

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

$$= \frac{0.0002}{0.0004}$$

$$= \mathbf{0.4756}$$

Table 3.4 Rencana Target Capaian Perhitungan

Sumber Validasi	df	Jumlah Kuadrat	MK	Fh	Ft	Kep
Total	60-1 = 59	0.0205				
Antar Kelompok	2-1=1	0.0002	0.0002	0.4756	5% 4.01 1% 7.09	Fh<Ft Jadi Ha diterima
Dalam Kelompok	60-2=58	0.0203	0.0004			

7. Menbandingkan harga F hitung dengan F table dengan df pembilang {m(2) – 1 = 1} dan df penyebut {N(60) – m(2)=58}, hasilnya adalah 4.01. F hitung (F<sub>hit</sub>) = 0.4756, jadi 0.4756>4.01 untuk probability 5% dan 0.4756>7.09 untuk probality 1% , Harga F hitung (F<sub>hit</sub>) lebih kecil dari F table (F<sub>tab</sub>) (Fh<Ft) tertentu. Jadi bila F hitung lebih kecil harga F table (Fh < Ft) maka Ho diterima dan Ha ditolak.

8. Jadi Tidak terdapat penurunan *time rendering* (lama waktu) terhadap penggunaan *Script Python 3D Animation LOCATION* dengan *Keyframe 3D Animation LOCATION*.

### 3.3 Evaluasi 3D Animation Scale

Penentuan hipotesis awal:

Ho : Tidak terdapat penurunan *time rendering* (lama waktu) terhadap penggunaan *Script Python 3D Animation SCALE* dengan *Keyframe 3D Animation SCALE*.

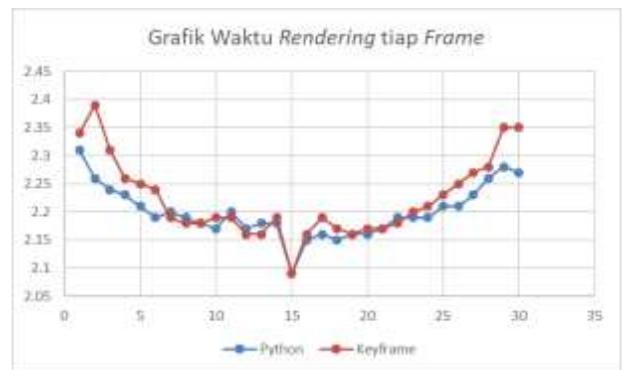
Ha : Terdapat penurunan *time rendering* (lama waktu) terhadap penggunaan *Script Python 3D Animation SCALE* dengan *Keyframe 3D Animation SCALE*

Pengujian Homogenitas Varians

Table 3.5 *Time Rendering tiap frame render* (X<sub>1</sub>,X<sub>2</sub>)

Frame	Time Rendering (s)				N	
	X <sub>1</sub>	X <sub>1</sub> <sup>2</sup>	X <sub>2</sub>	X <sub>2</sub> <sup>2</sup>	X <sub>tot</sub>	X <sub>tot</sub> <sup>2</sup>
1	2,31	5,34	2,34	5,48	4,65	10,81
2	2,26	5,11	2,39	5,71	4,65	10,82
3	2,24	5,02	2,31	5,34	4,55	10,35
4	2,23	4,97	2,26	5,11	4,49	10,08
5	2,21	4,88	2,25	5,06	4,46	9,95
6	2,19	4,80	2,24	5,02	4,43	9,81
7	2,2	4,84	2,19	4,80	4,39	9,64
8	2,19	4,80	2,18	4,75	4,37	9,55
9	2,18	4,75	2,18	4,75	4,36	9,50
10	2,17	4,71	2,19	4,80	4,36	9,51
11	2,2	4,84	2,19	4,80	4,39	9,64
12	2,17	4,71	2,16	4,67	4,33	9,37
13	2,18	4,75	2,16	4,67	4,34	9,42
14	2,18	4,75	2,19	4,80	4,37	9,55
15	2,09	4,37	2,09	4,37	4,18	8,74
16	2,15	4,62	2,16	4,67	4,31	9,29
17	2,16	4,67	2,19	4,80	4,35	9,46
18	2,15	4,62	2,17	4,71	4,32	9,33
19	2,16	4,67	2,16	4,67	4,32	9,33
20	2,16	4,67	2,17	4,71	4,33	9,37
21	2,17	4,71	2,17	4,71	4,34	9,42
22	2,19	4,80	2,18	4,75	4,37	9,55
23	2,19	4,80	2,2	4,84	4,39	9,64
24	2,19	4,80	2,21	4,88	4,4	9,68
25	2,21	4,88	2,23	4,97	4,44	9,86
26	2,21	4,88	2,25	5,06	4,46	9,95

Frame	Time Rendering (s)				N	
	X <sub>1</sub>	X <sub>1</sub> <sup>2</sup>	X <sub>2</sub>	X <sub>2</sub> <sup>2</sup>	X <sub>tot</sub>	X <sub>tot</sub> <sup>2</sup>
27	2,23	4,97	2,27	5,15	4,5	10,13
28	2,26	5,11	2,28	5,20	4,54	10,31
29	2,28	5,20	2,35	5,52	4,63	10,72
30	2,27	5,15	2,35	5,52	4,62	10,68
JML	65,98	145,17	66,66	148,26	132,64	293,43
$\bar{X}$	2,20		2,22		n	30
$\sum X^2$	145,17		148,26		x	2
s <sup>2</sup>	0,00209		0,00495		N	30*2=60
s	0,05		0,07			



Gambar 3.3 *Grafik Waktu Rendering tiap Frame Animation Scale*

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.

$$F \text{ hitung} = \frac{\text{Varian terbesar}}{\text{Varian terkecil}}$$

Dapat dilihat bahwa varian terbesar = 0.00495 dan varian terkecil = 0.00209 sehingga F hitung yang diperoleh:

$$F \text{ hitung} = \frac{0.00495}{0.00209} = 2.37$$

Selanjutnya F hitung akan dibandingkan dengan F table dengan df (degr freedom) pembilang m – 1 dan df penyebut N – 1. Dengan demikian dk pembilang = 2-1 = 1 dan dk penyebut = 60-2 = 58. Berdasarkan Tabel F, maka F tabel untuk 5% = 4.01 sedangkan F table untuk 1% = 7.09. Harga F

hitung lebih kecil dari F table ( $2.37 < 4.01$ ). Dengan demikian dapat disimpulkan bahwa varians data yang akan dianalisis homogen, sehingga perhitungan *anova* dapat dilanjutkan.

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

$$\begin{aligned} JK_{tot} &= \sum X_{tot}^2 - \frac{(\sum X_{tot})^2}{N} \\ &= 293,43 - \frac{132,64^2}{60} \\ JK_{tot} &= \mathbf{0.212} \end{aligned}$$

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

$$\begin{aligned} JK_{ant} &= \sum \frac{(\sum X_k)^2}{n_k} - \frac{(\sum X_{tot})^2}{N} \\ JK_{ant} &= \frac{(\sum X_1)^2}{n_1} + \frac{(\sum X_2)^2}{n_2} + \dots + \frac{(\sum X_m)^2}{n_m} - \frac{(\sum X_{tot})^2}{N} \\ &= \frac{65,98^2}{30} + \frac{66,66^2}{30} - \frac{132,64^2}{60} \\ &= 145.11 + 148.12 - 293,22 \\ &= \mathbf{0.008} \end{aligned}$$

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

$$\begin{aligned} JK_{dal} &= JK_{tot} - JK_{ant} \\ JK_{dal} &= 0.212 - 0.008 \\ JK_{dal} &= \mathbf{0.204} \end{aligned}$$

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

$$\begin{aligned} MK_{ant} &= \frac{JK_{ant}}{m-1} \\ &= \frac{0.008}{2-1} \\ &= \mathbf{0.008} \end{aligned}$$

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

$$\begin{aligned} MK_{dal} &= \frac{JK_{dal}}{N-m} \\ &= \frac{0.204}{60-2} \\ &= \mathbf{0.004} \end{aligned}$$

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

$$\begin{aligned} F_{hitung} &= \frac{MK_{ant}}{MK_{dal}} \\ &= \frac{0.008}{0.004} \\ &= \mathbf{2.188} \end{aligned}$$

Table 3.6 Rencana Target Capaian Perhitungan

Sumber Validasi	df	Jumlah Kuadrat	MK	Fh	Ft	Kep
Total	60-1 = 59	0.212				
Antar Kelompok	2-1=1	0.008	0.008	2.188	5% 4.01 1% 7.09	Fh<Ft Jadi Ha diterima
Dalam Kelompok	60-2=58	0.0204	0.004			

7. Menbandingkan harga F hitung dengan F table dengan df pembilang  $\{m(2) - 1 = 1\}$  dan df penyebut  $\{N(60) - m(2)=58\}$ , hasilnya adalah 4,01. F hitung ( $F_{hit}$ ) = 2.188, jadi  $2.188 < 4,01$  untuk probability 5% dan  $2.188 < 7,09$  untuk probality 1% , Harga F hitung ( $F_{hit}$ ) lebih kecil dari F table ( $F_{tab}$ ) ( $F_h < F_t$ ) tertentu. Jadi bila F hitung lebih kecil harga F table ( $F_h < F_t$ ) maka  $H_0$  diterima dan  $H_a$  ditolak.

8. Jadi **Tidak terdapat penurunan time rendering (lama waktu) terhadap penggunaan Script Python 3D Animation SCALE dengan Keyframe 3D Animation SCALE.**

#### 4 KESIMPULAN

Dari hasil yang telah dicapai terkait dengan komparasi teknik pembuatan *3d animation rotation, location* dan *scale* dengan *script python* dan *keyframe propertie* apakah akan mempengaruhi lamanya waktu *rendering* dengan menggunakan teknik statistik *Anova one way*. Bahwa hasil Evaluasi *3d animation rotation, location* dan *scale*, ternyata semua pengujian dapat dihitung dengan menggunakan teknik *Anova* yaitu *3d animation rotation, location* dan *scale* karena nilai F hitung lebih besar dari F table. Hasil pengujian *3d animation rotation* dengan Hipotesis Alternatif ditolak. Karena nilai F hitung lebih kecil dibandingkan dengan F table ( $F_h < F_t$ ) dengan nilai F hitung ( $F_{hit}$ ) = 0.145, jadi  $0.145 < 4.01$  untuk probability 5% dan  $0.145 < 7.09$  untuk probality 1% , Harga F hitung ( $F_{hit}$ ) lebih kecil dari F table ( $F_{tab}$ ) ( $F_h < F_t$ ) tertentu. Jadi bila F hitung lebih kecil harga F table ( $F_h < F_t$ ) maka  $H_0$  diterima dan  $H_a$  ditolak. Jadi **Tidak terdapat penurunan time rendering (lama waktu) terhadap penggunaan Script Python 3D Animation ROTATION dengan Keyframe 3D Animation ROTATION.**

Sedangkan pada Hasil pengujian *3d animation location* dengan Hipotesis Alternatif ditolak. Karena nilai  $F$  hitung lebih kecil dibandingkan dengan  $F$  table ( $F_h < F_t$ ) dengan nilai  $F$  hitung ( $F_{hit}$ ) = 4756, jadi  $0.4756 > 4.01$  untuk probability 5% dan  $0.4756 > 7.09$  untuk probality 1% , Harga  $F$  hitung ( $F_{hit}$ ) lebih kecil dari  $F$  table ( $F_{tab}$ ) ( $F_h < F_t$ ) tertentu. Jadi bila  $F$  hitung lebih kecil harga  $F$  table ( $F_h < F_t$ ) maka  $H_0$  diterima dan  $H_a$  ditolak. Jadi **Tidak terdapat penurunan *time rendering* (lama waktu) terhadap penggunaan Script Python 3D Animation LOCATION dengan Keyframe 3D Animation LOCATION.**

Sedangkan pada Hasil pengujian *3d animation rotation* dengan Hipotesis Alternatif ditolak. Karena nilai  $F$  hitung lebih kecil dibandingkan dengan  $F$  table ( $F_h < F_t$ ) dengan nilai  $F$  hitung ( $F_{hit}$ ) = 2.188, jadi  $2.188 < 4,01$  untuk probability 5% dan  $2.188 < 7,09$  untuk probality 1% , Harga  $F$  hitung ( $F_{hit}$ ) lebih kecil dari  $F$  table ( $F_{tab}$ ) ( $F_h < F_t$ ) tertentu. Jadi bila  $F$  hitung lebih kecil harga  $F$  table ( $F_h < F_t$ ) maka  $H_0$  diterima dan  $H_a$  ditolak. Jadi **Tidak terdapat penurunan *time rendering* (lama waktu) terhadap penggunaan Script Python 3D Animation SCALE dengan Keyframe 3D Animation SCALE.**

Hasil Evaluasi *3d animation rotation, location* dan *scale*, ternyata pengujian yang dapat dihitung dengan menggunakan teknik *Anova* adalah *3d animation rotation, locaation* dan *scale* karena varians bersifat homogen, karena nila  $F$  hitung lebih kecil disbanding  $F$  table. Kemudian untuk pengujian *3d animation rotation, location* dan *scale*  $H_0$  dapat diterima, karena nilai  $F$  hitung lebih kecil dari  $F$  table dan  $H_a$  ditolak. Hal ini membuktikan bahwa dengan menggunakan Teknik *Anova*, **Tidak terdapat penurunan *time rendering* (lama waktu) terdapat penggunaan script python dengan keyframe**, hal ini dikarenakan selisih perbedaan antara penggunaan *script python* dan *keyframe* tidak signifikan.

#### DAFTAR PUSTAKA

Blender. n.d. *About* . Blender.org. Accessed Mei 9, 2017. <https://www.blender.org/about/>.

Bokoro, P, and K Malandala. 2017. "Condition assessment of SF6 circuit breakers using analysis of variance (ANOVA)."

*AFRICON, 2017 IEEE*. Cape Town, South Africa: IEEE. 1374-1377.

Chaosangket, Nathaporn, Pruk Sasithong, Sanika K. Wijayasekara, Widhyakorn Asdornwised, Lunchakorn Wuttisittikulij, Pisit Vanichchanunt, and Muhammad Saadi. 2018. "A simulation tool for vertical transportation systems using python." *2018 5th International Conference on Business and Industrial Research (ICBIR)*. Bangkok: IEEE.

Foundation, Blender. 2018. *Blender Reference Manual us: Intruduction Scripting & Extending Blender*. Blender Manual. Maret 22. Accessed Mei 17, 2019. <https://docs.blender.org/manual/en/latest/advanced/scripting/introduction.html>.

Kramer, Jan-Peter, Michael Hennings, Joel Brandt, and Jan Borchers. 2016. "An Empirical Study of Programming Paradigms for Animation." *2016 IEEE/ACM Cooperative and Human Aspects of Software Engineering (CHASE)*. Austin: IEEE.

Lysy C, Moleong,, M. Rumagit Arthur, and A. Sugiarto Brave. 2013. "Implementasi Cluster Computing Untuk Render Animasi." *E-journal Teknik Elektro dan Komputer* 2 (3): 1-8.

Pramono, Waris, M Suyanto , and Amir Fatah Sofyan. 2017. "PERBANDINGAN METODE FRAME BY FRAME DAN EXPRESSION DALAM PEMBUATAN ANIMASI DUA DIMENSI." *Prosiding SEMNASTEK Fakultas Teknik Universitas Muhammadiyah Jakarta*. Jakarta: Universitas Muhammadiyah Jakarta.

- Pranatio, Garry, and Raymondus Kosala. 2010. "A Comparative Study of Skeletal and Keyframe Animations in a Multiplayer Online Game." *2010 Second International Conference on Advances in Computing, Control, and Telecommunication Technologies*. Jakarta, Indonesia: IEEE.
- Rumagit, Arthur. 2013. "IMPLEMENTASI GRID COMPUTING UNTUK HIGH THROUGHPUT COMPUTING." *JURNAL TEKNIK INFORMATIKA UNIVERSITAS SAM RATULANGI* 2 (1).
- Sugiyono, Prof. Dr. 2017. *Statistik untuk Penelitian*. Bandung: Penerbit Alfabeta.
- V. Patil, Ganesh, and Santosh L. Deshpande. 2016. "Distributed rendering system for 3D animations with Blender." *2016 IEEE International Conference on Advances in Electronics, Communication and Computer Technology (ICAECCT)*. Pune: IEEE.
- Verma, Shanti, and Kalyani Patel. 2017. "Association between shopping habit and demographics of m-commerce user's in India using two way ANOVA." *Convergence in Technology (I2CT), 2017 2nd International Conference for*. Mumbai, India: IEEE. 38-43.