

TESIS

**ANALISIS DAN OPTIMASI RENDERING PADA AUTODESK MAYA
DENGAN MENGGUNAKAN UE4**



Disusun oleh:

Nama : Winarja
NIM : 19.52.1216
Konsentrasi : Animation

**PROGRAM STUDI S2 TEKNIK INFORMATIKA
PROGRAM PASCASARJANA UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA**

2021

TESIS

**ANALISIS DAN OPTIMASI RENDERING PADA AUTODESK MAYA
DENGAN MENGGUNAKAN UE4**

**ANALYSIS AND OPTIMIZATION OF RENDERING ON AUTODESK
MAYA USING UE4**

Diajukan untuk memenuhi salah satu syarat memperoleh derajat Magister



Disusun oleh:

Nama : Winarja
NIM : 19.52.1216
Konsentrasi : Anlmatlon

**PROGRAM STUDI S2 TEKNIK INFORMATIKA
PROGRAM PASCASARJANA UNIVERSITAS AMIKOM YOGYAKARTA
YOGYAKARTA**

2021

HALAMAN PENGESAHAN

**ANALISIS DAN OPTIMASI RENDERING PADA AUTODESK MAYA DENGAN
MENGUNAKAN UE4
ANALYSIS AND OPTIMIZATION OF RENDERING ON AUTODESK MAYA
USING UE4**

Dipersiapkan dan Disusun oleh

Winarja

19.52.1216

Telah Diujikan dan Dipertahankan dalam Sidang Ujian Tesis
Program Studi S2 Teknik Informatika
Program Pascasarjana Universitas AMIKOM Yogyakarta
pada hari Senin, 05 Juli 2021

Tesis ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Magister Komputer

Yogyakarta, Senin, 05 Juli 2021

Rektor

Prof. Dr. M. Suyanto, M.M.
NIK. 190302001

HALAMAN PERSETUJUAN

ANALISIS DAN OPTIMASI RENDERING PADA AUTODESK MAYA DENGAN MENGUNAKAN UE4

ANALYSIS AND OPTIMIZATION OF RENDERING ON AUTODESK MAYA USING UE4

Dipersiapkan dan Disusun oleh

Winarja

19.52.1216

Telah Diujikan dan Dipertahankan dalam Sidang Ujian Tesis
Program Studi S2 Teknik Informatika
Program Pascasarjana Universitas AMIKOM Yogyakarta
pada hari Senin, 05 Juli 2021

Pembimbing Utama

Prof. Dr. M. Suyanto, M.M.
NIK. 190302001

Pembimbing Pendamping

Asro Nasiri, Drs, M.Kom.
NIK. 190302152

Anggota Tim Penguji

Dr.Kusrini, M.Kom.
NIK. 190302106

Dr.Andi Sunyoto, M.Kom.
NIK. 190302052

Prof. Dr. M. Suyanto, M.M.
NIK. 190302001

Tesis ini telah diterima sebagai salah satu persyaratan
untuk memperoleh gelar Magister Komputer

Yogyakarta, Senin, 05 Juli 2021
Direktur Program Pascasarjana

Dr. Kusrini, M.Kom.
NIK. 190302106

HALAMAN PERNYATAAN KEASLIAN TESIS

HALAMAN PERNYATAAN KEASLIAN TESIS

Yang bertandatangan di bawah ini,

Nama mahasiswa : Winarja
NIM : 19.52.1216
Konsentrasi : Animation

Menyatakan bahwa Tesis dengan judul berikut:
ANALISIS DAN OPTIMASI RENDERING PADA AUTODESK MAYA DENGAN
MENGGUNAKAN UE4

Dosen Pembimbing Utama : Prof. Dr. M. Susanto, M.M.
Dosen Pembimbing Pendamping : Asro Nasri, Drs, M.Kom.

1. Karya tulis ini adalah benar-benar ASLI dan BELUM PERNAH diajukan untuk mendapatkan gelar akademik, baik di Universitas AMIKOM Yogyakarta maupun di Perguruan Tinggi lainnya
2. Karya tulis ini merupakan gagasan, rumusan dan penelitian SAYA sendiri, tanpa bantuan pihak lain kecuali arahan dari Tim Dosen Pembimbing
3. Dalam karya tulis ini tidak terdapat karya atau pendapat orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan disebutkan dalam Daftar Pustaka pada karya tulis ini
4. Perangkat lunak yang digunakan dalam penelitian ini sepenuhnya menjadi tanggung jawab SAYA, bukan tanggung jawab Universitas AMIKOM Yogyakarta
5. Pernyataan ini SAYA buat dengan sesungguhnya, apabila di kemudian hari terdapat penyimpangan dan ketidakbenaran dalam pernyataan ini, maka SAYA bersedia menerima SANKSI AKADEMIK dengan pencabutan gelar yang mudah diperoleh, serta sanksi lainnya sesuai dengan norma yang berlaku di Perguruan Tinggi

Yogyakarta, 05 Juli 2021
Yang Menyatakan,



Winarja

HALAMAN PERSEMBAHAN

Terima kasih atas kasih sayang kedua orang tua saya yang berlimpah dari mulai saya lahir hingga saya sudah sebesar ini dan terima kasih juga atas limpahan doa yang tak berkesudahan serta segala hal yang telah dilakukan dengan yang terbaik.

Terima kasih juga yang tak terhingga untuk para dosen pembimbing, Bapak/Ibu yang dengan sabar melayani saya selama kuliah. Terima kasih juga untuk semua pihak yang mendukung keberhasilan tesis saya yang tidak bisa saya sebutkan satu per satu.

Ucapan terima kasih ini saya persembahkan juga untuk seluruh teman-teman saya di Universitas AMIKOM Yogyakarta 2020/2021. Terima kasih untuk memori yang kita rajut setiap harinya, atas tawa yang setiap hari kita miliki, dan atas solidaritas yang luar biasa. Sehingga masa kuliah ini menjadi lebih berarti.

Saya menyadari bahwa hasil karya tesis ini masih jauh dari kata sempurna, tetapi saya harap isinya tetap memberi manfaat sebagai ilmu dan pengetahuan bagi para pembacanya.

HALAMAN MOTTO

- *Seorang guru yang baik, selalu berharap bahwa murid-muridnya bisa lebih baik darinya (Jack man)*
- *Semakin banyak ilmu yang kita pelajari maka kita akan dapat menilai betapa bodohnya kita.*
- *Teman sejati adalah yang datang mendekati disaat yang lain melangkah pergi.*
- *Jangan pernah menyerah selalu bangkit dari setiap kesalahan, kekalahan, dan kegagalan*
- *Ilmu tiada batasnya, ambilah yang kamu minati dan perdalamilah*



KATA PENGANTAR

Puji syukur alhamdulillah penyusun panjatkan kehadiran Allah SWT yang senantiasa melimpahkan rahmat serta hidayah kepada setiap hamba-Nya yang beriman dan selalu berusaha. Shalawat serta salam tak lupa penyusun curahkan kepada junjungan kita Nabi Besar Muhammad SAW yang telah memberikan suri teladan mulia dalam menuntun ummatnya sampai pada zaman modern ini.

Tesis ini disusun sebagai salah satu syarat kelulusan MTI AMIKOM Yogyakarta dan untuk memperoleh gelar Magister Komputer.

Dengan selesainya tesis ini, penyusun tidak lupa mengucapkan terima kasih atas dukungan dan bimbingannya kepada:

1. Prof. Dr. M. Suyanto, M.M. sebagai Rektor Universitas "AMIKOM" Yogyakarta.
2. Asro Nasiri, Drs, M.Kom. selaku Dosen pembimbing yang telah banyak memberikan arahan dan masukan yang sangat membantu dalam penyelesaian Tesis ini.
3. Segenap Dosen dan staf tenaga pengajar Akademik Universitas "AMIKOM" Yogyakarta.
4. Ibunda dan Istri tercinta yang selalu memberikan dukungan untuk kelancaran Tesis dengan do'a restu dan harapannya.
5. Sahabat seperjuanganku, semoga kerja sama kita ini tak pernah berakhir.
Tank's for all.

6. Teman-teman S2 TI angkatan 2019 yang telah memberi support dan bantuan tambahan materi bagi Tesis saya.
7. Semua pihak yang tidak dapat kami sebut satu persatu yang telah membantu baik dukungan moril, pikiran dan tenaga dalam penyelesaian Tesis ini.

Penulis menyadari bahwa Tesis ini masih jauh dari sempurna, meskipun demikian penulis berharap semoga Tesis ini bermanfaat bagi yang membacanya, dan penulis dengan senang hati akan menerima kritik dan saran yang membangun dari para pembaca.

Akhirnya penulis mengharapkan semoga hasil karya ini dapat berguna serta bermanfaat bagi perkembangan teknologi dan informasi pada khususnya, serta sebagai kajian bagi mahasiswa Universitas "AMIKOM" Yogyakarta lainnya dalam pengambilan Tesis.

Yogyakarta, 05 Juli 2021

Penulis

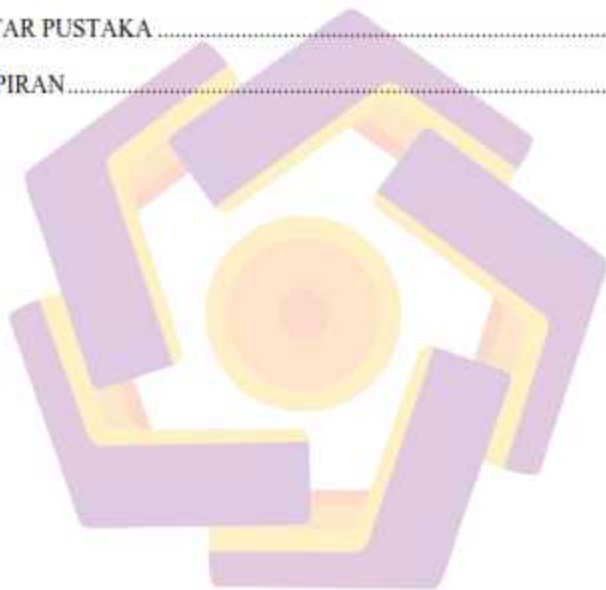
DAFTAR ISI

HALAMAN JUDUL.....	ii
HALAMAN PENGESAHAN.....	iii
HALAMAN PERSETUJUAN.....	iv
HALAMAN PERNYATAAN KEASLIAN TESIS.....	v
HALAMAN PERSEMBAHAN.....	vi
HALAMAN MOTTO.....	vii
KATA PENGANTAR.....	viii
DAFTAR ISI.....	x
DAFTAR TABEL.....	xiv
DAFTAR GAMBAR.....	xv
DAFTAR ISTILAH.....	xix
INTISARI.....	xx
<i>ABSTRACT</i>	xxi
BAB I PENDAHULUAN.....	1
1.1. Latar Belakang Masalah.....	1
1.2. Rumusan Masalah.....	3
1.3. Batasan Masalah.....	3
1.4. Tujuan Penelitian.....	4
1.5. Manfaat Penelitian.....	4
BAB II TINJAUAN PUSTAKA.....	5

2.1. Tinjauan Pustaka.....	5
2.2. Keaslian Penelitian.....	10
2.3. Landasan Teori.....	13
BAB III METODE PENELITIAN.....	19
3.1. Jenis, Sifat, dan Pendekatan Penelitian.....	19
3.2. Metode Pengumpulan Data.....	20
3.3. Metode Analisis Data.....	21
3.4. Alur Penelitian.....	22
BAB IV HASIL PENELITIAN DAN PEMBAHASAN.....	25
4.1. Asset 3D Model.....	25
4.1.1. Texture Shader Ground.....	25
4.1.2. Texture Shader Stone.....	26
4.1.3. Texture Shader Wood.....	26
4.1.4. Texture Shader Roof.....	27
4.1.5. Texture Shader Grass.....	27
4.1.6. Texture Shader Cloth Shape.....	28
4.1.7. Texture Shader Black Smith.....	29
4.1.8. Texture Shader Grass Small.....	29
4.1.9. Texture Shader Rope.....	30
4.2. Camera Turntable.....	31
4.3. Lighting Sky Dome.....	34
4.4. Lighting Directional.....	35

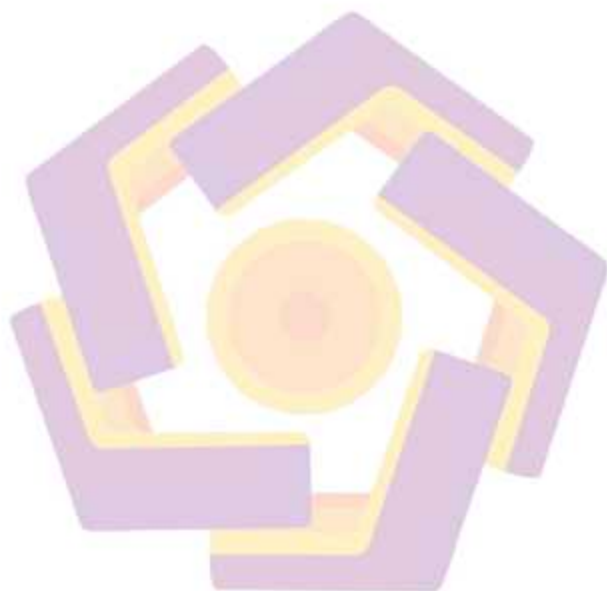
4.5. Rendering	36
4.6. Output Rendering.....	37
4.7. Setup UE4.....	39
4.8. Import Asset 3D model UE4.....	41
4.9. Import Asset Texture UE4.....	42
4.9.1. Assign UE4 Texture Shader Ground.....	43
4.9.2. Assign UE4 Texture Shader Stone.....	44
4.9.3. Assign UE4 Texture Shader Wood.....	45
4.9.4. Assign UE4 Texture Shader Roof.....	45
4.9.5. Assign UE4 Texture Shader Grass.....	46
4.9.6. Assign UE4 Texture Cloth Shape.....	46
4.9.7. Assign UE4 Texture Black Smith.....	47
4.9.8. Assign UE4 Texture Grass Small.....	48
4.9.9. Assign UE4 Texture Shader Rope.....	48
4.10. Lighting Sky Light UE4.....	49
4.11. Directional light UE4.....	50
4.12. Sequencer Camera.....	56
4.13. Build Lighting Quality.....	57
4.14. Post Process Volume.....	58
4.15. Render Movie.....	59
4.16. Sequence Render Image.....	60
4.17. Hasil perbandingan visual render Maya dan UE4.....	61

4.18. Hasil perbandingan waktu render Maya dan UE4.....	62
4.19. Pengujian render pada objek pohon.....	63
4.20. Pipeline Realtime Render	67
BAB V PENUTUP.....	71
5.1. Kesimpulan	71
5.2. Saran	71
DAFTAR PUSTAKA	73
LAMPIRAN.....	75



DAFTAR TABEL

Tabel 2. 1 Matriks literatur review dan posisi penelitian.....	10
Tabel 2. 2 Hasil uji render pada maya dan UE4	68



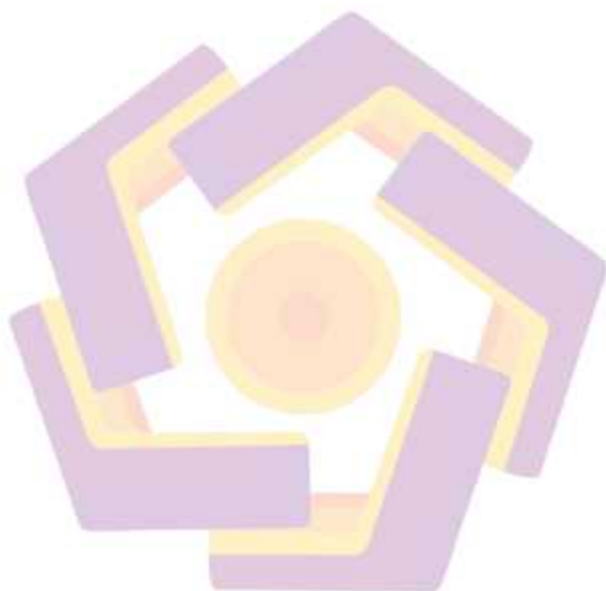
DAFTAR GAMBAR

Gambar 2. 1 Proses rendering	15
Gambar 2. 2 Ray Tracing	16
Gambar 2. 3 Realtime Render	17
Gambar 3. 1 Pipeline maya rendering	20
Gambar 3. 2 Alur Penelitian	24
Gambar 4. 1 Asset Medang Village datu House	25
Gambar 4. 2 Maya Shader Graound	26
Gambar 4. 3 Maya shader stone	26
Gambar 4. 4 Maya shader wood	27
Gambar 4. 5 Maya shader roof	27
Gambar 4. 6 Maya shader grass	28
Gambar 4. 7 Maya Cloth shape	28
Gambar 4. 8 Maya Black Smith	29
Gambar 4. 9 Maya Grass Small	30
Gambar 4. 10 Maya shader rope	30
Gambar 4. 11 Maya posisi camera	31
Gambar 4. 12 Motion Path camera	32
Gambar 4. 13 Frame rate	33
Gambar 4. 14 Maya camera output	34
Gambar 4. 15 Maya Sky dome	35
Gambar 4. 16 Maya Light Directional	35
Gambar 4. 17 Maya render seting	36

Gambar 4. 18 Maya Render	37
Gambar 4. 19 Maya Render squence	38
Gambar 4. 20 Waktu Maya Render squence	38
Gambar 4. 21 New project UE4	39
Gambar 4. 22 Template project UE4	40
Gambar 4. 23 Location project UE4	40
Gambar 4. 24 Imoprt 3d Asset UE4	41
Gambar 4. 25 UE4 Set Location Asset	42
Gambar 4. 26 Import teksture Asset UE4	43
Gambar 4. 27 UE4 Default material	43
Gambar 4. 28 UE4 Ground material	44
Gambar 4. 29 UE4 Stone material	44
Gambar 4. 30 UE4 Wood material	45
Gambar 4. 31 UE4 Roof material	45
Gambar 4. 32 UE4 Grass material	46
Gambar 4. 33 UE4 Cloth Shape material	47
Gambar 4. 34 UE4 Black Smith material	47
Gambar 4. 35 UE4 Grass Small material	48
Gambar 4. 36 UE4 Rope material	49
Gambar 4. 37 Object Sky light	49
Gambar 4. 38 Parameter Sky light	50
Gambar 4. 39 Object Directional light	51
Gambar 4. 40 Default Directional light	51

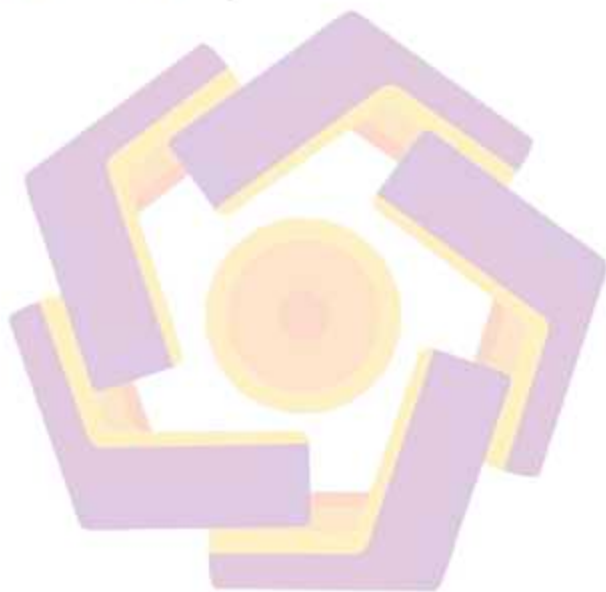
Gambar 4. 41 Directional light	52
Gambar 4. 42 Shadow Directtional light.....	53
Gambar 4. 43 Distance Shadow 100-800.....	54
Gambar 4. 44 Distance Shadow 900-20000.....	55
Gambar 4. 45 Focal length Camera UE4.	56
Gambar 4. 46 Squence Camera UE4.	57
Gambar 4. 47 Lighting Quality.....	57
Gambar 4. 48 Output Log.....	58
Gambar 4. 49 Post Process Volume.....	59
Gambar 4. 50 Render movie.....	60
Gambar 4. 51 Hasil Render squence.....	61
Gambar 4. 52 Hasil visual Render Maya dan UE4.....	62
Gambar 4. 53 Hasil Analisis waktu Render Maya dan UE4.....	62
Gambar 4. 54 Hasil visual Render big tree01.....	63
Gambar 4. 55 Hasil Analisis waktu Render big tree01.....	64
Gambar 4. 56 Hasil visual Render big tree02.....	64
Gambar 4. 57 Hasil Analisis waktu Render big tree02.....	65
Gambar 4. 58 Hasil visual Render big tree03.....	65
Gambar 4. 59 Hasil Analisis waktu Render big tree03.....	66
Gambar 4. 60 Hasil visual Render big tree04.....	66
Gambar 4. 61 Hasil Analisis waktu Render big tree04.....	67
Gambar 4. 62 Render noise.....	67
Gambar 4. 63 Perbandingan render noise	68

Gambar 4. 64 Render approval	69
Gambar 4. 65 Pipeline Realtime render	70



DAFTAR ISTILAH

- 3D : Tiga Dimensi
FPS : Frame Per Second
UE4 : Unreal Engine 4
VR : Virtual Reality



INTISARI

Proses rendering pada pembuatan film 3D animasi merupakan pekerjaan yang sangat memakan banyak waktu dikarenakan banyaknya frame yang diperlukan untuk membuat film 3D animasi. Pada proses rendering satu frame dalam film 3D animasi biasanya membutuhkan waktu hingga beberapa jam dikarenakan proses kalkulasi render yang terdiri dari data model, data shader, data texture, dan data lighting dalam sebuah shot pada adegan 3D animasi.

Dengan menggunakan proses render secara frame per frame maka proses render akan memerlukan waktu yang lama sehingga penulis akan melakukan penelitian pada proses rendering dengan menggunakan software game engine pada Unreal Engine 4 sebagai kalkulasi render secara realtime. Dari data yang semula dilakukan proses render pada software maya dan selanjutnya akan dilakukan perubahan data dari file maya diubah menjadi FBX file sebagai data yang akan di proses pada software Unreal Engine 4 (UE4). Dengan penggunaan game engine sebagai hasil akhir dalam menampilkan hasil render secara realtime sangatlah membantu dalam mendapatkan kelancaran pada sebuah proses produksi pada film 3D animasi.

Kata kunci: animasi 3D , rendering, real time render, game engine

ABSTRACT

The rendering process in making animated 3D movies is a very time-consuming job due to the large number of frames needed to make animated 3D films. In the rendering process one frame in an animated 3D movie usually takes up to several hours because the rendering calculation process consists of model data, shader data, texture data, and lighting data in a shot on a 3D animated scene.

By using the rendering process in a frame per frame, the rendering process will require a long time so the writer will conduct research on the rendering process by using the game engine software on Unreal Engine 4 as a realtime rendering calculation. From the data that was originally done in the virtual software rendering process and then the data will be changed from the virtual file to be converted into FBX file as data that will be processed in the Unreal Engine 4 software (UE4). Using the game engine as the final result in displaying rendering results in realtime is very helpful in getting fluency in a production process in 3D animated films.

Keyword: 3D animation, rendering, real time rendering, game engine



BAB I

PENDAHULUAN

1.1. Latar Belakang Masalah

Rendering pada 3D animasi adalah pekerjaan yang sangat memakan waktu. Rendering merupakan proses mengubah model geometris 3D menjadi sebuah gambar grafik. Rendering animasi menjadi tugas yang sangat memakan waktu dikarenakan banyaknya frame yang diperlukan untuk membuat animasi. Pada proses rendering satu frame dalam animasi biasanya membutuhkan waktu beberapa jam. Animasi komputer merupakan sebuah proses pada serangkaian gambar dari hasil render per frame dan kemudian ditampilkan secara berurutan frame demi frame untuk biasa menghasilkan ilusi gambar berupa efek gerakan yang kontinu. (Chong, Sourin, and Levinski 2006) Image yang dihasilkan komputer untuk produksi film selalu memiliki persyaratan yang sangat khusus mengenai kualitas gambar dan set fitur sistem rendering yang digunakan. Karena cuplikan film sinematik sering diambil pada frame rate biasanya 24FPS. (Fascione et al. 2018)

Ada dua jenis metode yang utama yang digunakan untuk rendering yaitu rasterisasi (juga dikenal sebagai scanline atau rendering poligon) dan Ray Tracing. Scanline rendering adalah algoritma untuk penentuan permukaan yang terlihat dari dalam grafik komputer 3D, yang bekerja secara baris per baris atau piksel per piksel. Semua poligon yang akan dirender diurutkan terlebih dahulu oleh koordinat y atas di mana pertama kali muncul. Kemudian tiap-tiap baris atau scanline dari image dikomputasikan dengan menggunakan perpotongan antara scanline dengan

polygon. Proses pada scanline ini bergerak secara berurutan turun menuju kebawah. Sedangkan Ray Tracing adalah teknik rendering untuk menghasilkan gambar dengan menelusuri jalur cahaya sebagai piksel dalam bidang gambar dan mensimulasikan efek pertemuannya dengan objek virtual. Teknik ini dapat menghasilkan tingkat realis visual yang sangat tinggi, biasanya hasil rendering lebih lama dari pada metode rendering scanline biasa (Peddie n.d.)

Frame pada film 3D animasi yang kita lihat ketika menonton film hari ini biasanya dibuat menggunakan render secara frame per frame, yang dapat membutuhkan banyak waktu beberapa jam untuk menghasilkan per frame. Proses yang memakan waktu ini menghasilkan konten yang sulit, sehingga untuk mengatasi masalah ini dengan memanfaatkan grafik real-time sebagai solusi untuk menyelesaikan pekerjaan secara efisien. Sementara grafik real-time dapat digunakan sebagai pratinjau pada tingkat kualitas yang belum mencapai standar yang dibutuhkan pada film. Namun selama bertahun-tahun perkembangan yang dihasilkan oleh grafis real-time menjadi lebih halus dan bahkan sebagai pravisualisasi dapat menggunakan dengan cara (VR)Virtual Reality. Hal ini dapat memberikan lebih banyak konteks untuk menjadi lebih kreatif. Mereka juga akan menggambarkan beberapa tantangan mereka tentang bagaimana mereka meramalkan masa depan grafik real-time dalam dunia film.(Jeremias-Vila et al. 2018)

Penggunaan game engine sebagai hasil akhir dalam menampilkan hasil render secara realtime sangatlah membantu dalam mendapatkan kelancaran pada sebuah produksi film. Meskipun UE4 dibuat untuk produksi game akan tetapi

system pada UE4 ini memiliki system yang bagus untuk produksi film. Game engine yang semakin populer sampai saat ini dengan banyaknya jumlah pengembang yang saat ini menghasilkan banyak minat dari siswa dalam belajar dan menggunakan engine ini di universitas dengan cara yang berbeda. Karena mereka sangat fleksibel, mereka tidak hanya belajar desain game tetapi mereka dapat digunakan untuk memproduksi film pendek, serial TV, aplikasi Virtual Reality, Visualisasi Desain dan penelitian ilmiah. (Merkel, Berger, and Braunreuther 2019)

1.2. Rumusan Masalah

Berdasarkan dari latar belakang masalah yang ada maka ada beberapa hal yang akan di analisa pada proses realtime render pada scene environment sebagai pengujian proses render dan beberapa point masalah tersebut adalah:

- a. Adakah perbedaan produktifitas pada kinerja render secara frame per frame dengan kinerja render secara realtime.
- b. Adakah pengaruh signifikan model render secara realtime terhadap pipeline pada produksi film 3D animasi.

1.3. Batasan Masalah

Bagian ini memuat penjelasan tentang:

1. Objek 3D yang akan di teliti yaitu berupa Scene Setdress environment 3D asset pada PT.MSV Studio.
2. Mengingat banyaknya divisi pada produksi film 3D animasi maka peneliti membatasi penggunaan asset pada divisi setdres emvironment.
3. Pengujian penelitian ini dilakukan pada Autodesk maya dan Unreal engine 4 dengan menggunakan camera turntable 360⁰ sebanyak 200 frame.

1.4. Tujuan Penelitian

Adapun tujuan yang ingin dicapai dalam penelitian tentang rendering ini adalah sebagai berikut :

- a. Sebagai syarat kelulusan MTI AMIKOM
- b. Penelitian ini bertujuan untuk meningkatkan produktifitas rendering dalam produksi film 3D animasi.

1.5. Manfaat Penelitian

Adapun manfaat dari penelitian ini adalah sebagai berikut:

- a. Dengan penelitian ini diharapkan dapat mempersingkat waktu proses pada tahapan rendering dan dapat mengurangi anggaran biaya pada proses rendering sehingga produksi menjadi lebih cepat terselesaikan.
- b. Penelitian ini dapat memberikan hasil berupa bentuk pipeline dari software Autodesk Maya pada proses rendering untuk pembuatan film 3D animasi dengan menggunakan software game engine pada Unreal Engine4.

BAB II

TINJAUAN PUSTAKA

2.1. Tinjauan Pustaka

Penelitian yang dilakukan oleh Taehyun Rhee, Member, IEEE, Lohit Petikam, Benjamin Allen, and Andrew Chalmers (2017) ini menyajikan sistem imersif baru yang disebut MR360 yang memberikan pengalaman realitas mixed reality (MR) menggunakan video panorama konvensional low dynamic range (LDR) 360° (360-video) yang ditampilkan di head mounted display (HMDs). MR360 secara mulus menggabungkan objek virtual 3D menjadi video 360 langsung menggunakan input video panorama sebagai sumber pencahayaan untuk menerangi objek virtual. Image based lighting (IBL) dioptimalkan secara perseptual untuk memberikan hasil yang cepat dan dapat dipercaya dengan menggunakan video LDR 360 sebagai sumber pencahayaan. Wilayah lampu paling menonjol dalam video panorama masukan dideteksi untuk mengoptimalkan jumlah lampu yang digunakan untuk menghasilkan bayangan yang terlihat. Kemudian, area cahaya yang terdeteksi menyesuaikan penumbra bayangan untuk memberikan bayangan lembut yang realistis. MR360 memberikan ilusi interaksi dengan objek dalam video, yang sebenarnya adalah objek virtual 3D yang digabungkan dengan mulus ke latar belakang video 360. MR360 diimplementasikan dalam game engine komersial dan diuji menggunakan berbagai video 360. Karena pipeline MR360 ini tidak memerlukan prapenghitungan apa pun dan pipeline dapat menyatukan adegan

MR interaktif menggunakan streaming video 360 secara langsung sambil memberikan rendering performa tinggi secara realistis.

Salah satu metode yang menghadirkan rendering real-time kualitas sinematik dengan mengintegrasikan ray tracing pada Unreal Engine 4 yang dilakukan oleh Edward Lui, Ignacio Llamas, Juan Canada, Patrick Kelly (2019) ini dapat meningkatkan kinerja mutakhir dalam ray tracing GPU. Selain pelajaran yang dipetik dari integrasi ray tracing di UE4, fase eksperimental awal sangat penting untuk mengeksplorasi kemungkinan ray tracing secara realtime. Untuk memulai dengan mirror reflections dan hard shadows, dilanjutkan dengan menambahkan denoising pada glossy reflections dan area light shadows dari limited ray yang terbatas, dan selanjutnya dengan menambahkan ambient occlusion, diffuse global illumination, dan translucency.

Pada penelitian ini memaparkan beberapa penelitian yang sedang dilakukandi DesignLab, di dalam Universitas Federal Santa Catarina (UFSC) - Brasil, oleh André Salomão, Flávio Andaló, and Milton Luiz Horn Vieira (2019) dengan menggunakan game engine dan bagaimana proses render tersebut dapat membantu mendapatkan sesuatu hasil render dengan lebih cepat dan lebih banyak lagi. DesignLab adalah laboratorium penelitian multi disiplin yang bekerja dengan profesor, peneliti, mahasiswa sarjana dan pascasarjana, terutama dari program Desain dan Animasi. Game engine yang dipilih sebagai uji coba adalah UnrealEngine 4 yang di gunakan untuk mengembangkan penelitian akademis tentang berbagai topik, seperti sebagai desain game, proyek animasi 3D.

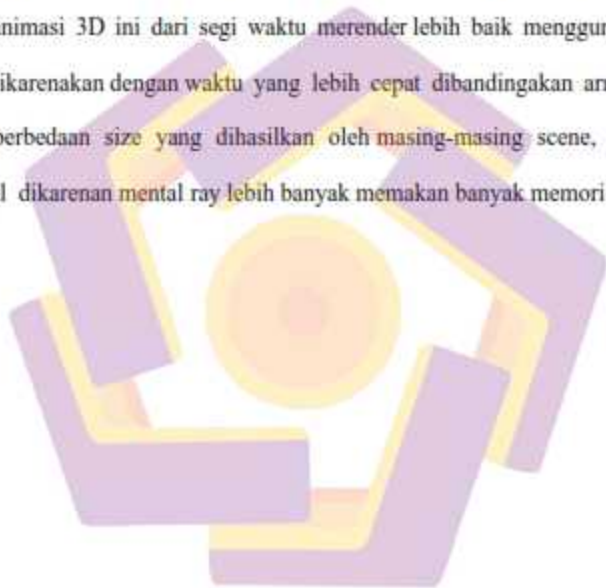
Rendering merupakan suatu tahapan untuk mengubah model menjadi bentuk video animasi 3D. Proses rendering memakan waktu cukup lama, dan memerlukan dukungan perangkat keras dengan spesifikasi tinggi. Sayangnya, komputer untuk kepentingan rendering dengan spesifikasi tinggi umumnya sangatlah mahal. Oleh karena itu, salah satu alternatif yang dapat ditempuh adalah dengan memanfaatkan teknologi grid computing yang dapat memanfaatkan beberapa komputer biasa yang dikonfigurasi dalam jaringan, sehingga secara bersama-sama dapat membantu menyelesaikan proses rendering dengan lebih cepat. Pada makalah ini, solusi rendering tersebut ditempuh dengan menggabungkan kerangka kerja berbasis teknologi grid computing, JPPF, yang dapat diintegrasikan dengan Blender 3D, yang merupakan salah satu perangkat lunak yang umum digunakan dalam pembuatan film animasi. Dalam penelitian ini, solusi tersebut juga dibandingkan hasilnya dengan fitur rendering berbasis jaringan yang memang telah ada pada Blender 3D, yang disebut Blender 3D Network Rendering, sebagai tolok ukur kinerja sistem yang dibangun. Hasil perbandingan menunjukkan bahwa solusi alternatif ini bekerja dengan kinerja dan fleksibilitas yang relatif lebih baik dan juga lebih mudah dioperasikan daripada fitur yang disediakan dalam Blender 3D. Penelitian ini yang dilakukan oleh Putrama, I Made Sparta, I Wayan Adi Darmawiguna, I Gede Mahendra Santyadiputra, Gede Saindra (2018).

Penelitian yang dilakukan oleh Sutarti, Ngatono, Noverla Seticaya,(2018) memberikan ide untuk menggunakan komputasi paralel berawal dari permasalahan waktu proses render, jika menggunakan satu komputer bisa memakan waktu yang

cukup lama dan menghasilkan hanya sebuah frame gambar dari proses, jika sebuah file animasi render yang diproses menggunakan proses komputer paralel atau dengan konsep jaringan cluster dapat menggunakan waktu seoptimal mungkin dan menghasilkan bagian-bagian dari proses rendering. Artinya setiap bagian-bagian dari proses rendering animasi tersebut diproses di komputer slave dan dicatat dari proses tersebut dan disimpan di komputer master. PC cluster yang diimplementasikan dalam kasus ini menggunakan middleware backburner. Dalam hal ini memungkinkan sebagai media pembelajaran ilmu baru dan menunjang kegiatan praktikum di Laboratorium Multimedia karena sifat rendering farm lebih efisien terhadap waktu kerjanya dan memungkinkan Laboratorium Multimedia Universitas Serang Raya dapat dijadikan tempat atau fasilitas rendering untuk berbagai kalangan, seperti pelajar, mahasiswa, dan yang lainnya. Setelah dilakukan pengujian maka didapatkan bahwa rendering dengan menggunakan PC cluster tiga slave lebih efisien dalam pemakaian processor masing-masing PC dan pemanfaatan waktu lebih baik karena dapat melakukan rendering secara bersamaan, dibandingkan dengan single computer.

Penelitian selanjutnya yang dilakukan oleh Rahmi Santy Nazir , Afdhol Dzikri , Oktavianto Gustin, (2017) dari hasil akhir dalam pembuatan animasi 3D adalah proses rendering dimana hasil rendering menentukan hasil akhir dari film yang dibuat. Animasi 3D tentunya akan melewati proses rendering yang merupakan proses sangat penting yang biasa digunakan oleh dunia film animasi. Setiap animasi menggunakan mesin rendering yang berbeda bergantung pada mesin rendering apa yang akan digunakan. Pada penelitian ini akan dihasilkan rendering waktu

rendering dari rendering menggunakan mental ray dan arnold pada film animasi 3D "Gara-gara HOAX". Selain itu, penelitian ini juga akan membandingkan hasil size per scene dari kedua teknik rendering yaitu mental ray dan arnold. Cara kerjanya adalah dengan menjalankan satu per satu rendering menggunakan Autodesk Maya, kemudian membandingkan hasil render dalam ukuran setiap scene dan lamanya waktu proses render dari setiap render. Dari kesimpulan peneliti ini bahwa pada film animasi 3D ini dari segi waktu merender lebih baik menggunakan mental ray, dikarenakan dengan waktu yang lebih cepat dibandingkan arnold dan dari segi perbedaan size yang dihasilkan oleh masing-masing scene, arnold lebih unggul dikarenakan mental ray lebih banyak memakan banyak memori file.



2.2. Keaslian Penelitian

Tabel 2. 1 Matriks literatur review dan posisi penelitian
Analisis Dan Optimasi Rendering Pada Autodesk Maya Dengan Menggunakan UE4

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
1	MR360: Mixed Reality Rendering for 360° Panoramic Videos	Tachyun Rhee, Member, IEEE, Lohit Petikam, Benjamin Allen, and Andrew Chalmers 2017	Menciptakan Virtual Reality (VR) Panorama 360 yang dikombinasikan dengan video.	Virtual Reality (VR) yang dihasilkan merupakan kombinasi dari objek 3d dan video panorama sebagai environment pencerahayan.	Virtual Reality (VR) ini dapat di gunakan sebagai pembelajaran siswa pada objek yang tidak bisa dijangkau oleh manusia ,seperti contoh pembelajaran tata surya di luar angkasa.	Penerapan pada penelitian ini di gunakan output sebagai Virtual Reality (VR) sedangkan pada peneliti akan di gunakan sebagai image sequence.
2	Cinematic Rendering in UE4 with Real-Time Ray Tracing and Denoising	Edward Lui, Ignacio Llamas, Juan Canada, Patrick Kelly 2019	Menciptakan cinematic rendering secara realtime dengan mengkombinasikan performa pada GPU	Pemanfaatan sebuah software game Engine untuk menciptakan cinematic terutama lebih pada reflection dan soft shadow dan ambient occlusion	Cinematic yang dilakukan peneliti masih kurang detail terutama untuk shader pada bahan metal, dikarenakan shader metal mempunyaikarakteristik yang berbeda-beda.	Pada perbandingan terdapat beberapa jenis nilai shadow yang akan di berikan kedalam objek yang berkaitan dengan hard shadow dan soft shadow

Tabel 2.1 (Lanjutan) Matriks literatur review dan posisi penelitian Analisis Dan Optimasi Rending Pada Autodesk Maya Dengan Menggunakan UE4

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
3	How Popular Game Engine Is Helping Improving Academic Research: The DesignLab Case	Andre Salomao, Flavio Andalo, and Milton Luiz Horn Vieira(2019)	mengembangkan penelitian akademik tentang berbagai topik, seperti desain game, proyck animasi 3D	Hasil pengujian didapat bahwa hasil Teknik pengembangan render dengan UE4 mengasilkan efisinsi	Pada pengujian ini belum adanya pipeline yang akan di gunakan	Untuk penelitian ini akan di hasilkan pipeline pada proses render dari import 3d model hingga pada proses aoutput image sequence.
4	Pengembangan Prototipe Sistem Network Rending Alternatif Berbasis JPPF	Putrama, I Made Sparta, I Wayan Adi Darmawiguna, I Gede Mahendra Santyadiputra, Gede Saindra, 2018	memanfaatkan teknologi grid computing yang dapat memanfaatkan beberapa komputer biasa yang dikonfigurasi dalam jaringan, sehingga secara bersama-sama dapat membantu menyelesaikan proses rendering dengan lebih cepat.	solusi rendering tersebut ditempuh dengan menggabungkan kerangka kerja berbasis teknologi grid computing, JPPF, yang dapat diintegrasikan dengan Blender 3D	Pada pengujian ini di perlukan beberapa komputer untuk melakukan proses rendering sehingga anggaran belanja untuk membeli perangkat computer menjadi semakin besar	Perbandingan dari penelitian ini tidak diperlukan konfigurasi dalam jaringan computer, sehingga lebih meminimalkan anggaran dalam proses rendering.

Tabel 2.1 (Lanjutan) Matriks literatur review dan posisi penelitian
 Analisis Dan Optimasi Rendering Pada Autodesk Maya Dengan Menggunakan UE4

No	Judul	Peneliti, Media Publikasi, dan Tahun	Tujuan Penelitian	Kesimpulan	Saran atau Kelemahan	Perbandingan
5	Implementasi Rendering Farm dengan Teknologi Cluster Computing Menggunakan Back Burner di Laboratorium Multimedia	Sutarti, Ngatono, Noverla, Seticaya, 2018	proses render menggunakan proses komputer paralel atau dengan konsep jaringan cluster	Proses rendering dengan menggunakan PC cluster yang diimplementasikan dalam kasus ini menggunakan middleware backburner untuk pemanfaatan waktu lebih baik karena dapat melakukan rendering secara bersamaan	Pada penelitian dengan PC cluster ini masih menggunakan system render secara frame per frame sehingga lamanya waktu proses render yang akan dihasilkan tergantung dari banyaknya computer yang di gunakan dalam proses render.	Untuk proses render ini akan mengubah dari proses render secara frame per frame yang kemudian di ubah dengan proses render secara realtime render pada sebuah game engine
6	Analisis Perbedaan Teknik Rendering Menggunakan Mental Ray dan Arnold Pada Film Animasi 3D "Gara-gara HOAX"	Rahmi Santy Nazir, Afdhol Dziki, Oktavianto Gustin, 2017	Pada proses analisa sistem ini bertujuan untuk menemukan permasalahan terhadap proses rendering dan menentukan kebutuhan apa yang sesuai yang terkait terhadap rendering.	penelitian yang digunakan adalah perbandingan render menggunakan mental ray dan Arnold. Proses analisa ini terdiri dari objek penelitian yang digunakan, analisis waktu dan size per scene	Untuk penelitian ini belum adanya penjelasan tentang proses pada konvesri dari setiap objek dari shader MentalRay yang kemudian dilakukan konversi ke shader Arnold.	Pada roses render secara realtime ini akan di jelaskan tahapan pada konversi objek dari FBX pada maya yang selanjutnya dilakukan proses penempatan pada shader game engine UE4

2.3. Landasan Teori

a. Autodesk Maya

Maya adalah program yang dibuat oleh Autodesk digunakan untuk membuat memodel, menggerakkan, dan membuat adegan 3D. Adegan 3D yang dibuat dengan Maya telah muncul di film, televisi, iklan, game, visualisasi produk, dan di Web. Dengan Maya, Anda dapat membuat dan menggerakkan adegan 3D Anda sendiri dan menjadikannya sebagai gambar diam atau sebagai rangkaian animasi. Beberapa versi Maya ada dan perbedaan di antara mereka terletak pada fitur yang termasuk dalam masing-masing. Versi komersial Maya mencakup semua yang Anda butuhkan untuk membuat adegan dan animasi 3D. Maya juga tersedia di Media dan Entertainment Collection, yang memasangkannya dengan produk Autodesk lainnya termasuk MotionBuilder, Mudbox dan 3ds Max. Versi Maya yang tersedia secara bebas bernama Maya LT. (Murdock 2019)

b. Unreal Engine4

Unreal Engine adalah sebuah aplikasi pengembangan game (game engine) yang dibuat oleh Epic Games, memulai debutnya pada 1998, dengan game bertema tembak-menembak orang-pertama (FPS). walau pada awalnya dibuat untuk game bertema itu saja, aplikasi ini mampu digunakan oleh pengembang dalam berbagai jenis genre game seperti; permainan bersembunyi (stealth), permainan pertempuran (fighting games), permainan perang multi pemain masif (MMORPGs), hingga beberapa permainan perang (RPGs). Dengan kode pemrograman yang ditulis dalam

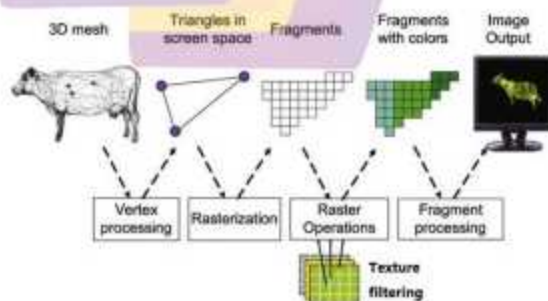
C++, aplikasi ini mampu membawakan kemudahan portabilitas yang tinggi serta merupakan aplikasi yang kerap digunakan berbagai pengembang permainan . diantara hasil output sebagai Virtual Reality(VR) yang dapat membantu dalam presentasi tentang keilmuan.(Rhee et al. 2017)

c. Surface Texture

Dalam seni visual, Tekstur adalah segala jenis detail permukaan yang membuat detail permukaan dengan tekstur yang terhubung pada objek material. Sejumlah tekstur yang dapat dipetakan pada objek untuk menerapkan tekstur ke atribut material objek. Atribut yang terhubung dengan tekstur menentukan bagaimana tekstur itu mempengaruhi hasil akhir pada rendering. Jika menghubungkan tekstur hitam dan putih ke atribut transparansi, maka menerapkan hasil transparansi pola menentukan bagian mana dari objek yang buram, semi-transparan, atau sepenuhnya transparan. Map specularity dengan memetakan tekstur ke atribut Specularity dari material objek, maka map specularity dapat memungkinkan mendeskripsikan bagaimana kilau yang akan muncul pada objek. Map refleksi dengan memetakan tekstur ke atribut Refleksi dari bahan objek, maka map refleksi yang memungkinkan akan menggambarkan bagaimana objek dapat mencerminkan di sekitarnya. Displacement maps memungkinkan untuk menambahkan dimensi sebenarnya ke permukaan pada waktu render, suatu proses yang dapat menghasilkan kebutuhan untuk membuat model yang kompleks(Morozov and Technology 2019).

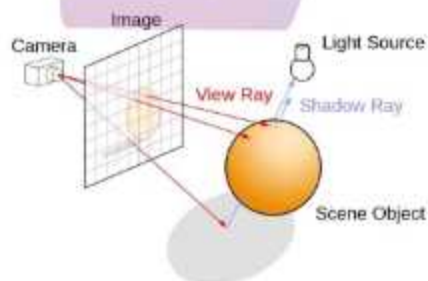
d. Rendering

Rendering pada 3D animasi adalah pekerjaan yang sangat memakan waktu. Rendering merupakan proses mengubah model geometris 3D menjadi sebuah gambar grafik seperti pada Gambar 2.1. Rendering animasi menjadi tugas yang sangat memakan waktu dikarenakan banyaknya frame yang diperlukan untuk membuat animasi. Pada proses rendering satu frame dalam animasi biasanya membutuhkan waktu beberapa jam. Animasi komputer merupakan sebuah proses pada serangkaian gambar dari hasil render per frame dan kemudian ditampilkan secara berurutan frame demi frame untuk biasa menghasilkan ilusi gambar berupa efek gerakan yang kontinu (Chong, Sourin, and Levinski 2006). Image yang dihasilkan komputer untuk produksi film selalu memiliki persyaratan yang sangat khusus mengenai kualitas gambar dan set fitur sistem rendering yang digunakan. Karena cuplikan film sinematik sering diambil pada frame rate biasanya 24FPS.(Fascione et al. 2018)



Gambar 2. 1 Proses rendering.

Ada dua jenis metode yang utama yang digunakan untuk rendering yaitu rasterisasi (juga dikenal sebagai scanline atau rendering poligon)(Francisco 2011) dan Ray Tracing(Liu et al. 2019). Scanline rendering adalah algoritma untuk penentuan permukaan yang terlihat dari dalam grafik komputer 3D, yang bekerja secara baris per baris atau piksel per piksel. Semua poligon yang akan dirender diurutkan terlebih dahulu oleh koordinat y atas di mana pertama kali muncul. Kemudian tiap-tiap baris atau scanline dari image dikomputasikan dengan menggunakan perpotongan antara scanline dengan polygon. Proses pada scanline ini bergerak secara berurutan turun menuju kebawah. Sedangkan Ray Tracing adalah teknik rendering untuk menghasilkan gambar dengan menelusuri jalur cahaya sebagai piksel dalam bidang gambar dan mensimulasikan efek pertemuannya dengan objek virtual seperti pada Gambar 2.2 Teknik ini dapat menghasilkan tingkat realis visual yang sangat tinggi, dan biasanya hasil rendering ini lebih lama dari pada metode rendering scanline biasa.



Gambar 2. 2 Ray Tracing

Frame pada film 3D animasi yang kita lihat ketika menonton film hari ini biasanya dibuat menggunakan render secara frame per frame, yang dapat membutuhkan banyak waktu beberapa jam untuk menghasilkan per frame. Proses yang memakan waktu ini menghasilkan konten yang sulit, sehingga untuk mengatasi masalah ini dengan memanfaatkan grafik real-time seperti pada Gambar 2.3 sebagai solusi untuk menyelesaikan pekerjaan secara efisien. Sementara grafik real-time (Larsson-ståhl and Larsson-ståhl 2020) dapat digunakan sebagai pratinjau pada tingkat kualitas yang belum mencapai standar yang dibutuhkan pada film. Namun selama bertahun-tahun perkembangan yang dihasilkan oleh grafis real-time menjadi lebih halus dan bahkan sebagai pra-visualisasi dapat menggunakan dengan cara Virtual Reality (Li, Yang, and Cao 2015). Hal ini dapat memberikan lebih banyak konteks untuk menjadi lebih kreatif. Mereka juga akan menggambarkan beberapa tantangan mereka tentang bagaimana mereka meramalkan masa depan grafik real-time dalam dunia film. (Jeremias-Vila et al. 2018).



Gambar 2. 3 Realtime Render.

Penggunaan game engine(Stadler and Hlavacs 2018) sebagai hasil akhir dalam menampilkan hasil render secara realtime sangatlah membantu dalam mendapatkan kelancaran pada sebuah produksi film. Meskipun UE4 dibuat untuk produksi game akan tetapi system pada UE4 ini memiliki system yang bagus untuk produksi film. Game engine yang semakin populer sampai saat ini dengan banyaknya jumlah pengembang yang saat ini menghasilkan banyak minat dari siswa dalam belajar dan menggunakan engine ini di universitas dengan cara yang berbeda. Karena mereka sangat fleksibel, mereka tidak hanya belajar desain game tetapi mereka dapat digunakan untuk memproduksi film pendek, serial TV, aplikasi Virtual Reality, Visualisasi Desain dan penelitian ilmiah.(Merkel, Berger, and Braunreuther 2019).



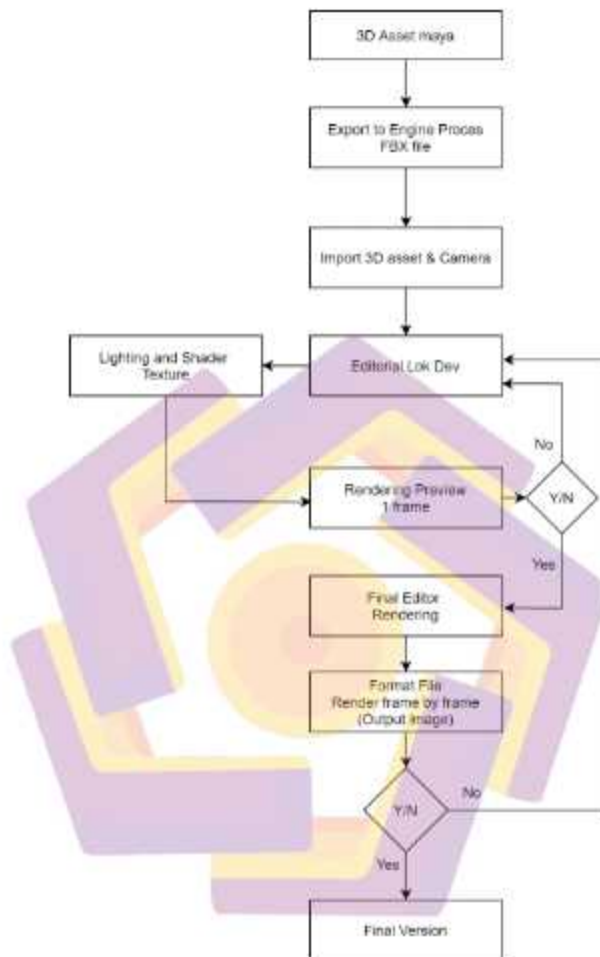
BAB III

METODE PENELITIAN

3.1. Jenis, Sifat, dan Pendekatan Penelitian

Tahapan dalam penelitian ini mulai dari pengumpulan data melalui study literature yang dilanjutkan dengan tahapan penelitian pada produksi animasi 3D pada divisi rendering. Dan selanjutnya akan dijadikan objek penelitian dalam hal ini pada projek film animasi 3D yang di produksi MSV Studio. Dari produksi animasi 3D ini mulai kita evaluasi kebutuhan dan kelemahan pada pipeline yang digunakan sebelumnya. Diantaranya set up asset 3D environment yang ada pada Renderes masih kurang efektif dalam prakteknya, karena memakan waktu yang cukup lama ketika pelaksanaan dalam proses rendering. Untuk itu penulis mencoba melakukan penelitian dengan metode realtime render untuk mendukung pipeline pada divisi rendering untuk melihat memungkinkan atau tidak dalam optimalisasi pengerjaan proses rendeering pada saat produksi film animasi 3D.

Pada tahapan perancangan, penulis mulai merancang pipeline rendering. Dimana dalam tahap ini penulis memperbaharui pipeline yang sudah ada dan selanjutnya masuk kedalam tahap implementasi, dimana pipeline rendering secara real-time dapat diimplementasikan dalam produksi animasi 3D pada divisi rendering.agian ini memuat jenis, sifat dan pendekatan penelitian yang digunakan beserta penjelasan secara garis besar mengenai penelitian ini sehingga dapat dikelompokkan dalam penelitian tersebut.



Gambar 3. 1 Pipeline maya rendering

3.2. Metode Pengumpulan Data

Untuk memperoleh data yang diperlukan maka peneliti menggunakan metode pengumpulan data sebagai berikut:

- a. Dalam hal ini penulis terjun langsung dalam produksi film animasi 3D Ajisaka. Penulis mengamati secara langsung proses, kendala dan permasalahan yang terjadi pada produksi film animasi 3D Ajisaka khususnya pada divisi rendering.
- b. Dalam hal ini penulis akan mengumpulkan beberapa objek asset 3D environment sebagai bahan uji pada proses rendering.
- c. Dengan melakukan wawancara mengenai kelebihan dan kekurangan pipeline yang digunakan. Dalam penelitian ini yang menjadi subjek wawancara adalah 3D artis rendering yang terlibat langsung dalam produksi film animasi 3D Ajisaka.

3.3. Metode Analisis Data

Analisis data yang dilakukan yaitu dengan menarik kesimpulan dari data yang telah di kumpulkan dari observasi dan wawancara terhadap artist rendering. Dan ntuk menganalisa data dari objek Scene Setdress environment 3D asset dengan pengujian render dengan camera turntable 200 frame pada software maya yang selajutnya diperoleh data dari 200 frame yaitu hasil waktu yang diperlukan untuk menyelesaikan proses render tersebut. Dan tahap selanjutnya Scene Setdress environment 3D asset di lakukan pengujian camera turntable 200 frame dengan menggunakan software UE4 yang selanjutya akan di peroleh hasil waktu untuk menyelesaikan pada proses rendering tersebut.

Data yang di hasilkan dalam penelitian ini adalah sebuah data image pada 200 frame dengan waktu pada proses render tersebut. Untuk pengecekan hasil dari rendering dengan UnrelEngine4 apakah pipeline sudah berjalan dengan baik atau

belum ketika di terapkan di divisi render. Dan berpengaruh tidaknya dengan timeline proses produksinya. Paramater yang digunakan untuk mengukur kualitas hasil render pada maya dan UE4 ini yaitu dengan persetujuan atau approval dari seorang produser pada perusahaan animasi. Seorang produser akan melihat hasil output render pada UE4 dengan mengacu pada image dari hasil render yang dilakukan pada maya, bilamana hasil image render pada UE4 telah sama mirip pada hasil image render pada maya maka hasil render UE4 sudah bisa disetujui dan dapat di proses pada tahap selanjutnya yaitu proses compositing.

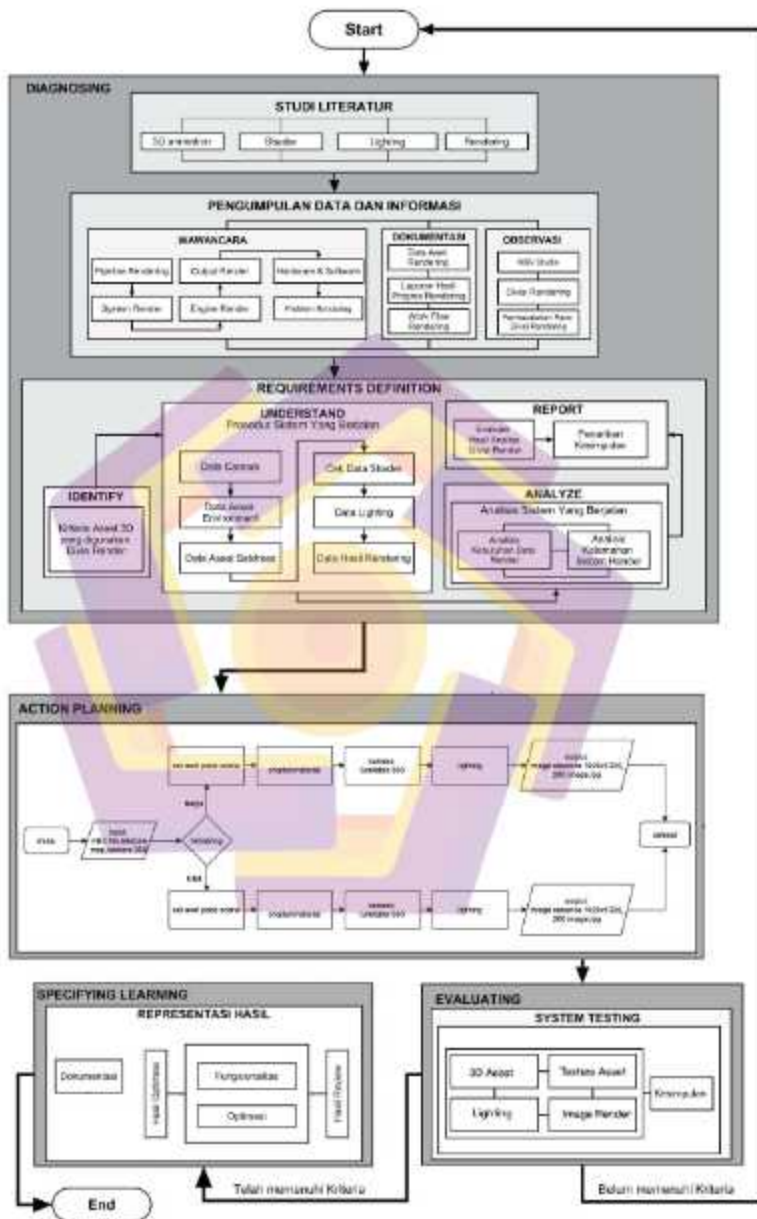
3.4. Alur Penelitian

Tahapan dalam penelitian ini mulai dari pengumpulan data melalui study literature yang dilanjutkan dengan tahapan penelitian pada produksi animasi 3D pada divisi rendering. Dan selanjutnya akan dijadikan objek penelitian dalam hal ini pada projek film animasi 3D yang di produksi MSV Studio. Dari produksi animasi 3D ini mulai kita evaluasi kebutuhan dan kelemahan pada pipeline yang digunakan sebelumnya. Diantaranya set up render yang masih kurang efisien dikarenakan proses render yang di lakukan secara frame per frame sehingga memakan waktu yang cukup lama ketika pelaksanaan dalam proses rendering. Untuk itu penulis mencoba melakukan penelitian dengan metode realtime render untuk mendukung pipeline(Lowe 2019) pada divisi rendering.

Metode pengumpulan data yang dilakukan dengan cara pengamatan langsung dan mencatat secara sistematis dari permasalahan yang akan diselidiki. Observasi digunakan untuk memperoleh data atau informasi dengan mengamati

secara langsung ditempat kejadian. Dalam hal ini penulis terjun langsung dalam produksi film animasi 3D Ajisaka. Penulis mengamati secara langsung proses, kendala dan permasalahan yang terjadi pada produksi film animasi 3D Ajisaka khususnya pada divisi rendering. Sebagai bahan penelitian ini data yang akan di uji yaitu model MedangVilagedatuHouse yang di ambil sebagai sampel dalam sebuah aset 3D environment.

Analisa data yang dilakukan pada Gambar 5 yaitu dengan merangkum kesimpulan dari data yang telah di kumpulkan dari observasi dan wawancara terhadap artist rendering. Dan untuk menganalisa data dari objek Scene Setdress environment 3D asset dengan pengujian render dengan camera turntable (Tamboli et al. 2018) sebanyak 200 frame pada software maya yang selanjutnya diperoleh data dari 200 frame yaitu hasil waktu yang diperlukan untuk menyelesaikan proses render tersebut. Dan tahap selanjutnya Scene Setdress environment 3D asset di lakukan pengujian camera turntable 200 frame dengan menggunakan software UE4 yang selanjutnya akan di peroleh hasil waktu untuk menyelesaikan pada proses rendering tersebut.



Gambar 3. 2 Alur Penelitian.

BAB IV

HASIL PENELITIAN DAN PEMBAHASAN

4.1. Asset 3D Model

Pada awal tahapan ini pengambilan data asset 3D model yang akan digunakan sebagai bahan uji pada proses Rendering. Objek yang akan di ambil sebagai bahan uji rendering yaitu 3D model asset dari MSV Studio dengan sebuah bangunan rumah yang bernama Medang Village datu House.



Gambar 4. 1 Asset Medang Village datu House

4.1.1. Texture Shader Ground

Setelah asset 3d model selesai di setup maka untuk memberikan warna atau tekstur pada objek harus di berikan material yang sesuai dengan konsep yaitu dengan memberikan tekstur pada ground dengan menempatkan data texture sesuai dengan tiga jenis tekstur yaitu color, specular dan normal.



Gambar 4. 2 Maya Shader Graound

4.1.2. Texture Shader Stone

Selanjutnya objek yang harus di berikan pada dinding rumah berupa tekstur batu dengan komposisi tiga tekstur meliputi teksture batu, specular batu dan normal map batu dengan penempatan pada shader batu.



Gambar 4. 3 Maya shader stone

4.1.3. Texture Shader Wood

Pada tahapan berikutnya yaitu dengan menempatkan texture kayu pada dinding rumah dan juga kerangka atap pada rumah dengan menempatkan tiga jenis tekstur wooden plank_TEX , wooden plank_SPC dan wooden pank_NMM yang kemudian di tempatkan pada objek dinding.



Gambar 4. 4 Maya shader wood.

4.1.4. Texture Shader Roof

Langkah berikutnya adalah dengan memberikan tekstur pada atap rumah dengan tiga jenis tesktur yaitu roof_TEX, roof_SPC dan roof_NMM yang masing-masing di tempatkan sesuai dengan shader Roof.



Gambar 4. 5 Maya shader roof

4.1.5. Texture Shader Grass

Selanjutnya yaitu penempatan tekstore pada shader Grass yang terdiri dari jenis grass_TEX, grass_SPC, grass_NMM dan grass_MSK yang kemudian dapat di tempatkan pada objek rumput yang sesuai dengan shader grass tersebut.



Gambar 4. 6 Maya shader grass.

4.1.6. Texture Shader Cloth Shape

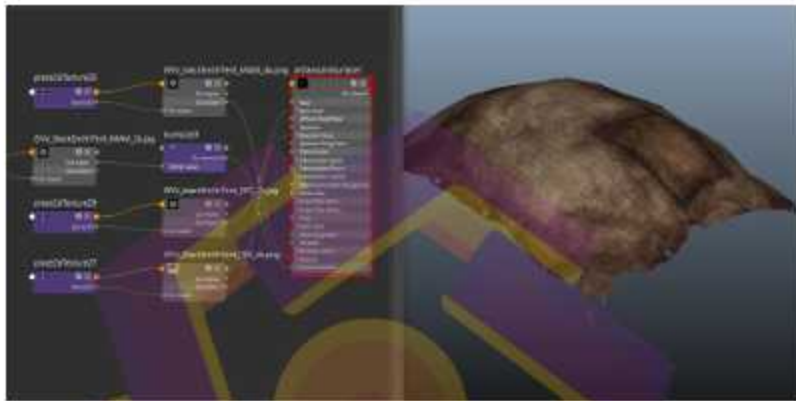
Pada model medang village datu house terdapat sebuah taing di samping depan rumah dengan sebuah logo atau simbol yang terbuat dari kain dengan menggunakan tekstur opacity atau masking sebagai pemotong dari tekstur.



Gambar 4. 7 Maya Cloth shape.

4.1.7. Texture Shader Black Smith

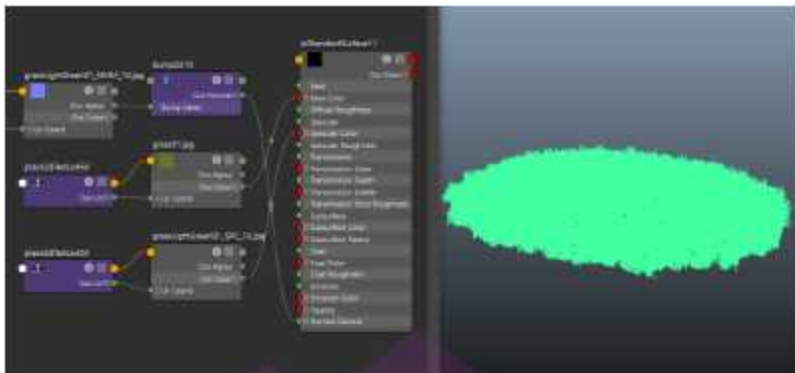
Terdapat sebuah kain besar yang menutupi pada bagian atap rumah medang village datu house yang berbahankan kain goni dengan tekstur color, specular, normal map dan masking sebagai pemotong tekstur



Gambar 4. 8 Maya Black Smith.

4.1.8. Texture Shader Grass Small

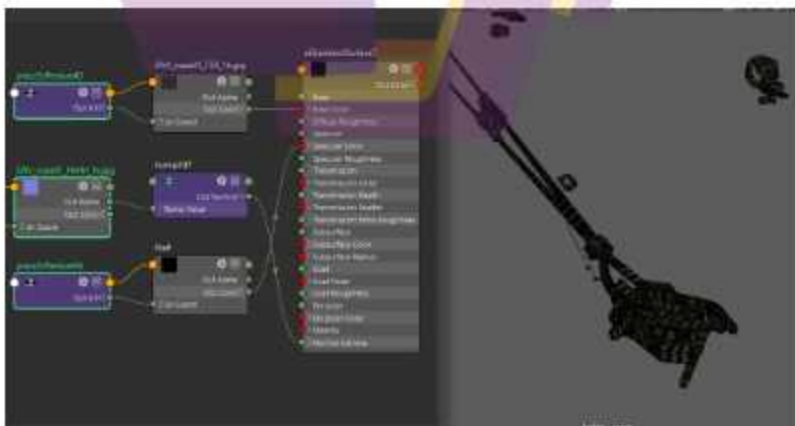
Dari alas permukaan atau ground pada samping rumah terdapat beberapa tanaman rumput disekitarnya yang didalam shader juga terdapat tekstur color, specular dan normal map. Pada bagian ini terdapat beberapa gerobol dari objek rumput yang hanya menggunakan satu jenis shader yang sama.



Gambar 4. 9 Maya Grass Small.

4.1.9. Texture Shader Rope

Selanjutnya pada objek rumah medang village datu house juga terdapat model tali sebagai pengikat dari rumah tersebut dan jenis dari tekstur yang ada pada shader tersebut juga sama seperti yang lainya yaitu menggunakan color, specular dan normal map.



Gambar 4. 10 Maya shader rope.

4.2. Camera Turntable

Setelah semua shader terselesaikan dengan penempatan pada tekstur masing-masing maka tahapan selanjutnya yaitu dengan pembuatan camera yang selanjutnya akan di animasikan secara turntable dengan memutari objek medan village datu house sebanyak 200 frame atau 360 derajat dengan tujuan supaya dari semua sudut pandangan depan ,samping dan belakan dapat diambil hasil render dari camera tersebut dan dapat dilihat sebagai hasil pada saat pengujian rendering.

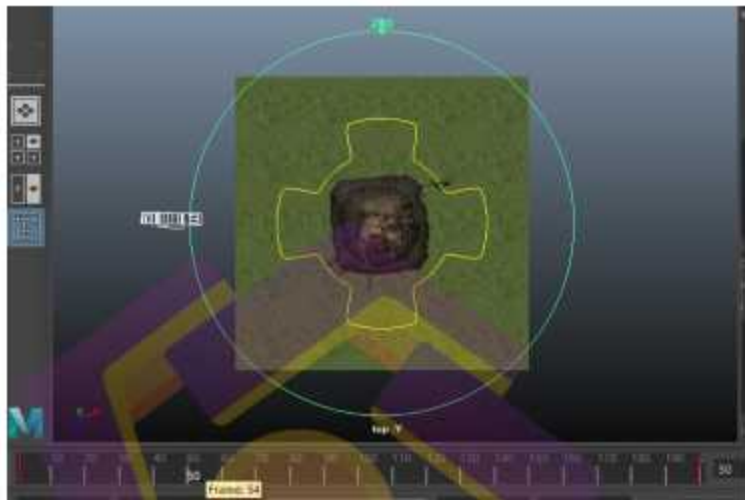
Pada tahap pembautan camera turntable yang dilakukan yang pertama yaitu dengan membuat camera standar dengan nilai focal length 35, setelah selesai pembuatan camera langkah selanjutnya yaitu pembuatan jalur melingkar 360 derajat dengan memosisikan pada posisi tengah dari ketinggian diantara batas paling atas dari objek dan batas paling bawah dari objek.



Gambar 4. 11 Maya posisi camera.

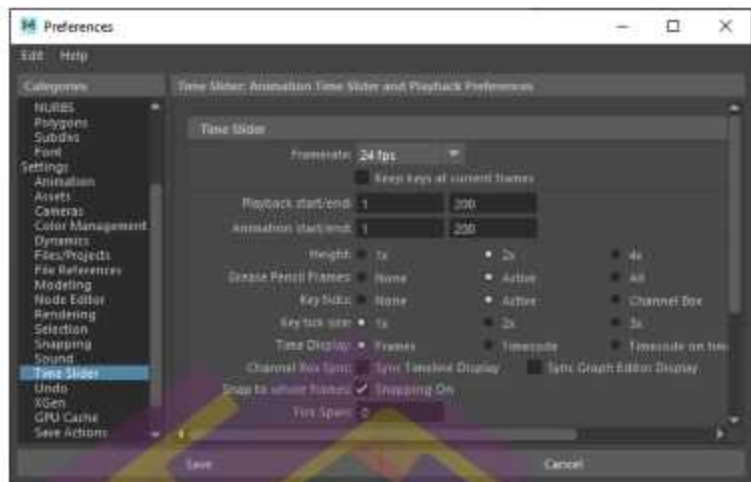
Camera yang sudah sesuai dengan posisi pengambilan gambar maka untuk menjalankan animasi 360 derajat disini tidak dilakukan secara manual akan tetapi dilakukan secara otomatis dengan menggunakan jalur lingkaran sehingga camera

berputar 360 derajat dengan mengikuti jalur yang di buat dengan menggunakan nurbs circle yang di constainkan dengan menggunakan motion path.



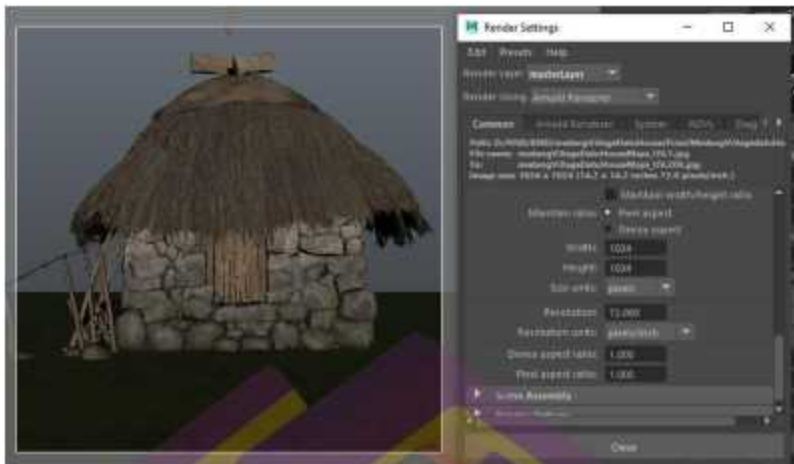
Gambar 4. 12 Motion Path camera

Untuk menentukan frame rate atau banyaknya gambar yang akan ditampilkan dalam perdetik maka diperlukan pengaturan pilihan yang sesuai dengan kebutuhan yang akan kita gunakan, dalam pengujian ini akan di gunakan dengan kebutuhan yang sesuai dengan standart pada film animasi yaitu menggunakan frame rate sebanyak 24 frame per detik dan banyaknya waktu yang akan di gunakan yaitu selama 8 detik atau sebanyak 200 frame sebagai pengujian pada render.



Gambar 4. 13 Frame rate.

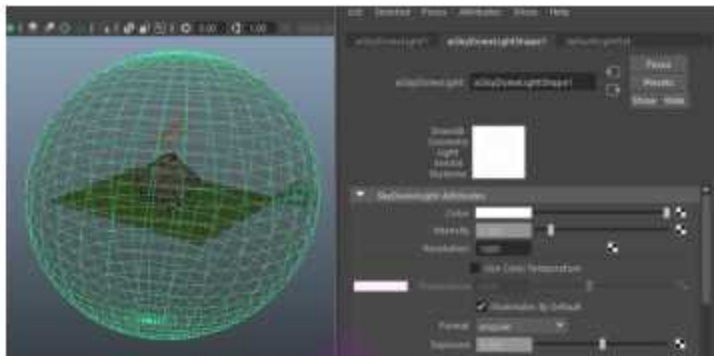
Selanjutnya untuk hasil dari output pada kamera juga harus di tentukan sebagai hasil ukuran image dari hasil pada proses render dan sebagai output dari camera disini dengan menggunakan ukuran 1024 x 1024 pixel. Pengaturan output pada camera disini tidak diatur pada atribut editor yang terdapat pada camera akan tetapi dilakukan pengaturan outputnya melalui render seting dengan memberikan nilai width 1024 dan height 1024.



Gambar 4. 14 Maya camera output.

4.3. Lighting Sky Dome

Untuk langkah selanjutnya setelah penempatan camera selesai yaitu dengan memberikan pencahayaan yang berfungsi sebagai penerangan pada area objek medang village datu house . Terdapat dua jenis pencahayaan yang akan digunakan disini yaitu dengan memberika sky dome sebagai penerangan global dan directional lighting sebagai lapu atau penerangan utamanya. Sky dome yang berbentuk bola yang mengelilingi dari berbagai sudut dengan fungsi memberikan penerangan secara global.



Gambar 4. 15 Maya Sky dome.

4.4. Lighting Directional

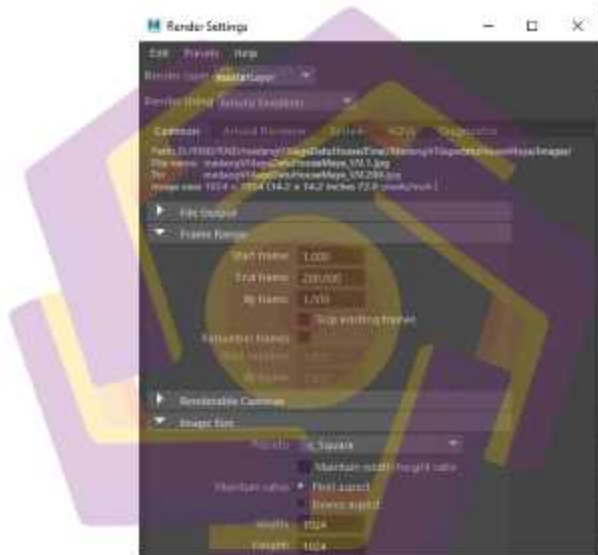
Pencahayaan yang kedua yaitu dengan menempatkan jenis directional dengan tujuan untuk membarikan pencahayaan seperti layaknya sinar matahari yang menghasilkan pancaran intensitas cahaya dan bayangan yang memberikan kesan penerangan secara alami.



Gambar 4. 16 Maya Light Directional.

4.5. Rendering

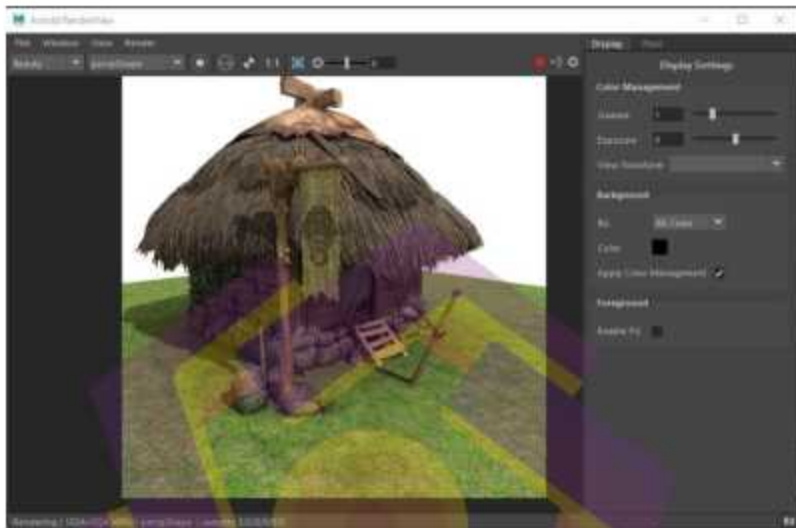
Dari beberapa tahap yang sudah terselesaikan pada penempatan model, tekstur, camera dan lighting maka tahap selanjutnya yaitu dengan melakukan pengujian proses render dengan camera turntable 360 sebanyak 200 frame/gambar image sequence dengan 24 fps dan output render dengan ukuran 1K atau 1024x1024 pixel.



Gambar 4. 17 Maya render seting.

Pada saat proses render dapat dipastikan dengan benar dikarenakan bila terdapat kesalahan dalam pengaturan seperti jumlah frame atau frame rate dan juga dipastikan bahwa output yang diambil dengan menggunakan camera yang telah kita setting secara benar. Dikarenakan bila terjadi kesalahan dalam proses rendering

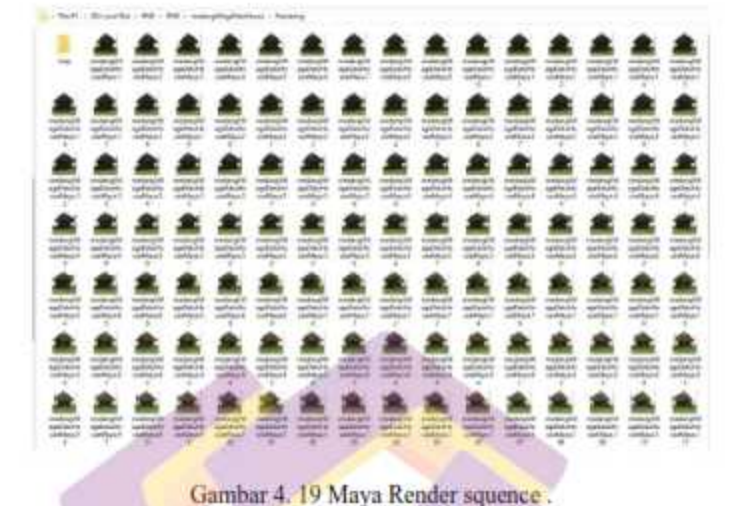
maka akan memakan waktu yang banyak dikarenakan harus mengulangi dari frame 0 sampai 200.



Gambar 4. 18 Maya Render.

4.6. Output Rendering

Dari hasil proses render yang berjalan dapat dilihat hasilnya kedalam folder yang telah kita set pada saat kita seting output image render. Hasil dari proses output render dengan sebanyak 200 data gambar image dan dengan ukuran 1024x1024 yang selanjutnya akan di cek apakah sudah berjalan dengan benar atau belum. Dari hasil output render ini selanjutnya akan di bandingkan pada hasil dari proses render pada UE4 sebagai pembanding hasil dari kualitas dan waktu render apakah lebih baik dalam 2 segi tersebut. Dari hasil render pada maya yang sudah selesai selajutnya dapat dihitung waktu yang di butuhkan untuk menyelesaikan proses render sebanyak 200 frame tersebut.



Frame	Start Time	End Time	Render Time
1	20:00:00.0000	20:00:00.0000	00:00:00.0000
2	20:00:00.0000	20:00:00.0000	00:00:00.0000
3	20:00:00.0000	20:00:00.0000	00:00:00.0000
4	20:00:00.0000	20:00:00.0000	00:00:00.0000
5	20:00:00.0000	20:00:00.0000	00:00:00.0000
6	20:00:00.0000	20:00:00.0000	00:00:00.0000
7	20:00:00.0000	20:00:00.0000	00:00:00.0000
8	20:00:00.0000	20:00:00.0000	00:00:00.0000
9	20:00:00.0000	20:00:00.0000	00:00:00.0000
10	20:00:00.0000	20:00:00.0000	00:00:00.0000
11	20:00:00.0000	20:00:00.0000	00:00:00.0000
12	20:00:00.0000	20:00:00.0000	00:00:00.0000
13	20:00:00.0000	20:00:00.0000	00:00:00.0000
14	20:00:00.0000	20:00:00.0000	00:00:00.0000
15	20:00:00.0000	20:00:00.0000	00:00:00.0000
16	20:00:00.0000	20:00:00.0000	00:00:00.0000
17	20:00:00.0000	20:00:00.0000	00:00:00.0000
18	20:00:00.0000	20:00:00.0000	00:00:00.0000
19	20:00:00.0000	20:00:00.0000	00:00:00.0000
20	20:00:00.0000	20:00:00.0000	00:00:00.0000
21	20:00:00.0000	20:00:00.0000	00:00:00.0000
22	20:00:00.0000	20:00:00.0000	00:00:00.0000
23	20:00:00.0000	20:00:00.0000	00:00:00.0000
24	20:00:00.0000	20:00:00.0000	00:00:00.0000
25	20:00:00.0000	20:00:00.0000	00:00:00.0000
26	20:00:00.0000	20:00:00.0000	00:00:00.0000
27	20:00:00.0000	20:00:00.0000	00:00:00.0000
28	20:00:00.0000	20:00:00.0000	00:00:00.0000
29	20:00:00.0000	20:00:00.0000	00:00:00.0000
30	20:00:00.0000	20:00:00.0000	00:00:00.0000
31	20:00:00.0000	20:00:00.0000	00:00:00.0000
32	20:00:00.0000	20:00:00.0000	00:00:00.0000
33	20:00:00.0000	20:00:00.0000	00:00:00.0000
34	20:00:00.0000	20:00:00.0000	00:00:00.0000
35	20:00:00.0000	20:00:00.0000	00:00:00.0000
36	20:00:00.0000	20:00:00.0000	00:00:00.0000
37	20:00:00.0000	20:00:00.0000	00:00:00.0000
38	20:00:00.0000	20:00:00.0000	00:00:00.0000
39	20:00:00.0000	20:00:00.0000	00:00:00.0000
40	20:00:00.0000	20:00:00.0000	00:00:00.0000
41	20:00:00.0000	20:00:00.0000	00:00:00.0000
42	20:00:00.0000	20:00:00.0000	00:00:00.0000

Gambar 4. 20 Waktu Maya Render squence .

Pada gambar diatas menunjukan bahwa proses render pada maya membutuhkan waktu rata-rata 3 menit untuk menyelesaikan kalkulasi render dan

untuk melakukan selama 200 frame membutuhkan total waktu sebanyak 10 jam 34menit 47 detik.

4.7. Setup UE4

Proses pengujian render yang ke dua selanjutnya dilakukan uji dengan menggunakan software game engine yang bernama unreal engine 4 (UE4). Pada tahapan ini yang pertama dilakukan yaitu dengan melakukan pengaturan pada saat awal program UE4 dibuka terdapat beberapa pilihan dikarenakan software ini di desain juga sebagai game engine maka terdapat 4 pilihan project categories yaitu game, film television and live event, architecture engineering and construction, automotive product design and manufacturing dari keempat pilihan tersebut kita pilih yang Film, television and live event.



Gambar 4. 21 New project UE4 .

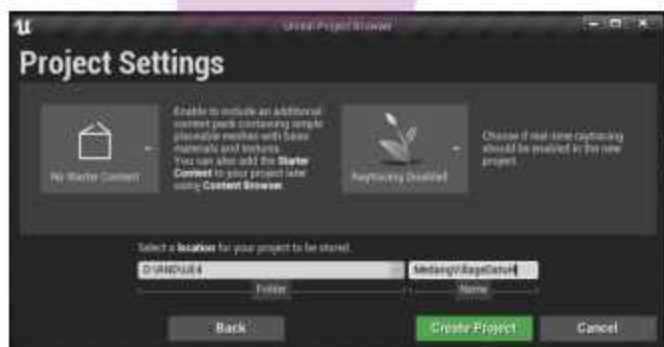
Selanjutnya pada tahapan pilihan Film, television and live event akan di hadapkan kembali dengan tiga pilihan yaitu template yang akan kita gunakan, dikarenakan penelitian ini dengan tujuan pengujian render maka kita pilih Blank

template dengan tujuan untuk mengetahui kebutuhan apa saja yang diperlukan untuk melakukan pemrosesan data hingga finish pada tahapan rendering.



Gambar 4. 22 Template project UE4.

Setelah pembuatan tempale maka langkah selanjutya dengan menentukan lokasi penyimpanan dan pemberian nama dari proyek yang akan kita gunakan dalam penelitian ini. Dalam project seting ini tidak menyertakan content yang diberikan dari bawaan software UE4 dikarenakan agar dapat diketahui keperluan yang akan kita butuhkan pada tahapan ini.



Gambar 4. 23 Location project UE4.

4.8.Import Asset 3D model UE4

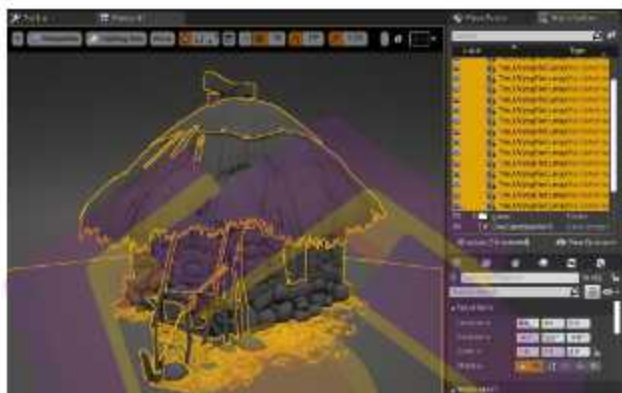
Untuk pengujian rendering pada UE4 ini dengan pengambilan data dari aset 3d model beserta semua tekstur yang berhubungan dengan objek tersebut dari masing -masing shader dari file color,specular, normal map dan opacity yang selanjutnya di tempatkan pada setiap shader masing-masing. Langkah awal pada pengambilan data ini dengan cara mengimport objek FBX file dengan nama Medang village datu house



Gambar 4. 24 Import 3d Asset UE4.

Dengan mengimport aset 3d model maka akan secara otomatis shader atau material akan masuk kedalam satu folder pada konten browser. Setelah objek masuk pada tahap import maka langkah selanjutnya yaitu dengan menempatkan objek tersebut ke dalam area viewport dengan cara pilih satu persatu atau bisa juga dengan memilih semua objek yang kemudian drag masuk kedalam viewport. Akan

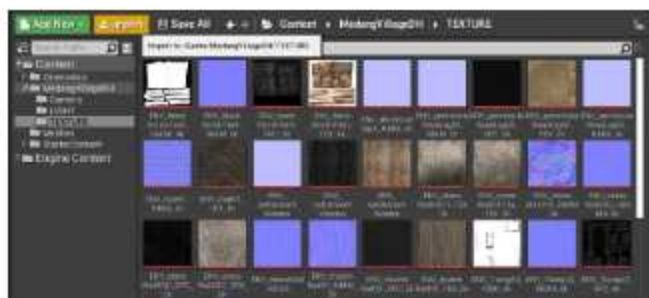
tetapi biasanya posisi objek tidak tepat pada tengah-tengah koordinat nol , untuk mempermudah dalam memposisikan pada koordinat nol yaitu dengan mengubah nilai x,y,z pada transform > location jadikan semuanya ke angka nol maka objek akan berada pada tengah koordinat x,y,z.



Gambar 4. 25 UE4 Set Location Asset.

4.9. Import Asset Texture UE4

Data yang dibutuhkan berikutnya adalah file berupa tekstur dari semua objek yang ada pada asset tersebut, untuk itu kita perlu mengimport semua file tekstur dari color, specular, normal dan opacity. Langkah yang dilakukan sama dengan pada saat mengimport aset 3d model diatas dan untuk mempermudah dalam manajemen data perlu di buat folder khusus untuk semua jenis tekstur, karena jika di campurkan dengan aset maka akan sedikit kesulitan pada saat pemilihan tekstur pada saat penempatan file pada setiap shader.



Gambar 4. 26 Import tekstur Asset UE4.

4.9.1. Assign UE4 Texture Shader Ground

Shader atau material yang ada secara default hanya akan menampilkan parameter pada base color dan untuk tekstur specular, normal map dan opacity harus di masukan pada slot sesuai dengan posisinya masing-masing.



Gambar 4. 27 UE4 Default material.

Pada langkah pertama yang di lakukan dengan menempatkan tesktur pada ground berupa tekstur rumput dengan tiga jenis tekstur yaitu base color, specular dan normal map.



Gambar 4. 28 UE4 Ground material.

4.9.2. Assign UE4 Texture Shader Stone

Pada tahap selanjutnya yaitu dengan memberikan tekstur pada dinding objek berupa tekstur batu dengan jumlah file tekstur yaitu base color atau difuse, specular dan normal map yang di tempatkan sesuai dengan posisinya.



Gambar 4. 29 UE4 Stone material.

4.9.3. Assign UE4 Texture Shader Wood

Untuk penempatan pada shader berikutnya yaitu pada bagian kayu yang terdapat dibagian dinding rumah medang village datu huse yang terbuat dari papan kayu yang disusun melingkari area rumah tersebut.



Gambar 4. 30 UE4 Wood material.

4.9.4. Assign UE4 Texture Shader Roof

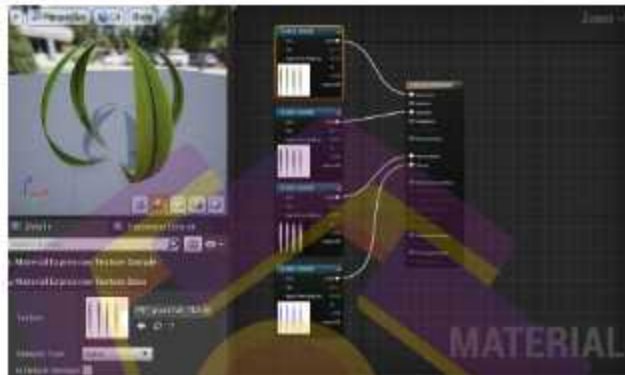
Penempatan sahadar berikutnya yang terdapat pada atap bangunan dengan bahan kain goni sebagai penutup tambahan pada bagian ujung atap rumah medang village datu house.



Gambar 4. 31 UE4 Roof material.

4.9.5. Assign UE4 Texture Shader Grass

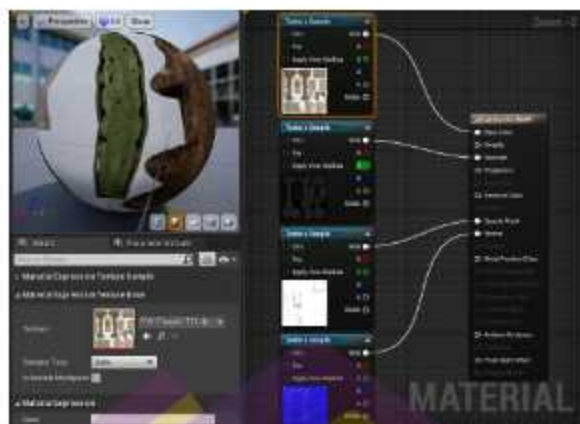
Selanjutnya proses penempatan tekstur rumput pada shader material pada rumput dengan jenis file color tekstur, specular, normal map dan opacity atau masking yang berfungsi sebagai pemotong dari color tekstur.



Gambar 4. 32 UE4 Grass material.

4.9.6. Assign UE4 Texture Cloth Shape.

Pada cloth shape terdapat satu tekstur tambahan dengan nama opacity atau masking yang berfungsi sebagai pemotong dari color tekstur. Selain dari tekstur opacity ada tambahan yang harus di ganti pada parameter material yang secara default blend mode menggunakan opaque maka diubah menjadi maksed sehingga hasil dari tekstur opacity akan berfungsi dengan benar.



Gambar 4. 33 UE4 Cloth Shape material.

4.9.7. Assign UE4 Texture Black Smith.

Selanjutnya penempatan pada shader black smith yang terdapat pada atap rumah medang village datu house dengan tekstur color, specular, normal map dan opacity atau masking sebagai pemotong pada tekstur color.



Gambar 4. 34 UE4 Black Smith material.

4.9.8. Assign UE4 Texture Grass Small.

Penempatan material selanjutnya yaitu pada rumput yang terdapat pada beberapa bagian sudut rumah medang village datu house dengan komposisi tiga tekstur yaitu color map, specular map dan normal map.



Gambar 4. 35 UE4 Grass Small material.

4.9.9. Assign UE4 Texture Shader Rope.

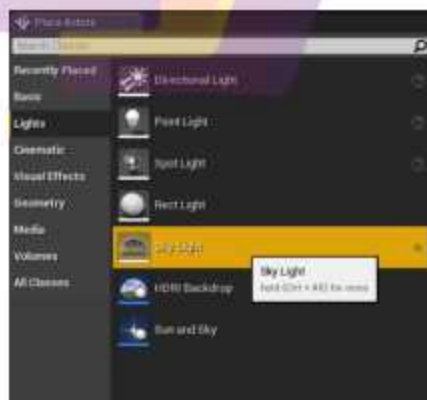
Tekstur yang terakhir yaitu pada objek tali yang mengikat pada bagian rumah medang village datu house dengan komposisi tekstur yaitu color map, specular map dan normal map yang di tempatkan sesuai dengan slot pada shader rope.



Gambar 4. 36 UE4 Rope material.

4.10. Lighting Sky Light UE4

Tahapan selanjutnya yaitu pemberian lighting atau pencahayaan yang berfungsi sebagai penerangan pada area aset medang village datu house. Pada tahapan ini yang di butuhkan sama dengan proses yang dilakukan pada maya yaitu dengan menggunakan sky light dan directional light. Untuk pembuatan sky light pada UE4 terdapat pada Place Actor kemudian dikategori light akan muncul beberapa jenis light kemudian pilih yang Sky Light.



Gambar 4. 37 Object Sky light.

Di dalam pencahayaan sky light pada pengujian ini dengan menggunakan intensity 0,2 dengan alasan dikarenakan untuk menyesuaikan dengan pencahayaan yang di hasilkan mada maya. Secara default bawaan dari software UE4 memberikan nilai paramater intensity pada angka 1.



Gambar 4. 38 Parameter Sky light.

4.11. Directional light UE4

Pencahayaan yang kedua yaitu dengan menggunakan directional light yang berfungsi sebagai lampu utama yang menyinari area pada objek medan village datu house. Tata letak pencahayaan pada UE4 akan disamakan dengan kondisi penelitian pada maya yang mempunyai kesamaan dalam pencahayaan yaitu dengan sky light dan directional light. Dalam pemberian pecahayaan directional light pada UE4 yaitu dengan memulih menu pada place actor dan dalam kategori light kemudian pilih type lampu Directional Light dan drag masuk kedalam viewport



Gambar 4. 39 Object Directional light.

Setelah objek directional light berada di dalam viewport maka langkah selanjutnya yaitu dengan menyamakan orientasi arah dari pencahayaan dengan yang berada penelitian pada maya,y,z adalah $-45,0,0$. Secara default dari UE4 sumbu rotasi xyz pada posisi $0,-45,0$ maka untuk menyamakan dengan posisi pada maya yaitu dengan mengumah arah rotasi menjadi $90,-45,-90$ maka arah pencahayaan akan sama dengan posisi pada maya.



Gambar 4. 40 Default Directional light.

Pada langkah berikutnya setelah orientasi disamakan dengan benar maka ada parameter yang harus disesuaikan lagi yaitu intensity dari cahaya tersebut, dikarenakan secara default intensity pada pencahayaan directional light adalah bernilai 10 lux dan sedangkan hasil dari nilai tersebut masih terlalu terang dalam hasil pencahayaan maka untuk menyamakan pada hasil maya render tersebut maka di atur kembali pada nilai intensity yang di turunkan hingga menjadi 1,5 lux.



Gambar 4. 41 Directional light.

Dari hasil pemberian lighting Directional light disini mendapatkan yang belum sesuai dengan hasil pencahayaan yaitu terkait dengan hasil bayangan atau shadow yang dihasilkan dari pencahayaan directional light yang secara default terlihat bayangan yang soft atau kurang tegas, sedangkan yang dihasilkan pada render maya dengan bayangan yang tegas.



Gambar 4. 42 Shadow Directional light.

Dalam tahapan Lighting masih di perlukan proses penyesuaian terhadap hasil bayangan atau shadow. Untuk mengatur parameter yang sesuai dengan hasil pada render maya maka diperlukan kembali untuk pengujian nilai parameter terhadap hasil pada shadow directional lighting. Dari parameter dynamic shadow distance pada UE4 secara default pada nilai 20000.0 maka hasil dari nilai tersebut menjadi soft shadow. Selanjutnya dari hasil pengujian ini didapatkan nilai parameter shadow yang sesuai dengan hasil render pada maya yaitu dengan nilai dynamic shadow distance antara 600.0- 1500.0 yang menghasilkan bayangan yang tegas sesuai dengan render pada maya.

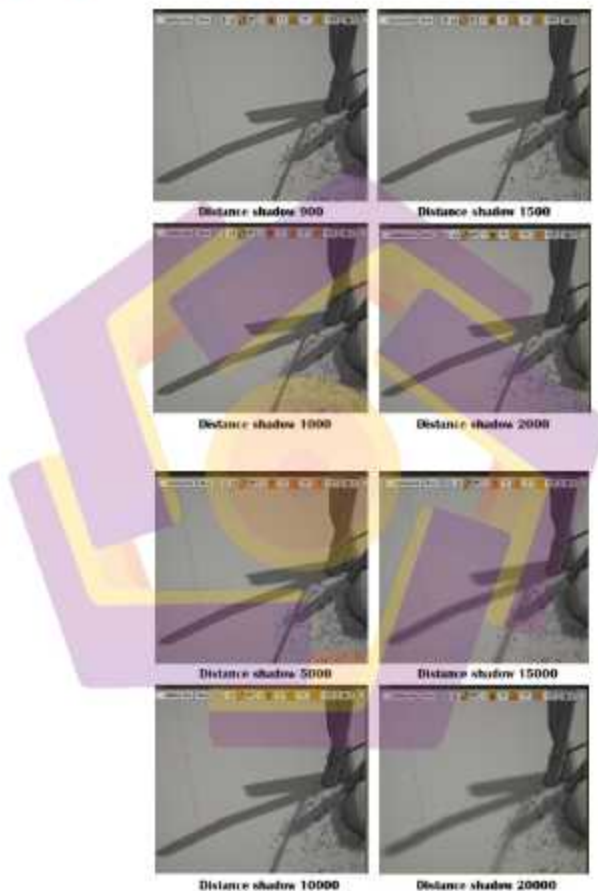
Hasil dari pengujian pada shadow directional light dapat dilihat pada gambar bawah dengan muali pengujian yang dilakukan terhadap pencahayaan pada aset medan village datu house menggunakan pencahayaan directional light yang menghasilkan shadow dapat beragam dari tanpa adanya shadow ,soft shadow dan hard shadow. Pada awal pengujian tersebut dimulai dari nilai distance dynamic shadow pada angka 100 yang menghasilkan tanpa adanya bayangan yang nampak pada objek sehingga nilai dinaikan kembali dengan angka 200 dengan hasil

shadow hanya terlihat tidak tampak sepenuhnya atau terpotong. Selanjutnya dilakukan penambahan kembali dengan memberikan nilai ke angka 300 dengan hasil dapat terlihat semua shadow dengan hasil masih sedikit terlihat blur maka dilakukan penambahan nilai kembali beberapa kali dari 400,500,600,700 dan 800. Dari hasil yang terlihat bayangan yang tegas dari 600 hingga 800.



Gambar 4. 43 Distance Shadow 100-800.

Selanjutnya dilakukan kembali pengujian untuk menaikkan nilai pada distance shadow untuk mendapatkan nilai range yang bisa digunakan untuk mendapatkan bayangan tegas atau hard shadow.



Gambar 4. 44 Distance Shadow 900-20000.

Dari hasil pengujian pada nilai distace shadow yang dilakukan dari nilai 100 hingga nilai 20000 maka dapat di ketahui bahwa untuk mendapatkan hasil shadow atau bayangan yang tegas dapat digunakan pada angka 600 hingga 1500.

4.12. Sequencer Camera

Pada tahapan berikutnya yaitu penempatan camera pada UE4 dimana camera yang digunakan harus sama dengan yang di gunakan pada maya. Untuk mengecek atau menyamakan camera yaitu dengan nilai focal length pada maya yaitu 35, maka diperlukan pengecekan kembali pada camera yang akan digunakan supaya hasil output image render akan sama dengan hasil pada maya render.



Gambar 4. 45 Focal length Camera UE4.

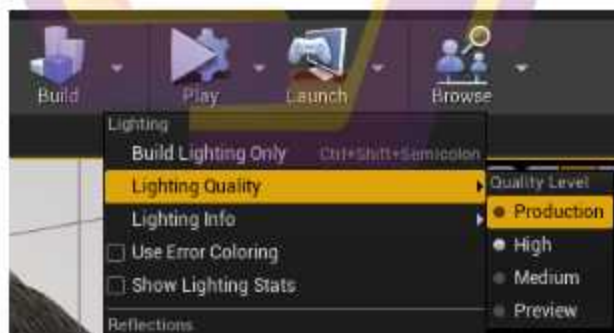
Selanjutnya untuk menempatkan cinematic camera dengan jumlah 200 frame dan dengan 24 fps yaitu pada tab sequencer terdapat pilihan CineCameraActor pada klik kanan pilih import camera yang berasal dari file FBX camera sehingga akan secara otomatis menyesuaikan dengan animasi camera turntable yang dari camera maya.



Gambar 4. 46 Squence Camera UE4.

4.13. Build Lighting Quality

Pada tahapan ini terdapat proses yang bernama build lighting quality sebagai proses render yang terdapat pada software UE4 ini. Dalam tahapan ini terdapat pilihan dari kualitas pencahayaan diantaranya dari preview,medium,high dan production sebagai pilihan yang bisa di gunakan dalam proses render.



Gambar 4. 47 Lighting Quality.

Pada proses kalkulasi render disini akan membutuhkan waktu untuk memprosesnya dari exporting scene,lightmass starting hingga finish. Proses

realtime render pada UE4 akan berbeda dengan proses yang dilakukan pada maya dikarenakan UE4 di desain sebagai engine game maka sistem render bukan frame by frame melainkan secara realtime render. Dalam tahapan proses render pada UE4 ini membutuhkan waktu sebanyak 26 menit 47 detik untuk menyelesaikan hasil render, dari hasil waktu tersebut maka dapat kita ketahui waktu rata-rata pada 200 frame yaitu 8.035 detik per frame.

```

LogVolumetricLightmapart: Imported Volumetric Lightmap in 0.073s
LogVolumetricLightmapart: Indirection Texture 16x16x16 = 1.0Mb
LogVolumetricLightmapart: BrickData (all levels) 0.10Mb (truncated 0.2Mb due to 0.015000 VnBrickError)
LogVolumetricLightmapart: Bricks as depts
LogVolumetricLightmapart: 0: 20.0% covering 100.0% of volume
LogVolumetricLightmapart: 1: 12.0% covering 8.3% of volume
LogVolumetricLightmapart: 2: 2.0% covering 0.2% of volume
LogVolumetricLightmapart: Bricks in each level
LogVolumetricLightmapart: Main: 3 bricks 100.0% (Persistent Level)
LogTexture: Display: Building textures: H5_Lightmap_1 (AutoGen, 16240x1624)
LogTexture: Display: Building textures: L2_Lightmap_1 (AutoGen, 16240x1624)
LogStaticLightingSystem: Main: BuildData storing Lightmap data for 33 meshes in 2 LightmapSourceClusters (16.5 Meshes per cluster)
Lightmaps on DESKTOP-VI7756M: 6:34 min total, 11.4 sec importing, 2.65 sec setup, 3127 tex photons, 3154 min processing, 0 ms extra exporting [12/33 meshes], Threads: 26147 min total, 5142 min processing.
LogObjectHash: Compacting FIDefaultHashTables data took 3.57ms
LogStaticLightingSystem: Illumination: 0:46 min (145 ms encoding lightmaps, 0 ms encoding shadowmaps)
LogStaticLightingSystem: Lightmap texture memory: 2.1 MB (2.1 MB streaming, 0.0 MB non-streaming), 2 textures
LogStaticLightingSystem: Shadowmap texture memory: 0.0 MB (0.0 MB streaming, 0.0 MB non-streaming), 0 textures
LogBuildLightInteractions
Lights with default interactions: 0
Primitives with default interactions: 0
See: WWW_CHECK_RENDERER_SETTINGS
  
```

Gambar 4. 48 Output Log.

4.14. Post Process Volume

Fitur yang diberikan dari UE4 terdapat Pos Process Volume dengan adanya fitur ini maka hasil dari render masih dapat di atur kembali pada tingkan kecerahan dan warna pada hasil render dengan realtime render pada viewport. Pada tahap ini diperlukan tambahan with balance sebagai pengatur warna yang sesuai pada maya render.



Gambar 4. 49 Post Process Volume.

4.15. Render Movie

Pada tahapan selanjutnya yaitu proses render movie yang bertujuan untuk pengambilan pada realtime render pada viewpot dengan cara mengcapture berdasarkan dari sudut pandang camera squencer yang telah dilakukan sebelumnya. Pada tahap ini yang di perlukan yaitu dalam hal jenis file yang akan kita ambil dapat bermacam-macam diantaranya file .avi, .png, .jpg, .exr, .bmp, render passes. Untuk menyamakan output pada render maya maka dalam hal ini dipilih jenis file .jpg sebagai outputnya dan dengan ukuran 1024x1024. Setelah setingan selesai maka selanjutnya dapat menentukan output directory sebagai lokasi pada hasil render nantinya dan setelah ditentukan maka tinggal capture movie dan tunggu beberapa saat hingga selesai.



Gambar 4. 50 Render movie.

4.16. Sequence Render Image

Hasil render sequence pada UE4 dapat dilihat bahwa dari proses render yang telah selesai dengan proses capture movie tersebut maka dapat dihasilkan dengan type jenis file jpg dan dengan ukuran file 1024x1024 yang sesuai dengan camera pada maya dan jumlah frame yang dihasilkan sebanyak 200 frame.



Gambar 4. 52 Hasil visual Render Maya dan UE4.

4.18. Hasil perbandingan waktu render Maya dan UE4

Dari hasil pengujian yang dilakukan dapat dilihat bahwa proses render yang dilakukan pada maya memerlukan waktu sebanyak 10 jam 34menit 47 detik. Sedangkan proses render yang dilakukan pada UE4 memerlukan waktu sebanyak 26 menit 47 detik dengan hasil prosentase 95,7% lebih cepat dari maya.



Gambar 4. 53 Hasil Analisis waktu Render Maya dan UE4.

4.19. Pengujian render pada objek pohon

Pada tahapan ini dilakukan dengan objek yang berbeda yaitu dengan menggunakan 4 pohon sebagai bahan uji render pada maya dan uji render pada UE4. Proses pengujian ini akan dilakukan dengan pencahayaan dan camera yang sama dan jumlah frame yang akan diambil yaitu sebanyak 60 frame.

4.19.1 Pengujian big tree 01

Pada roses pengujian yang dilakukan pada objek big tree 01 yaitu dengan hasil waktu render yang dibutuhkan pada maya sebanyak 3jam, 13 menit,59 detik sedangkan waktu yang di butuhkan pada UE4 sebanyak 55 menit, 43 detik dengan hasil prosentase 71,2% lebih cepat dan gambar hasil render sebagai berikut.



Gambar 4. 54 Hasil visual Render big tree01.



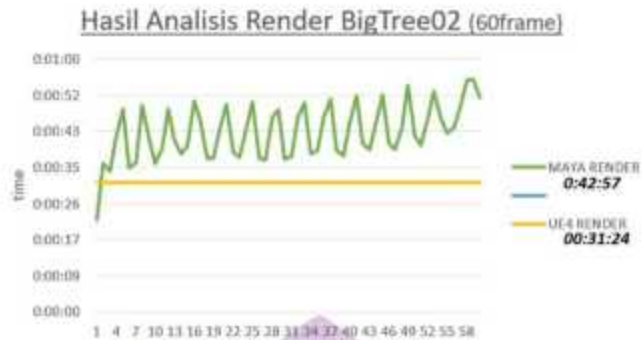
Gambar 4. 55 Hasil Analisis waktu Render big tree01.

4.19.2 Pengujian big tree 02

Pada roses pengujian yang dilakukan pada objek big tree 02 yaitu dengan hasil waktu render yang dibutuhkan pada maya sebanyak 42 menit,57 detik sedangkan waktu yang di butuhkan pada UE4 sebanyak 31 menit, 24 detik dengan hasil prosentase 26,8% lebih cepat dan gambar hasil render sebagai berikut.



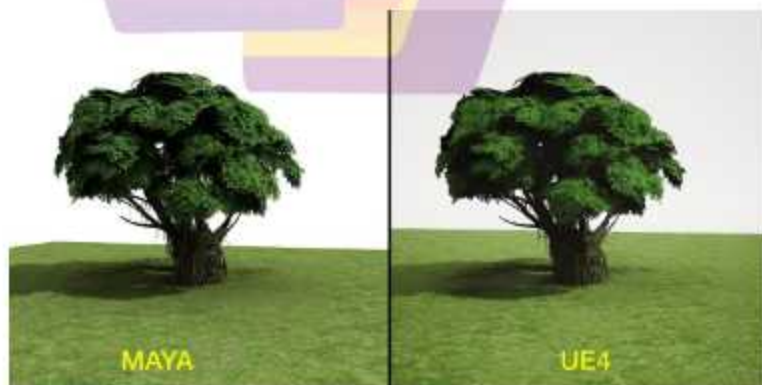
Gambar 4. 56 Hasil visual Render big tree02.



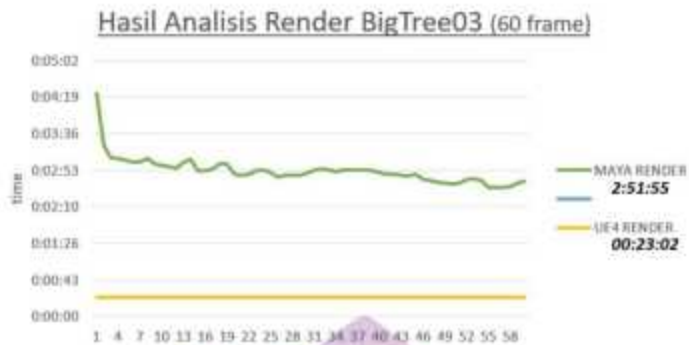
Gambar 4. 57 Hasil Analisis waktu Render big tree02.

4.19.3 Pengujian big tree 03

Pada roses pengujian yang dilakukan pada objek big tree 01 yaitu dengan hasil waktu render yang dibutuhkan pada maya sebanyak 2jam, 51 menit,55 detik sedangkan waktu yang di butuhkan pada UE4 sebanyak 23 menit, 02 detik dengan hasil prosentase 86,6% lebih cepat dan gambar hasil render sebagai berikut.



Gambar 4. 58 Hasil visual Render big tree03.



Gambar 4. 59 Hasil Analisis waktu Render big tree03.

4.19.4 Pengujian big tree 04

Pada roses pengujian yang dilakukan pada objek big tree 01 yaitu dengan hasil waktu render yang dibutuhkan pada maya sebanyak 1jam, 28 menit,42 detik sedangkan waktu yang di butuhkan pada UE4 sebanyak 26 menit, 53 detik dengan hasil prosentase 69,6% lebih cepat dan gambar hasil render sebagai berikut.



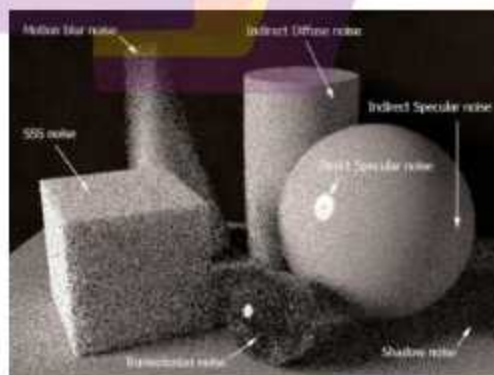
Gambar 4. 60 Hasil visual Render big tree04.



Gambar 4. 61 Hasil Analisis waktu Render big tree04.

4.20. Parameter kualitas render

Kualitas gambar yang dihasilkan pada saat merender akan menjadi baik bilamana image tersebut tidak terdapat noise atau terdapat bintik-bintik kotor yang menyebabkan gambar menjadi tidak bagus, contoh adanya nois dapat dilihat pada gambar 4.62 . Sebagai perbandingan antara hasil render yang mengandung noise pada nilai samples 1 dan 2 dan dengan hasil render yang tidak mengandung noise pada nilai samples 3 dapat dilihat pada gambar 4.63.



Gambar 4. 62 Render noise.



Gambar 4. 63 Perbandingan render noise.

Hasil pengujian render yang telah dilakukan terhadap beberapa 3D model pada objek rumah dengan jumlah frame sebanyak 200 dan 4 objek pohon dengan jumlah 60 frame dengan camera turntable 360 dapat dilihat pada tabel 2.2 . Hasil output render UE4 sudah sesuai kualifikasi untuk film animasi dengan hasil output yang sama pada maya yaitu 1024x1024 pixel dan jumlah FPS yang sama yaitu 24FPS.

Tabel 2. 2 Hasil uji render pada maya dan UE4

HASIL UJI RENDER MAYA DAN UE4					
No	Nama 3D Asset	Jumlah frame	Waktu Render Maya	Waktu Render UE4	Prosentase Hasil uji render
01	Medang House	200	10:34:47	00:26:47	95,7%
02	Bigtree01	60	3:13:59	00:55:43	71,2%
03	Bigtree02	60	0:42:57	00:31:24	26,8%
04	Bigtree03	60	2:51:55	00:23:02	86,6%
05	Bigtree04	60	1:28:42	00:26:53	69,6%

Image yang telah selesai di render selanjutnya akan di ajukan kepada produser untuk dinilai apakah hasil kualitas render sudah sesuai atau belum, jika belum sesuai maka harus dilakukan revisi kembali dan bilamana hasil sudah sesuai maka image hasil render tersebut akan mendapatkan tanda aprovel sebagai persetujuan untuk di lanjutkan pada proses selanjutnya yaitu pada divisi kompositing.

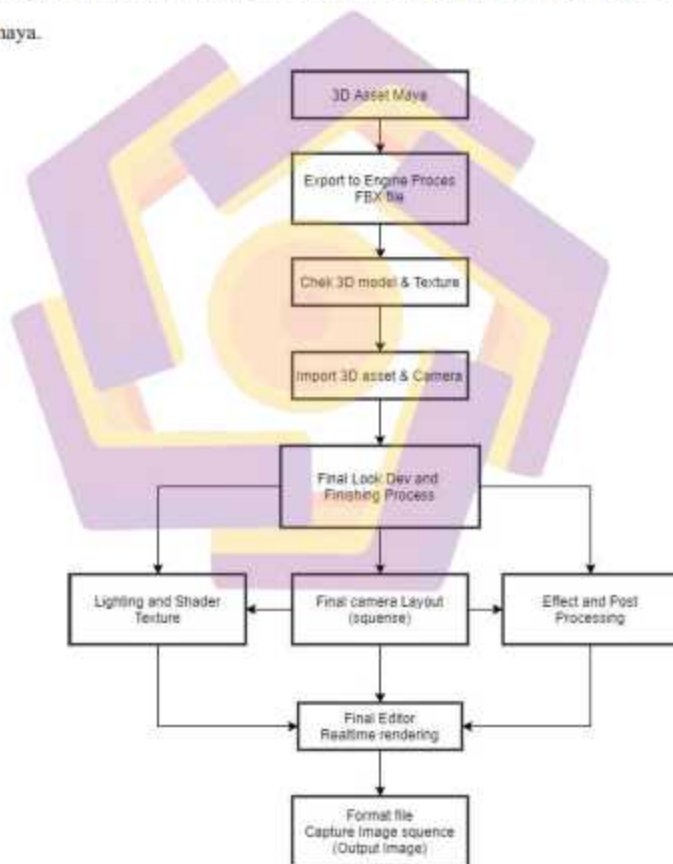


Gambar 4. 64 Render approve

Proses render yang tepat pada UE4 ini akan lebih optimal bilamana jumlah frame yang akan di proses lebih dari 50 frame atau terdapat beberapa shot camera pada scene atau lokasi tersebut, akan tetapi bilamana proses render yang dilakukan hanya 1 frame dengan tujuan hanya untuk render background dan kurang dari 50 frame maka akan lebih tepat menggunakan maya sebagai proses rendernya.

4.21. Pipeline Realtime Render

Dari hasil perbandingan render tersebut maka dapat di hasilkan pipeline terkait dengan faktor-faktor yang dapat mempengaruhi kecepatan pada proses render. Hasil dari image render adanya perbedaan pada proses render pada maya dengan proses kalkulasi render frame per frame, sedangkan proses render pada UE4 dengan kalkulasi secara realtime yang dapat menghasilkan proses waktu lebih cepat di bandingkan proses render pada maya.



Gambar 4. 65 Pipeline Realtime render.

BAB V

PENUTUP

5.1. Kesimpulan

Kesimpulan yang dapat diambil dari penelitian pada proses render secara realtime ini dapat diambil beberapa kesimpulan yaitu:

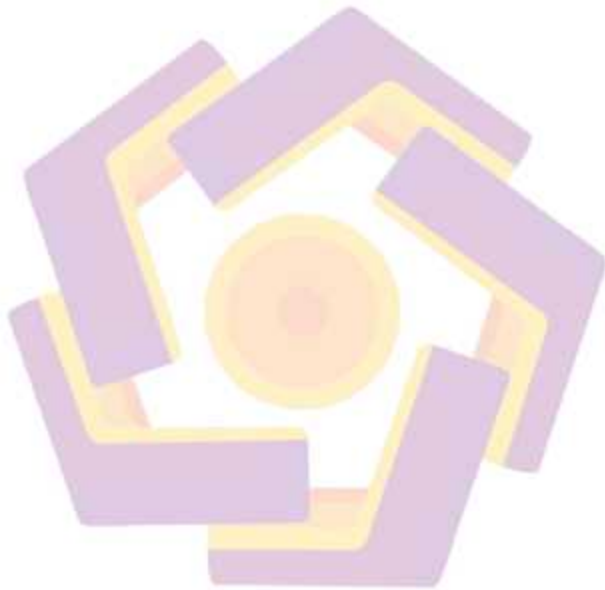
1. Terdapat perbedaan antara produktifitas render secara frame per frame dengan render secara realtime, dalam proses render secara realtime dapat menghasilkan waktu render lebih cepat dibandingkan render dengan cara frame per frame.
2. Dengan proses realtime render dapat mengoptimasikan privity final render dikarenakan tidak perlu membutuhkan waktu lagi untuk kalkulasi pada setiap akan melakukan proses privity render.
3. Proses render secara realtime akan lebih efisien jika digunakan dengan jumlah frame lebih dari 50 frame.
4. Dengan Pipeline render secara realtime pada UE4 dapat mempengaruhi terhadap proses waktu render menjadi lebih efisien hingga lebih dari 50%.

5.2. Saran

Agar penelitian dimasa mendatang bisa menjadi lebih baik, beberapa saran yang dapat diambil antara lain:

1. Untuk melengkapi pipeline pembuatan film 3D animasi di perlukan penelitian pada divisi animasi yang terkait dengan data animasi supaya dapat di integrasikan pada setiap shot camera.

2. Pengembangan yang masih di perlukan pada divisi render diantaranya pada proses simulasi dari hair, cloth, dan vfx .
3. Objek yang di teliti tidak hanya objek diam saja akan tetapi dapat dilakukan pada objek yang bergerak seperti karakter yang di animasikan.

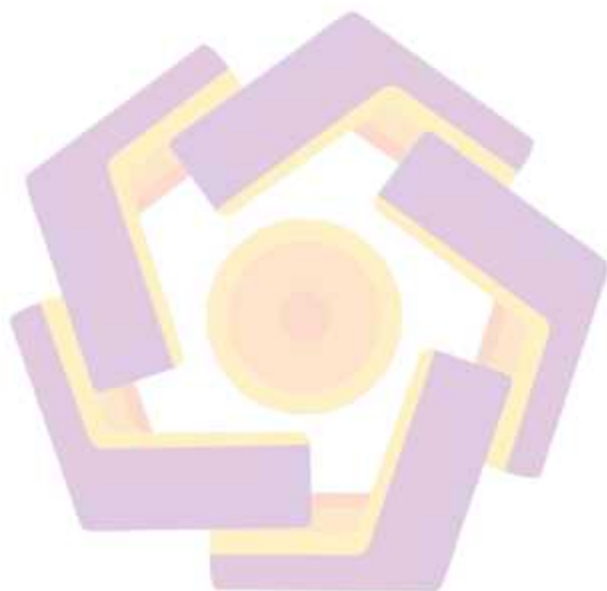


DAFTAR PUSTAKA

- Chong, Anthony, Alexei Sourin, and Konstantin Levinski. 2006. "Grid-Based Computer Animation Rendering." *Proceedings - GRAPHITE 2006: 4th International Conference on Computer Graphics and Interactive Techniques in Australasia and Southeast Asia* 1(212): 39-47.
- Fascione, Luca et al. 2018. "Manuka." *ACM Transactions on Graphics* 37(3): 1-18. <https://dl.acm.org/doi/10.1145/3182161>.
- Francisco, Roig Perez. 2011. "Photorealistic Physically Based Render Engines A Comparative Study." (September): 25.
- Jeremias-Vila, Pol et al. 2018. "The Present and Future of Real-Time Graphics in Film." *ACM SIGGRAPH 2018 Panels, SIGGRAPH 2018*.
- Larsson-ståhl, Jennifer, and Jennifer Larsson-ståhl. 2020. "A Study on the Perceived Realism Of Strand-Based Hair Simulated By Style." (June).
- Li, Cong, Lingjun Yang, and Sanxing Cao. 2015. "Design and Implementation of Games Based on Html5." (Iccis): 233-38.
- Liu, Edward, Ignacio Llamas, Juan Cañada, and Patrick Kelly. 2019. Ray Tracing Gems: High-Quality and Real-Time Rendering with DXR and Other APIs *Cinematic Rendering in UE4 with Real-Time Ray Tracing and Denoising*.
- Lowe, Kunta Kwaku. 2019. "Analyzing and Developing Aspects of the Artist Pipeline for Clemson University Art."
- Merkel, Lukas, Christoph Berger, and Stefan Braunreuther. 2019. "Advances in Human Factors in Wearable Technologies and Game Design." 795(January): 198-207. <http://link.springer.com/10.1007/978-3-319-94619-1>.

- Morozov, Vadim, and Information Technology. 2019. "PHYSICALLY BASED 3D RENDERING USING UNBIASED RAY TRACING."
- Murdock, Kelly L. 2019. "Autodesk Maya 2019."
- Peddie, Jon. *Ray Tracing : A Tool for All*.
- Rhee, Taehyun, Lohit Petikam, Benjamin Allen, and Andrew Chalmers. 2017. "MR360: Mixed Reality Rendering for 360 Panoramic Videos." *IEEE Transactions on Visualization and Computer Graphics* 23(4): 1-1. <http://ieeexplore.ieee.org/document/7829404/>.
- Stadler, Viktor, and Helmut Hlavacs. 2018. "Blind Adventure - A Game Engine for Blind Game Designers." *CHI PLAY 2018 - Proceedings of the 2018 Annual Symposium on Computer-Human Interaction in Play*: 511-22.
- Tamboli, Roopak R. et al. 2018. "A High-Angular-Resolution Turntable Data-Set for Experiments on Light Field Visualization Quality." *2018 10th International Conference on Quality of Multimedia Experience, QoMEX 2018*: 1-3.

LAMPIRAN



1. Hasil image render squence Maya















2. Hasil image render squence UE4













