

**SIMULASI GERAK HEWAN GAJAH MENGGUNAKAN METODE
INVERSE KINEMATICS DALAM PEMBUATAN ANIMASI 3D
ASBABUN NUZUL AL-FIIL**

SKRIPSI

**OLEH:
M. BAGUS SYAIFULLAH
NIM. 13650092**



**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2019**

**SIMULASI GERAK HEWAN GAJAH MENGGUNAKAN METODE
INVERSE KINEMATICS DALAM PEMBUATAN ANIMASI 3D
ASBABUN NUZUL AL-FIIL**

SKRIPSI

**Diajukan kepada:
Universitas Islam Negeri (UIN) Maulana Malik Ibrahim Malang
Untuk Memenuhi Salah Satu Persyaratan Dalam
Memperoleh Gelar Sarjana Komputer (S.Kom)**

**Oleh :
M. BAGUS SYAIFULLAH
NIM. 13650092**

**JURUSAN TEKNIK INFORMATIKA
FAKULTAS SAINS DAN TEKNOLOGI
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM
MALANG
2019**

HALAMAN PERSETUJUAN

**SIMULASI GERAK HEWAN GAJAH MENGGUNAKAN METODE
INVERSE KINEMATICS DALAM PEMBUATAN ANIMASI 3D
ASBABUN NUZUL AL-FIIL**

SKRIPSI

**OLEH:
M. BAGUS SYAIFULLAH
NIM. 13650092**

**Telah Diperiksa dan Disetujui untuk Diuji
Tanggal : Juni 2019**

Dosen Pembimbing I

**Dr. M. Faisal, MT
NIP. 19740510 200501 1 007**

Dosen Pembimbing II

**Hani Nurhayati, MT
NIP. 19780625 200801 2 006**

**Mengetahui,
Ketua Jurusan Teknik Informatika
Fakultas Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang**



**Dr. Cahyo Crysdiyan
NIP. 19740424 200901 1 008**

HALAMAN PENGESAHAN

**SIMULASI GERAK HEWAN GAJAH MENGGUNAKAN METODE
INVERSE KINEMATICS DALAM PEMBUATAN ANIMASI 3D
ASBABUN NUZUL AL-FIIL**

SKRIPSI

Oleh :
M. BAGUS SYAIFULLAH
NIM. 13650092

Telah Dipertahankan di Depan Dewan Penguji Skripsi dan
Dinyatakan Diterima Sebagai Salah Satu Persyaratan untuk
Memperoleh Gelar Sarjana Komputer (S.Kom)
Tanggal : Juni 2019

Susunan Dewan Penguji

Tanda Tangan

Penguji Utama : Fachrul Kurniawan, M.MT
NIP. 19771020 200912 1 001

()

Ketua Penguji : Yunifa Miftachul Arif, M.T
NIP. 19830616 201101 1 004

()

Sekretaris Penguji : Dr. M. Faisal, MT
NIP. 19740510 200501 1 007

()

Anggota Penguji : Hani Nurhayati, MT
NIP. 19780625 200801 2 006

()

Mengetahui dan Mengesahkan,
Ketua Jurusan Teknik Informatika
Kampus Sains dan Teknologi
Universitas Islam Negeri Maulana Malik Ibrahim Malang



Dr. Caryo Crysdiyan
NIP. 19740424 200901 1 008

HALAMAN PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan dibawah ini :

Nama : M. Bagus Syaifullah
Nim : 13650092
Fakultas / Jurusan : Sains dan Teknologi / Teknik Informatika
Judul Skripsi : **SIMULASI GERAK HEWAN GAJAH
MENGUNAKAN METODE *INVERSE KINEMATICS*
DALAM PEMBUATAN ANIMASI 3D ASBABUN
NUZUL AL-FIIL.**

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar – benar merupakan hasil karya saya sendiri, bukan merupakan pengambilalihan data, tulisan atau pikiran orang lain yang saya akui sebagai hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila dikemudian hari terbukti atau dapat dibuktikan Skripsi ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, Mei 2019

Yang membuat pernyataan



M. Bagus Syaifullah
NIM. 13650092

MOTTO

“BODO AMAT DENGAN KEGAGALAN”

-Muhammad Bagus Syaifullah-



HALAMAN PERSEMBAHAN

Alhamdulillah puji syukur kehadiran Allah SWT atas segala nikmat yang diberikan sehingga penulis dapat menyelesaikan predikat mahasiswa abadi tanpa menunggu surat peringatan dari jurusan. Sholawat serta salam selalu tercurahkan kepada baginda Nabi Muhammad SAW, yang telah membimbing umatnya menuju jalan yang benar.

Skripsi ini saya persembahkan untuk bapak yang telah mendidik saya dengan penuh kedisiplinan namun masih belum bisa saya terapkan dalam kehidupan saya. Mamak yang selalu membimbing saya untuk selalu bersabar dalam menjalani apapun dan selalu mengasihi orang-orang disekitar saya yang belum bisa saya terapkan juga. Serta seluruh keluarga dan para tetangga yang tidak mempunyai kontribusi yang signifikan terhadap kuliah saya dan selalu bertanya kapan saya lulus.

Tidak lupa untuk sahabat-sahabat saya yang mendukung dan berkontribusi dalam kehidupan saya saat berada di kampus tercinta UIN Malang. Teman-teman satu perjuangan Teknik Informatika 2013, sahabat-sahabat saya dalam naungan PMII rayon pencerahan Galileo terutama generasi micin cangkarok. Serta penghuni kontrakan 70 tempat saya tinggal beberapa tahun baik Alibi selaku ketua suku, Dzulfikri selaku sekben dan pembimbing penulisan dibalik layar, serta anggota-anggota musiman kontrakan. Ferly Firdaus NA yang sampai saat ini saya tidak tahu kepanjangan NA dari namanya, serta adik-adik saya terutama Nikmatur Rizqi dan Ulfa Hidayati yang selalu menemani dan menyemangati saya dalam penulisan ini. Serta jodoh saya dimasa depan dan mantan saya yang namanya tidak bisa saya sebutkan karena kegalakan pacarnya yang mungkin membaca ini.

Terima kasih sebesar-besarnya untuk para dosen yang dengan sabar membimbing saya dalam perkuliahan saya selama ini terutama Dr. Muhammad Faisal, M.T, Hani Nurhayati, M.T, Roro Indah Melani, M.Sc, Fatchurrohman, M.Kom, Fachrul Kurniawan M.MT, dan Yunifa Miftachul Arif, M.T yang telah membimbing saya dalam penulisan ini.



KATA PENGANTAR

Assalamualaikum Warahmatullahi Wabarokatuhu.

Alhamdulillah Robbil 'Alamiin, segala puji bagi Allah yang selalu memberikan kesehatan, ketabahan dan kekuatan dalam proses penyelesaian skripsi ini. Sholawat serta salam selalu tercurahkan kepada Nabi Muhammad SAW yang telah memberikan teladan, bimbingan dan petunjuk, sehingga umat manusia menjadi lebih beradab.

Dalam menyelesaikan skripsi ini, banyak pihak yang telah memberikan bantuan. Atas segala bantuan yang telah diberikan, penulis ingin menyampaikan doa dan ucapan terima kasih yang sedalam-dalamnya kepada:

1. Dr. M. Faisal, M. T., selaku dosen pembimbing I yang telah meluangkan waktu untuk membimbing, mengarahkan dan memberi masukan kepada penulis dalam pengerjaan skripsi ini hingga akhir.
2. Hani Nurhayati, selaku dosen pembimbing II yang telah membimbing serta memberikan masukan kepada penulis dalam pengerjaan skripsi ini.
3. Dr. Cahyo Crysdian, selaku Ketua Jurusan Teknik Informatika yang telah memberikan motivasi untuk terus berjuang.
4. Segenap dosen teknik informatika yang telah memberikan bimbingan keilmuan kepada penulis selama masa studi.
5. Teman-teman seperjuangan teknik informatika Fortinity 2013.

Berbagai kekurangan dan kesalahan mungkin pembaca temukan dalam penulisan skripsi ini, untuk itu penulis menerima segala kritik dan saran yang membangun dari pembaca sekalian. Semoga apa yang menjadi kekurangan bisa

disempurnakan oleh peneliti selanjutnya dan semoga karya ini senantiasa dapat memberi manfaat.

Wassalamualaikum Warahmatullahi.Wabarokatuhu.

Malang, Mei 2019

Penulis



DAFTAR ISI

HALAMAN JUDUL	i
HALAMAN PERSETUJUAN.....	ii
HALAMAN PENGESAHAN	iii
HALAMAN PERNYATAAN KEASLIAN TULISAN	iv
MOTTO	v
HALAMAN PERSEMBAHAN.....	vi
KATA PENGANTAR.....	viii
DAFTAR ISI.....	x
DAFTAR GAMBAR.....	xiii
DAFTAR TABEL.....	xv
ABSTRAK	xvi
ABSTRACT	xvii
ملخص.....	xviii
BAB I PENDAHULUAN	1
1.1 Latar Belakang Masalah.....	1
1.2 Pernyataan Masalah	4
1.3 Batasan Masalah.....	4
1.4 Tujuan Penelitian	4
1.5 Manfaat. Penelitian	4
1.6 Sistematika Penulisan.....	5
BAB II TINJAUAN PUSTAKA.....	6
2.1 Simulasi.....	6
2.2 <i>Animation Pipeline</i>	6
2.3 Kinematika	9
2.4 Inverse Kinematics (IK).....	9
2.5 <i>Degree of Freedom (DOF)</i>	11
2.6 Animasi 3D	11
2.7 <i>Rig</i> Karakter	12
2.8 Blender	13

2.9 Python	15
2.10 Penelitian Terkait	16
BAB III METODOLOGI PENELITIAN	17
3.1 Analisis Kebutuhan	17
3.1.1 Analisis Kebutuhan Perangkat Yang Digunakan	17
3.1.1.1 Kebutuhan Perangkat Keras	17
3.1.1.2 Kebutuhan Perangkat Lunak	18
3.1.2 Objek Penelitian	19
3.1.2.1 Analisis Gerakan Gajah	22
3.1.2.2 Analisis Derajat Sendi	23
3.1.2.3 Desain Karakter 2D	24
3.1.3 <i>Modeling</i> Karakter 3D	25
3.1.4 <i>Rigging</i> Karakter 3D	30
3.2 Penulisan Kode Program	34
3.2.1 <i>Add On</i> Blender	36
3.2.2 Data Bone	39
3.3 Perancangan Pengujian Metode <i>Inverse Kinematics</i> dan <i>Add On</i>	42
3.4 Penerapan Metode <i>Inverse Kinematics</i> dan <i>Add On</i>	44
BAB IV HASIL DAN PEMBAHASAN	47
4.1 Pra Produksi	47
4.1.1 Storyboard	47
4.2 Produksi	49
4.2.1 <i>Modeling</i>	49
4.2.2 <i>Texturing</i>	52
4.2.3 <i>Rigging</i>	53
4.2.4 <i>Animating</i>	54
4.2.5 <i>Lighting</i>	55
4.2.6 <i>Camera Operating</i>	55
4.2.7 <i>Rendering</i>	56
4.3 Pengujian <i>Add On</i>	58
4.3.1 <i>Importing Add On</i>	58
4.3.2 <i>Generate Bone</i>	61

4.3.3 <i>Rigging</i>	61
4.3.4 Pengujian <i>Add On</i> dengan Metode <i>Inverse Kinematics</i>	62
4.3.5 Pembuatan Simulasi Gerak Gajah.....	63
4.3.6 Hasil Uji Coba.....	65
4.4 Integrasi Dalam Islam	68
BAB V KESIMPULAN DAN SARAN	71
5.1 Kesimpulan	71
5.2 Saran.....	71
DAFTAR PUSTAKA	72



DAFTAR GAMBAR

Gambar 2.1 IK Solver	10
Gambar 2.2 Gerakan memukul menggunakan <i>inverse kinematics</i>	11
Gambar 2.3 Skeleton Gajah diambil dari (https://www.moving-picture.com/advertising/work/moneysupermarket-elephunk/).	13
Gambar 2.4 Contoh animasi 3D dari Blender Foundation.	14
Gambar 3.1 Diagram alur penelitian.	20
Gambar 3.2 Video gajah berjalan pada <i>software</i> Vegas 14..... (https://www.youtube.com/watch?v=iOE7dTuLCiU).	22
Gambar 3.3 Pengukuran sudut derajat gajah.....	23
Gambar 3.4 Desain kerangka 2D.....	24
Gambar 3.5 <i>Import</i> desain 2D ke blender.	25
Gambar 3.6 <i>Modeling</i> badan tampak depan.	26
Gambar 3.7 <i>Modeling</i> badan tampak samping.	27
Gambar 3.8 <i>Modeling</i> kaki tampak depan.	27
Gambar 3.9 <i>Modeling</i> kaki tampak samping.	27
Gambar 3.10 <i>Modeling</i> kepala tampak depan.....	28
Gambar 3.11 <i>Modeling</i> kepala tampak samping.....	28
Gambar 3.12 <i>Modeling</i> Belalai dan Gading Tampak Depan.	29
Gambar 3.13 <i>Modeling</i> belalai dan gading tampak samping.....	29
Gambar 3.14 Blok diagram pembuatan karakter 3D.....	30
Gambar 3.15 Blok diagram pembuatan <i>bone</i> gajah.....	31
Gambar 3.16 <i>Single bone</i>	31
Gambar 3.17 Kerangka <i>bone</i> gajah.....	32
Gambar 3.18 <i>Setting bone</i> menggunakan <i>inverse kinematics</i>	33
Gambar 3.19 Mengunci gerakan <i>bone</i> pada <i>keyframe</i>	33
Gambar 3.20 <i>Flowchart add on</i>	31
Gambar 3.21 <i>Flowchart class create</i>	31
Gambar 3.22 <i>Flowchart inverse kinematics</i>	31
Gambar 4.1 Ide cerita animasi (https://www.youtube.com/watch?v=GFKtn2LkiYE&t=28s).	47
Gambar 4.2 <i>Modeling</i> gajah.	50

Gambar 4.3 <i>Modeling environment</i>	51
Gambar 4.4 <i>Modeling prajurit</i>	51
Gambar 4.5 <i>Texture environment menggunakan node editor</i>	52
Gambar 4.6 <i>Generate bone dengan add on</i>	53
Gambar 4.7 <i>Mengombinasikan bone dengan source code</i>	53
Gambar 4.8 <i>Animating menggunakan metode inverse kinematics</i>	54
Gambar 4.9 <i>Setting lighting</i>	55
Gambar 4.10 <i>Camera operating</i>	56
Gambar 4.11 <i>Pengaturan rendering</i>	57
Gambar 4.12 <i>Proses final editing</i>	57
Gambar 4.13 <i>Membuka user preferences</i>	58
Gambar 4.14 <i>Install add on</i>	59
Gambar 4.15 <i>Open add on file</i>	59
Gambar 4.16 <i>Save user setting</i>	60
Gambar 4.17 <i>Generating bone</i>	60
Gambar 4.18 <i>Generate bone</i>	61
Gambar 4.19 <i>Add on rigging</i>	62
Gambar 4.20 <i>Add on inverse kinematics</i>	63
Gambar 4.21 <i>Simulasi Gerak Gajah</i>	64

DAFTAR TABEL

Tabel 3.1 Perangkat Keras Yang Digunakan.....	18
Tabel 3.2 Perangkat Lunak Yang Digunakan.....	18
Tabel 3.3 Rekapitulasi sudut derajat untuk diterapkan pada <i>software</i> blender.....	24
Tabel 4.1 <i>Storyboard</i>.....	48
Tabel 4.2 Koordinat <i>bone</i> kaki kanan.....	64
Tabel 4.3 Koordinat <i>bone</i> Kaki Kiri.....	65
Tabel 4.4 Pengujian simulasi gerak gajah saat berjalan.....	66
Tabel 4.5 Tingkat kesalahan derajat pada simulasi.....	68



ABSTRAK

Syaifullah, M. Bagus. **Simulasi Gerak Hewan Gajah Menggunakan Metode *Inverse Kinematics* Dalam Pembuatan Animasi 3d Asbabun Nuzul Al- Fiiil**. Skripsi. Jurusan Teknik Informatika Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing (I) Dr. M. Faisal, MT, (II) Hani Nurhayati, MT.

Kata Kunci : Animasi 3D, *Rigging*, Gajah, *Inverse Kinematics*, *Add On*.

Teknologi multimedia saat ini berkembang begitu pesat, termasuk di dalamnya adalah film animasi 3D. Dalam produksi animasi 3D ada beberapa tahapan yaitu : *modelling*, *rigging*, *texturing*, *animating*, *visual effects technician*, dan *rendering*. *Rigging* adalah salah satu tahapan dalam pembuatan animasi 3D atau *pipeline*. Proses *rigging* bertujuan untuk menggerakkan objek 3D dengan cara memasukkan *rig* kontrol ke dalam objek geometrik sehingga *animator* dapat memindahkan objek tersebut. Terdapat beberapa metode yang diterapkan dalam proses *rigging* salah satunya adalah metode *inverse kinematics*. *Inverse kinematics* adalah metode untuk menghitung nilai-nilai rotasi bersama derajat kebebasan individu melalui rotasi dan posisi yang telah ditetapkan. Penelitian ini berfokus untuk membuat *add on rigging* yang dapat di *generate* menjadi kerangka gajah dan dapat dikombinasikan dengan *add on* menggunakan metode *inverse kinematics* dan untuk mengetahui derajat kebebasan tulang hewan gajah yang dihasilkan dengan metode *inverse kinematics*. Pengujian *add on* dilakukan nilai kesalahan derajat kebebasan tulang pada simulasi gerak gajah saat berjalan. Hasil pengujian *add on Elephant Rig* yang diimplementasikan dengan metode *inverse kinematics* pada *software* blender dapat berjalan dengan baik dengan rata-rata *error* 8,03% dari 16 data. Data tersebut berupa 2 *bone* dari masing-masing kaki dari gajah dengan 2 *pose* yang membentuk pola berjalan gajah. *Add on* tersebut dapat digunakan dalam pembuatan animasi dengan objek gajah.

ABSTRACT

Syaifullah, M. Bagus. Simulation of Elephant Animal Motion Using the Inverse Kinematics Method in Making 3D Animation of Asbabun Nuzul Al-Fiil. Undergraduate Thesis. Informatics Engineering Department of Science and Technology Faculty Islamic State University Maulana Malik Ibrahim Malang. Pembimbing (I) Dr. M. Faisal, MT, (II) Hani Nurhayati, MT.

Keywords : 3D Animation, Rigging, Elephant, Inverse Kinematics, Add On.

Multimedia technology is currently developing so rapidly, including 3D animation films. In the production of 3D animation there are several stages, such as: modelling, rigging, texturing, animating, visual effects technician, dan rendering. Rigging is one of the stages in making 3D animations or pipelines. The rigging process aims to move 3D objects by entering the control rig into a geometric object so that the animator can move the object. There are several methods applied in the rigging process, one of which is the inverse kinematics method. Inverse kinematics is a method for calculating rotation values along with the degree of individual freedom through a predetermined rotation and position. This research focuses on making add on rigging that can be generated into an elephant framework and can be combined with add on using the inverse kinematics method and to determine the degree of freedom of elephant animal bones produced by this method. Testing of add on is done by the error value of bone freedom in the simulation of elephant motion while walking. The results of testing the add on Elephant Rig implemented by the inverse kinematics method on the blender software can work well with an average error of 8.03% from 16 data. The data is in the form of 2 bones from each leg of an elephant with 2 poses that form an elephant's walking pattern. These add-ons can be used in making animations with elephant objects.

ملخص

سيف الله، محمد باجوس. محاكاة حركة الحيوانات الفيل باستخدام طريقة **Inverse Kinematics** في صنع الرسوم المتحركة 3D لأسباب النزول سورة الفيل . أطروحة الجامعية. قسم هندسة المعلوماتية لكلية العلوم والتكنولوجيا في جامعة الإسلامية مولانا مالك إبراهيم مالانج. المشرف : (الأوال) الدكتور محمد فيصل ، ماجستير ، (الثاني) هاني نورحياتي، ماجستير.

كلمة الرئيسية : رسوم المتحركة 3D ، تزوير ، الفيل ، **Inverse Kinematics** ، Add On

تتطور تقنية الوسائط المتعددة حالياً بسرعة كبيرة ، بما في ذلك أفلام الرسوم المتحركة 3D. هناك عدة مراحل في إنتاج الرسوم المتحركة 3D ، مثل: النمذجة والتزوير والتركيب والرسومات وفني التأثيرات المرئية والتقديم. تعد التزوير إحدى المراحل في إنشاء رسوم المتحركة 3D أو خطوط أنابيب. تهدف عملية التزوير إلى تحريك الكائنات 3D عن طريق إدخال وحدة التحكم في كائن هندسي حتى يتمكن المتحرك من تحريك الكائن. هناك العديد من الطرق المطبقة في عملية التزوير ، واحدة منها هي الطريقة **Inverse Kinematics**. **Inverse Kinematics** هي طريقة لحساب قيم الدوران جنباً إلى جنب مع درجة الحرية الفردية من خلال دوران محدد وموضع. يركز هذا البحث على جعل الإضافة على التزوير التي يمكن أن تتولد في إطار الفيل ويمكن دمجها مع الإضافة على استخدام الطريقة **Inverse Kinematics** وتحديد درجة حرية عظام حيوانات الفيل التي تنتجها هذه الطريقة. يتم اختبار **Add On** بواسطة قيمة الخطأ لحرية العظام في محاكاة حركة الفيل أثناء المشي. نتائج اختبار **Add On** على **Elephant Rig** التي تنفذها الطريقة الحركية العكسية على برنامج الخلاط يمكن أن تعمل بشكل جيد مع وجود خطأ متوسط قدره 8,03% من 16 بيانات. البيانات في شكل 2 عظام من كل ساق من الفيل مع 2 poses يشكلان نمط سير الفيل. يمكن استخدام هذه **Add Ons** في إنشاء رسوم متحركة بكائنات الفيل.

BAB I

PENDAHULUAN

Bab pendahuluan akan membahas mulai dari alasan atau latar belakang masalah, pernyataan masalah yang diangkat, batasan masalah yang diambil, sampai dengan sistematika dalam penulisan penelitian ini.

1.1 Latar Belakang Masalah

Teknologi saat ini sangat berkembang dengan pesat seperti teknologi multimedia, termasuk film animasi 3D. Animasi 3D saat ini sangat diandalkan dalam banyak bidang seperti film, video *game*, televisi, bahkan di bidang kedokteran, arsitektur, hukum, dan forensik (beane, 2012). Dalam produksi animasi 3D ada beberapa tahapan yaitu : *modelling*, *rigging*, *texturing*, *animating*, *visual effects technician*, dan *rendering* (beane, 2012).

Proses *animating* merupakan salah satu proses dari 12 tahapan dalam pembuatan film animasi 3D. Proses *animating* adalah proses berupa rangkaian gerak pada karakter dan objek pada animasi. Proses tersebut sangat bergantung dengan *animator* yaitu orang yang mengerjakan proses ini. *Animator* bekerja untuk mencapai tujuan memberikan gerak pada animasi agar terlihat nyata untuk memberikan pengalaman menarik bagi penonton film animasi tersebut karena dari kualitas pergerakan dalam animasi sangat berpengaruh dengan proses yang akan disampaikan dengan alur dari cerita dalam sebuah film animasi 3D.

Rigging merupakan salah satu tahapan dalam pembuatan animasi 3D atau *pipeline*. Proses *rigging* bertujuan untuk menggerakkan objek 3D dengan cara

memasukkan *rig* kontrol ke dalam objek geometrik sehingga *animator* dapat memindahkan objek tersebut (beane, 2012). Namun jika gerakan karakter maupun objek dalam animasi mempunyai variasi yang sangat banyak maka akan diberikan ratusan hingga ribuan titik kontrol pada karakter dan objek tersebut. Tahapan selanjutnya dalam pembuatan animasi adalah *skinning*, *skinning* merupakan proses menyatukan antara *rig* (tulang) dengan *character*. Sehingga objek tulang maupun *controller* yang telah dibuat dapat difungsikan untuk menggerakkan objek karakter. Tugas dari pembuat *rig* tidak hanya membuat tulang saja, tujuan utama dari pembuatan tulang yaitu agar dapat meringankan pekerjaan *animator* dalam memberikan gerakan pada objek 3D dalam proses *animating* sesuai apa yang diinginkan.

Terdapat beberapa metode yang dapat diterapkan dalam menggerakkan karakter film animasi salah satu metode yang bisa dipakai dalam pergerakan animasi adalah metode *inverse kinematics*. *Inverse kinematics* adalah studi tentang bagaimana suatu hubungan struktur tulang dapat bergerak berlawanan. *Inverse kinematics* mendeskripsikan gerak hubungan dari struktur tulang. Metode *inverse kinematics* merupakan metode yang umum digunakan dalam memanipulasi nilai interaktif untuk menghasilkan suatu pola. Metode *inverse kinematics* juga banyak digunakan dalam animasi komputer, komputer grafis, teknik mesin, dan robotika. Model kerangka *invers kinematics* digunakan untuk mengontrol suatu objek agar terlihat hidup dalam dunia virtual, seperti manusia atau hewan yang sering kita lihat pada film dan video *game* (Jing Huang, 2005). Terdapat dua metode pendekatan berbasis *kinematics* yang biasa digunakan dalam memosisikan setiap hubungan tulang atau rotasi sendi dari anggota badan

yaitu disebut dengan *forward kinematics* (FK) dan *inverse kinematics* (IK) (Bhatti, 2013). Pada dasarnya, *inverse kinematics* adalah kebalikan dari *forward kinematics* yang dapat digunakan untuk mengambil satu set target akhir efektor (*end-effector*) dari DOF (*Degree of Freedom*) atau derajat kebebasan (Lei, 2013).

Inverse kinematics (IK) adalah metode yang berfungsi untuk menghitung nilai-nilai rotasi dari derajat kebebasan individu melalui posisi dan rotasi yang telah ditetapkan. Hal ini sering digunakan dalam menganimasikan bagian tulang untuk membuat sebuah gerakan yang telah ditetapkan atau sebelum perhitungan dalam *keyframe* yang dapat dilakukan secara *offline* (Bhatti, 2013) . Setiap pergerakan dalam sendi rotasi tulang memiliki derajat kebebasan (*degree of freedom*) yang bertujuan untuk mendefinisikan seberapa banyak kemungkinan pergeseran dan rotasi yang dimiliki oleh suatu objek.

Berdasarkan pernyataan di atas proses *rigging* dan *animating* adalah dua proses yang saling berhubungan. Proses-proses tersebut adalah proses yang sangat penting dalam pembuatan animasi. Proses *rigging* dan *animating* membutuhkan waktu yang tidak sedikit dalam pembuatannya. Oleh karena itu dalam penelitian ini akan dibahas tentang cara pembuatan *add on rigging* untuk *generate* bentuk kerangka tulang gajah pada *software* blender yang akan dikombinasikan dengan *add on* dengan penerapan metode *inverse kinematics* pada *software* . Dari kombinasi kedua *add on* tersebut akan dilakukan pembuatan simulasi gerak gajah berjalan. Hasil dari *add on rigging* dan pembuatan simulasi gerak gajah berjalan akan berfungsi untuk memudahkan

dalam pembuatan film animasi terutama dalam pembuatan animasi yang berbentuk hewan gajah.

1.2 Pernyataan Masalah

Berdasarkan dari latar belakang masalah maka penulis menyatakan masalah penelitian seperti berikut :

1. Bagaimana membuat *add on rigging* gajah dalam aplikasi blender ?
2. Berapa tingkat *error* dari sumbu derajat kebebasan tulang kaki gajah yang dihasilkan dengan metode *inverse kinematics* ?

1.3 Batasan Masalah

Penelitian ini berfokus pada objek penelitian seperti berikut :

1. Objek penelitian pada hewan gajah.
2. Metode yang digunakan adalah *inverse kinematics*.
3. Animasi dibuat dengan menggunakan *software Blender*.
4. Simulasi animasi berbentuk 3D.
5. Pengukuran derajat sendi hanya pada empat kaki gajah.

1.4 Tujuan Penelitian

Tujuan penelitian ini yaitu :

1. Membuat *add on rigging* yang dapat digenerate menjadi kerangka gajah dan dapat dikombinasikan dengan *add on* yang menggunakan metode *inverse kinematics*.
2. Mengetahui tingkat *error* derajat kebebasan tulang kaki gajah yang dihasilkan dengan metode *inverse kinematics*.

1.5 Manfaat Penelitian

Manfaat dari penelitian ini adalah :

1. Membantu dalam tahapan pembuatan *rigging* gajah.
2. Mempermudah dalam proses *animating* gajah saat berjalan.
3. Mempercepat dalam pembuatan film animasi yang menggunakan karakter gajah.

1.6 Sistematika Penulisan

Sistematika dan penulisan dalam laporan ini adalah sebagai berikut:

Bab I Pendahuluan : berisi latar belakang penelitian, rumusan masalah, tujuan penelitian batasan masalah dan manfaat penelitian.

Bab II Tinjauan Pustaka : berisi tentang pembahasan tentang penelitian yang telah dilakukan sebelumnya, dasar - dasar teori tentang *inverse kinematics*, animasi 3D, 3D *modelling* dan *software* blender.

Bab III Analisis dan Perancangan : berisi analisa dan rancangan pembuatan karakter dan implementasi metode *Inverse Kinematics* pada blender.

Bab IV Hasil dan Pembahasan : berisi hasil dari pengerjaan rancangan yang telah dibuat pada bab sebelumnya. Dan bahasan tentang hasil dari implementasi metode *invers kinematics* pada gerak gajah.

Bab V Kesimpulan dan Saran : berisi kesimpulan dari seluruh bagian penelitian beserta saran agar dapat membangun untuk penelitian yang lebih lanjut pada masa depan.

BAB II

TINJAUAN PUSTAKA

Bab tinjauan pustaka akan membahas tentang rujukan atau *studi literature* yang akan dipakai sebagai landasan dari penelitian ini dan juga menjelaskan penelitian-penelitian sebelumnya yang terkait dengan penelitian yang akan dilakukan.

2.1 Simulasi

Simulasi adalah suatu model pengambil keputusan yang mengacu pada kondisi sebenarnya dari suatu keadaan atau kondisi dalam kehidupan nyata tanpa harus mengalami keadaan atau kondisi tersebut (M Hasan, 2002). Simulasi juga dapat diartikan sebagai metodologi dalam melaksanakan percobaan dengan memakai model sistem dari salah satu atau beberapa sistem nyata yang dijalankan di dunia maya (Siagian, 1987)

2.2 Animation Pipeline

Animation pipeline adalah proses dalam pembuatan film animasi 3D. Terdapat proses yang lebih panjang dalam pembuatan animasi 3D, dibandingkan dengan tahapan yang dilakukan dalam pembuatan animasi 2D. Pembuatan film animasi 2D dapat dilakukan secara langsung dengan memberikan gambar pada tiap *frame*, sedangkan dalam pembuatan film animasi 3D harus menggambarkan dalam tiap *frame* yang dilakukan secara digital dalam bentuk 3D.

Terdapat 15 tahapan dalam proses pembuatan. film animasi, di antaranya :

1. Ide Cerita

Ide cerita adalah sebuah naskah cerita yang bentuknya narasi atau skrip yang akan dikembangkan berupa gambar dalam bentuk *storyboard*.

2. Rancangan Cerita

Rancangan cerita adalah gambar dari *storyboard* yang telah dibuat yang akan digabungkan dan ditambah dengan efek, suara dan musik.

3. Pewarnaan

Pewarnaan adalah proses dalam memberi warna film yang telah dibuat dalam *storyboard*. Proses pemberian warna melingkupi objek dari karakter dan kondisi dalam lembar kerja. *Output* dari proses ini berbentuk 2D.

4. Pemodelan

Modeling adalah proses pembuatan model 3D yang meliputi karakter, *asset*, dan *environment* dalam film animasi. Pembuatan objek-objek tersebut dibuat sesuai dengan skenario yang telah dibuat dari *storyboard*.

5. Tulang

Tulang adalah sebuah proses penambahan kerangka tulang atau *curve* pada model 3D yang bertujuan untuk menggerakkan model tersebut yang meliputi anggota gerak pada karakter sampai dengan ekspresi dari wajah.

6. Pemberian Tekstur

Pemberian tekstur adalah proses pemberian warna mulai dilakukan dalam bentuk model 3D, dengan *texture* dari karakteristik dari tiap objek.

7. Tata letak

Tata letak adalah proses penempatan dari kamera yang bertujuan untuk mengambil gambar yang akan ditampilkan dalam animasi.

8. Penyempurnaan Tata Letak

Penyempurnaan tata letak adalah proses lanjutan dari tata letak. Tahapan dalam proses ini adalah pengaturan suara, memberikan desain animasi untuk memberikan sebuah efek agar film menjadi menarik.

9. Animasi

Animasi adalah proses menggerakkan model karakter 3D. agar sesuai dengan *storyboard*. Proses ini dilakukan dengan memberikan gerakan berjalan, ekspresi dari wajah dan lain - lain.

10. Tokoh Pendukung

Tokoh pendukung adalah proses menambahkan objek-objek pendukung dalam animasi sekaligus melakukan proses *animate* pada tiap karakter tersebut.

11. Efek Karakter

Efek karakter adalah proses penambahan gerakan pada semua gerakan yang ada dalam karakter, seperti efek duduk, gerak rambut, dan sebagainya.

12. Penambahan Efek

Penambahan efek adalah proses penambahan efek pada film animasi, seperti efek asap, efek jejak kaki, efek ledakan. dan efek lainnya.

13. Pewarnaan objek

Pewarnaan objek adalah proses pemberian warna pada *environment*, seperti gedung, tanah, langit, dan penambahan *asset* untuk mendukung suasana yang terjadi dalam sebuah film.

14. Pencahayaan

Pencahayaan adalah proses untuk mengatur kecerahan cahaya dalam film animasi.

15. Fenyempurnaan Film

Penyempurnaan film adalah proses penyempurnaan akhir dengan memberikan efek yang memberikan kesan menarik dalam film animasi. (CGIMeetup, 2016).

2.3 Kinematika

Kinematika yang diambil dari Ecarta Dictionary dan dikutip dari Nilsson didefinisikan sebagai *study* gerak yang meliputi cabang dari fisika tentang hubungan sistem gerak tanpa referensi untuk massa dan kekuatan. Model dari sebuah *kinematics* dibangun dari segmen dan sendi dalam struktur hierarki yang disebut dengan hierarki antara *parent* dan *child* (Nilsson, 2009).

Kinematika merupakan salah satu metode untuk menggerakkan tulang kerangka karakter dalam animasi. Terdapat dua kinematika yang digunakan dalam menggerakkan karakter pada animasi yaitu *inverse kinematics* (IK) dan *forward kinematics* (FK). *Forward kinematics* dapat dilakukan dengan cara menggerakkan atau memutar sendi pada karakter hingga mendapat posisi yang diinginkan sedangkan untuk *inverse kinematics* dengan cara menggerakkan *parent* dari tulang untuk mendapatkan posisi dan sudut yang diinginkan.

2.4 Inverse Kinematics (IK)

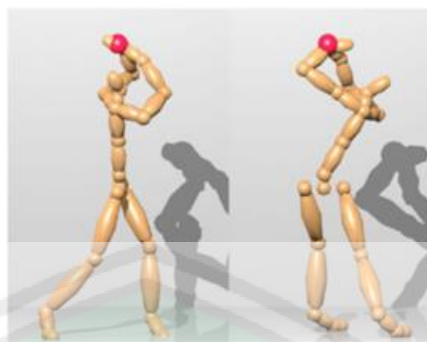
Inverse kinematics (IK) adalah salah satu metode dalam menghitung nilai-nilai rotasi bersama dengan derajat kebebasan individu melalui posisi dan rotasi yang sudah ditetapkan.

Inverse kinematics merupakan salah satu teknik dalam gerakan animasi yang sangat rumit dengan tulang dan sejumlah sendi gerak pada tulang. Perbedaan dari *forward kinematics*, *inverse kinematics* ditentukan oleh sudut akhir dari *inverse* beberapa sendi yang mendefinisikan dari gerakan tulang (Hasibuan, 2007).



Gambar 2.1 IK Solver.

Inverse kinematics memiliki sistem yaitu *animator* hanya perlu menentukan posisi akhir dari anggota sendi tulang dengan menggunakan *end-effector* dan semua rotasi sendi tulang dihitung secara otomatis untuk menempatkan sendi tulang pada tempat yang diinginkan. *Inverse kinematics* dapat memindahkan *child* tulang terakhir dalam hierarki dan semua sendi induk dari tulang akan memutar bergerak secara *inverse* atau kinematika. Sistem *inverse kinematics* berbeda dengan *forward kinematics* yang menggunakan pendekatan langsung. *Inverse kinematics* memiliki *end-effector* yang diterjemahkan langsung ke posisi yang diinginkan semua sendi rotasi dihitung dan disesuaikan secara otomatis sesuai dengan penempatan dari *end-effector* (J. Huang, 2016).



Gambar 2.2 Gerakan memukul menggunakan *inverse kinematics*.

2.5 Degree of Freedom (DOF)

Degree of Freedom (DOF) adalah jumlah derajat kebebasan sendi yang dapat memiliki terkait banyaknya dimensi sendi dalam tulang secara langsung. Animasi tiga dimensi digerakkan dengan sendi yang dapat memiliki tiga rotasi dan tiga derajat kebebasan transisi dengan rotasi dan *axis* transisi yang *orthogonal* (Nilsson, 2009).

Singkatnya tingkat kebebasan tulang menjelaskan bagaimana sendi dapat bergerak secara bebas dalam ruang tersebut. Karakteristik dari sendi dalam animasi tiga dimensi adalah sendi engsel, sendi poros, universal dan sendi bola. Sendi engsel dan sendi poros memiliki satu derajat kebebasan, sendi poros dibatasi untuk memutar sepanjang sumbu terhadap anaknya. Sendi umum didasarkan pada dua engsel sendi dengan 90 derajat perpindahan, yang berarti bahwa hal itu dapat menekuk segala arah.

2.6 Animasi 3D

Chris Broomhall menyatakan bahwa 3D adalah gambar yang memiliki kedalaman dalam sebuah komputer (Sukintaka, 2004). Animasi 3D adalah objek animasi yang berbentuk dalam ruang 3D. Animasi 3D dapat diputar dan dipindahkan seperti benda nyata. Animasi 3D juga merupakan komponen utama

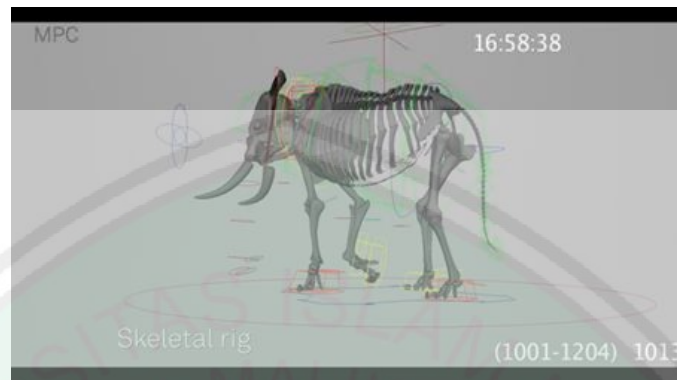
dari permainan *virtual reality*. Animasi 3D juga dapat digunakan dalam grafik presentasi untuk menambahkan kesan menarik dari presentasi tersebut. Selain dipakai dalam grafik presentasi animasi 3D juga dipakai dalam perhitungan geometri, VRML, dan pemodelan permukaan (Pcmag, 2018).

2.7 Rig Karakter

Rig karakter adalah sebuah kerangka yang berbentuk digital dan terhubung dengan objek 3D layaknya kerangka pada dunia nyata. Terdapat dua bagian dari *rig* karakter yaitu *join* dan *bone*, yang memiliki tugas masing-masing agar dapat menggerakkan objek saat proses *animating*. Gerakan tersebut yang akan di olah oleh *animator* untuk menggerakkan objek karakter. Fungsi utama dari *rig* adalah untuk memudahkan *animator* dalam memberikan gerakan dari objek karakter agar sesuai dengan keinginan. *Rig* dapat memiliki bentuk yang sederhana sampai dengan bentuk yang rumit, tergantung dari gerakan yang akan dilakukan pada objek karakter tersebut. Sebuah *rig* dapat dibuat dalam waktu singkat jika gerakan tersebut adalah sebuah gerakan yang sederhana. Namun akan memakan waktu sehari-hari bahkan berminggu-minggu jika *rig* digunakan dalam sebuah objek karakter yang memiliki gerakan yang rumit (Syalabi, 2014).

Tantangan terbesar dalam animator adalah menciptakan sebuah karakter yang bisa mengeluarkan gerakan terbaik dari karakter tersebut, namun juga mengeluarkan gerakan yang natural dan terlihat hidup. Membuat gerakan sesuai kemampuan karakter sebenarnya juga sangat penting. *Supervisor 3D*, Fabian Frank menjelaskan “Mengonsep *bone* dalam karakter merupakan suatu hal yang sangat penting dalam *rig* karakter. Sebuah *pra-visual* digunakan untuk

menentukan dasar gerakan karakter kemudian digunakan sebagai panduan untuk *rig* karakter. (VFX, 2014).



Gambar 2.3 Skeleton Gajah diambil dari (<https://www.moving-picture.com/advertising/work/moneysupermarket-elephunk/>).

2.8 Blender

Blender adalah *software* untuk membuat desain animasi 3D gratis dan open source. *Software* ini mendukung keseluruhan dari tahapan dalam pembuatan film animasi 3D yang meliputi dari simulasi, *rigging*, *animating*, *motion tracking*, *rendering*, dan *compositing* bahkan *video editing* dan juga pembuatan game. Blender juga mendukung untuk pembuatan API dalam pembuatan skrip menggunakan bahasa pemrograman Python untuk menyesuaikan aplikasi dan pembuatan *add on* yang bertujuan untuk mengembangkan aplikasi ini. Blender sangatlah cocok bagi individu maupun studio kecil yang mendapatkan manfaat dari *pipeline* dan proses pengembangan yang responsif. Blender juga sangat cocok sebagai aplikasi dalam pembelajaran animasi karena pada dasarnya blender merupakan aplikasi *open source* yang bisa digunakan dan dikembangkan secara gratis bagi pengguna aplikasi ini.



Gambar 2.4 Contoh animasi 3D dari Blender Foundation.

Blender merupakan aplikasi *cross platform* dan dapat berjalan dengan baik dalam sistem operasi seperti Linux, Windows, dan Macintosh. *User interface* atau tampilan tatap muka pada aplikasi blender menggunakan sistem OpenGL yang bertujuan memberikan tampilan yang sama pada semua pengguna aplikasi blender dalam berbagai *platform*. Pemberian sistem OpenGL yang dirancang oleh tim pengembang aplikasi blender memiliki tujuan agar aplikasi blender dapat berjalan dalam berbagai *platform*.

CEO dari Blender *Foundation* memiliki keinginan untuk memberikan fasilitas bagi tim kecil maupun perorangan dalam perkembangan animasi 3D secara gratis. Penggunaan aplikasi blender dalam semua tujuan bersifat gratis termasuk untuk tujuan pendidikan ataupun tujuan yang bersifat komersial. Kebebasan dalam penggunaan aplikasi blende sudah ditentukan oleh *General Public Licence* (GPL) yang berarti aplikasi blender gratis digunakan oleh pengguna umum. Sebagai proyek yang berbasis komunitas di bawah GNU *General License* (GPL), pengguna diberikan wewenang untuk membuat perubahan baik kecil maupun besar pada *source code* yang menambahkan

beberapa fitur baru, perbaikan *bug* yang responsif, dan kemajuan aplikasi blender.

Blender saat ini dikembangkan secara aktif oleh ratusan orang dari seluruh dunia, termasuk *animator*, seniman, ahli VFX, penggemar animasi, ilmuwan, dan masih banyak lagi. Mereka mempunyai satu tujuan yaitu memajukan jalur pembuatan animasi 3D yang memberikan fasilitas gratis dan terbuka bagi seluruh penggunanya (Foundation B., www.blender.org, 2017).

2.9 Python

Python merupakan salah satu bahasa pemrograman yang didirikan oleh Guido Van Rossum. Saat ini bahasa pemrograman python dikembangkan oleh Python Software Foundations. Python Software Foundation (PSF) adalah organisasi atau komunitas nirlaba yang mengabdikan diri untuk memajukan teknologo *open source* terkait dengan bahasa pemrograman python. Misi utama dari PSF adalah mempromosikan, melindungi, dan memajukan bahasa pemrograman python, dan bertujuan untuk mendukung serta memfasilitasi pertumbuhan komunitas *programmer* python yang beragam dari seluruh dunia. Python dikembangkan di bawah lisensi *open source* yang memiliki *OSI-approved* dan mendukung untuk dapat digunakan dan didistribusikan secara bebas, bahkan untuk tujuan komersial.

Bahasa pemrograman python memiliki pemodelan sintaks yang sangat mudah bahkan pengguna yang baru belajar bahasa pemrograman python, kemudahan sintaks yang dipakai dalam python juga dapat dibaca dan diterjemahkan dengan mudah bagi pemula. Karakteristik python yang mengutamakan kepentingan umum (*general purpose*) biasa digunakan dalam

pembuatan berbagai aplikasi seperti aplikasi web, database, dekstop GUI, aplikasi numerik, aplikasi pembelajaran, pemrograman jaringan, dan aplikasi pembuat video game.

Aplikasi blender menggunakan bahasa pemrograman python versi 3.3.0 untuk pembuatan aplikasi tersebut seperti dalam pembuatan *add on*, data blender, *user interface*, dan pengaturan dalam aplikasi baik tombol maupun peralatan tambahan untuk mengembangkan aplikasi blende. (Foundation P. , 2017).

2.10 Penelitian Terkait

Penelitian ini memiliki kesamaan pada penelitian sebelumnya yaitu penelitian dengan judul “*Simulasi Gerak Kepiting Menggunakan Metode Inverse Kinematics*” karya Muga Linggar Famukhit dimana penulis melakukan penelitian lalu membuat simulasi gerak kepiting menggunakan *software* 3Ds Max yang digunakan untuk Mengetahui derajat kebebasan tulangtulang hewan kepiting untuk diterapkan dalam karakter animasi 3D dengan memasukan informasi untuk mendapatkan hasil derajat kebebasan tulang (*degree of freedom*). Sedangkan perbedaan dari penelitian yang telah dilakukan oleh penelitian sebelumnya adalah *software* dan objek yang digunakan, penelitian tersebut menggunakan *software* 3Ds Max dan objek kepiting. Sedangkan pada penelitian ini, menggunakan *software* Blender dan objek Gajah. Diharapkan dengan dibuatnya simulasi ini dapat membantu dalam pembuatan animasi ke depannya.

BAB III

METODOLOGI PENELITIAN

Bab ini menjelaskan tentang analisa dan perancangan simulasi gerak hewan gajah menggunakan metode *inverse kinematics* dan pembuatan *add on* tulang gajah. Analisa dan perancangan simulasi dilakukan sesuai dengan kebutuhan yang bertujuan agar mempermudah dalam proses pembuatan animasi.

3.1 Analisis Kebutuhan

Bab ini menjelaskan tentang analisa dan perancangan simulasi gerak hewan gajah menggunakan metode *inverse kinematics* , dan Pembuatan *add on* tulang gajah dalam pembuatan animasi *asbabun nuzul al-fil*.

3.1.1 Analisis Kebutuhan Perangkat Yang Digunakan

Sebelum membuat simulasi gerak hewan gajah menggunakan metode *inverse kinematics*, dan Pembuatan *add on* Tulang gajah dalam pembuatan animasi *asbabun nuzul al-fil*, diperlukan beberapa peralatan komputasi guna mengimplementasikan metode *inverse kinematics* pada gerak hewan gajah.

3.1.1.1 Kebutuhan Perangkat Keras

Penelitian ini tentunya membutuhkan beberapa perangkat keras untuk pembuata *add on* dan animasi, Pada penelitian ini penulis menggunakan perangkat keras dengan spesifikasi komputer untuk membuat *add on* dan animasi, sebagai berikut :

Tabel 3.1 Perangkat Keras Yang Digunakan.

No	Perangkat Keras	Spesifikasi
1	Processor	Intel Core i7-6700HQ CPU@2.60GHz
2	VGA	Nvidia GTX950-4GB
3	RAM	8 GB DDR4
4	HDD	1TB HDD 7200 RPM
5	Mouse	Logitech
6	Keyboard	On

Spesifikasi perangkat keras tersebut tidak menjadi acuan minimal untuk membuat *add on* dan animasi, namun spesifikasi perangkat keras tersebut adalah spesifikasi perangkat keras yang digunakan oleh dalam membuat sebuah *add on* dan animasi.

3.1.1.2 Kebutuhan Perangkat Lunak

Selain perangkat keras yang disebutkan pada sub bab sebelumnya penelitian ini juga memerlukan beberapa perangkat lunak dalam membuat *add on* dan animasi. Beberapa perangkat lunak yang digunakan oleh penulis untuk membuat *add on* *elephant rig* dan pembuatan film animasi adalah sebagai berikut ini :

Tabel 3.2 Perangkat Lunak Yang Digunakan.

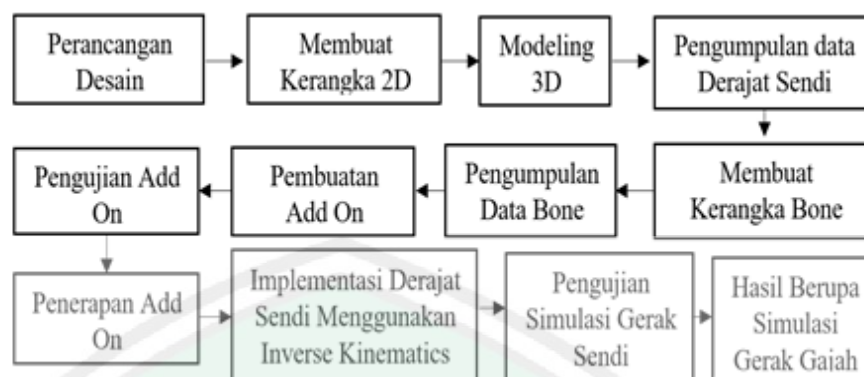
No	Software	Fungsi
1	Windows 10	Sistem Operasi
2	Blender	Desain 3D
3	Photoshop CS 6	Desain 2D
4	Vegas 14	Video Editing Analisis gerak kaki gajah

Pada Tabel 3.2 tersebut dijelaskan beberapa perangkat lunak yang digunakan oleh penulis. aplikasi blender berfungsi untuk membuat *add on*, selain pembuatan *add on* dilakukan pembuatan

model kerangka gajah serta penulisan kode program untuk *addon*. Photoshop berfungsi untuk pembuatan *texture* objek dan pembuatan desain karakter 2D. Vegas 14 berfungsi sebagai proses *final editing* untuk membuat hasil akhir dari film animasi *asbabun nuzul al-fil*.

3.1.2 Objek Penelitian

Tahap pertama yang dilakukan adalah mengambil data penelitian yang berasal dari desain gambar 2D dari karakter gajah. Data gambar yang diambil merupakan gambar gajah dari tampak depan dan tampak samping. Dari gambar tersebut akan dibuat karakter gajah 3D. Gambar 2D tersebut diambil dari gambar hewan gajah dari gambar asli Gajah Asia. Setelah itu penulis mengamati gerakan gajah yaitu bertujuan untuk mengetahui pola pergerakan gajah saat berjalan yang berfokus pada kakinya. Pengamatan dilakukan dengan cara mengambil sampel video gajah dengan durasi berkisar 30 detik. Hasil pengambilan video dimasukkan dalam *software* Vegas pro selanjutnya dilakukan analisa pergerakan gajah melalui *frame* di dalam Vegas. Selanjutnya data diperoleh dari beberapa gambar yang diambil dari vegas dan diterapkan pada *bone* yang akan dibuat ke dalam versi karakter 3D. Kumpulan *bone* tersebut yang nantinya dijadikan sebagai pembuatan gerak gajah saat berjalan menggunakan implementasi dari metode *inverse kinematics*.



Gambar 3.1 Diagram alur penelitian.

a. Perancangan Desain

Tahap selanjutnya yaitu perancangan desain. Perancangan desain simulasi gerak Gajah, dimulai dengan mengumpulkan data yang dibutuhkan dalam pembuatan simulasi gerak gajah saat berjalan, berdasarkan dari studi literature yang dijadikan rujukan.

b. Membuat Kerangka 2D

Tahapan selanjutnya yaitu pembuatan kerangka 2D yang akan dijadikan ke dalam model karakter 3D.

c. *Modeling* 3D

Setelah pembuatan kerangka 2D tahapan selanjutnya melakukan *modeling* 3D dengan bahan kerangka 2D yang sebelumnya telah dibuat.

d. Analisis Derajat Sendi

Setelah itu melakukan pengukuran derajat sendi pada 4 kaki gajah untuk mengetahui *degree of freedom* dari gerakan gajah saat berjalan melalui video yang telah didapat.

e. Membuat Kerangka *Bone*

Setelah itu membuat *bone* sesuai dengan objek yang telah dibuat serta mengacu berdasarkan studi literature yang telah dilakukan.

f. Pengumpulan Data *Bone*

Setelah itu mengumpulkan data dari *bone* yang telah dibuat. Data *bone* meliputi nama *bone*, koordinat *head*, *tail* dari *bone*, *parent bone*, koneksi antar *bone*, dan *roll bone*.

g. Pembuatan *Add On*

Setelah itu dilakukan pembuatan *add on Elephant Rig* yang diambil dari data *bone* yang sudah dibuat sebelumnya dengan menggunakan *text editor* pada software blender.

h. Pengujian *Add On*

Setelah *add on* selesai dibuat maka selanjutnya menguji fungsi dari *add on* tersebut pada software blender 2.78.

i. Implementasi Derajat Sendi Menggunakan *Inverse Kinematics*

Setelah itu mengimplementasikan data yang diperoleh dari analisis derajat sendi menggunakan metode *inverse kinematics* pada *bone* yang sudah dibuat.

j. Pengujian Simulasi Gerak Sendi

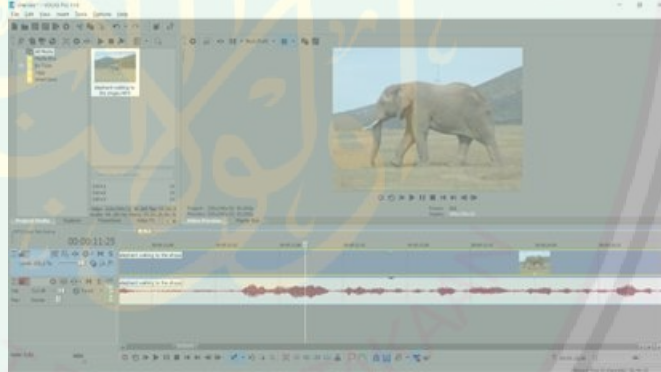
Pengujian dilakukan dengan beberapa responden dari mahasiswa yang telah lulus matakuliah Grafika Komputer, orang yang bekerja dalam bidang animasi, atau dosen yang pernah mengajar animasi.

k. Hasil Berupa Simulasi Gerak Gajah

Hasil akhir berupa simulasi pola gerak Gajah berjalan (*walk cycle*).

3.1.2.1 Analisis Gerakan Gajah

Tahapan analisa dilakukan dengan cara mengamati gerakan gajah yaitu bertujuan untuk mengetahui pola pergerakan gajah saat berjalan yang berfokus pada kakinya. Pengamatan dilakukan dengan cara mengambil sampel video gajah dengan durasi berkisar 30 detik. Pengambilan hanya dilakukan pada 1 gerakan gajah saat berjalan (*walk cycle*). Acuan video dimasukkan dalam program Vegas 14. Setelah itu dilakukan pengambilan gambar guna mengukur derajat sendi kebebasan tulang dari gajah. Hasil dari pengamatan merupakan *pose* dan sumbu derajat kebebasan tulang pada gajah saat berjalan.



Gambar 3.2 Video gajah berjalan pada *software* Vegas 14 (<https://www.youtube.com/watch?v=iOE7dTuLCiU>).

Hasil yang didapat berupa beberapa gambar yang menunjukkan pola gajah berjalan (*walk cycle*) secara berurutan. Hasil gambar tersebut yang akan dijadikan bahan sebagai pengukuran sumbu derajat kebebasan tulang dari empat kaki gajah.

3.1.2.2 Analisis Derajat Sendi

Analisis derajat sendi bertujuan untuk mengetahui derajat kebebasan tulang (*degree of freedom*) dari hewan gajah. Pengukuran derajat kebebasan tulang dengan menggunakan busur derajat sebagai alat ukur. Pengukuran derajat kebebasan tulang tersebut dilakukan pada sendi bagian kaki hewan gajah. Pengukuran sendi meliputi sendi engsel atas dan bawah serta pergelangan kaki pada dua kaki gajah. hal tersebut dikarenakan struktur tubuh gajah berbentuk simetris antara sisi kanan dan kiri. Pengukuran pada hewan gajah dilakukan dengan mengukur sudut derajat pangkal kaki dari sumbu koordinat datar 0° atau 180° , kemudian kaki gajah bergerak ke arah depan dan belakang. Pengukuran derajat hewan gajah tersebut ditujukan sebagai pengaturan sendi putar *inverse kinematics* pada *software* Blender.



Gambar 3.3 Pengukuran sudut derajat gajah.

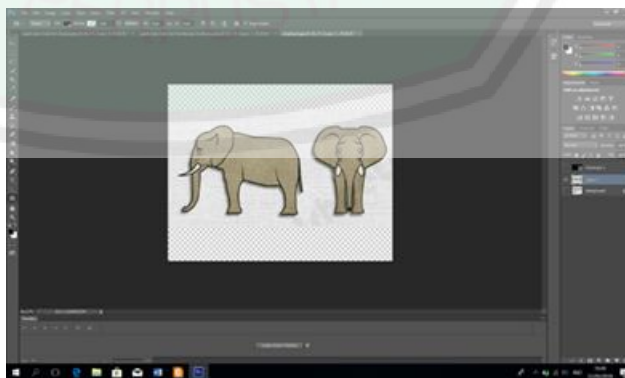
Pengukuran dilakukan dengan menganalisa gerakan Gajah saat berjalan dari video yang didapatkan dengan cara Screen Shoot video dimasukkan di dalam aplikasi Photoshop CS 6 dan di ukur dengan penggaris busur.

Tabel 3.3 Rekapitulasi sudut derajat untuk diterapkan pada *software* blender.

No	Kaki		Pose	Sudut Derajat
	Depan	Belakang		
1	Sendi Atas		1	50
2	Sendi Bawah		1	70
3	Sendi Atas		2	40
4	Sendi Bawah		2	0
5	Sendi Atas		3	0
6	Sendi Bawah		3	0
7	Sendi Atas		4	20
8	Sendi Bawah		4	20
9		Sendi Atas	1	25
10		Sendi Bawah	1	0
11		Sendi Atas	2	20
12		Sendi Bawah	2	0
13		Sendi Atas	3	20
14		Sendi Bawah	3	0
15		Sendi Atas	4	30
16		Sendi Bawah	4	0

3.1.2.3 Desain Karakter 2D

Desain karakter 2D yang dipakai oleh penulis dalam penelitian ini adalah kerangka 2D berbentuk karakter hewan gajah. Untuk mempermudah dalam pembuatan kerangka 3D. Desain 2D akan dibuat dalam gambar tampak depan dan samping.



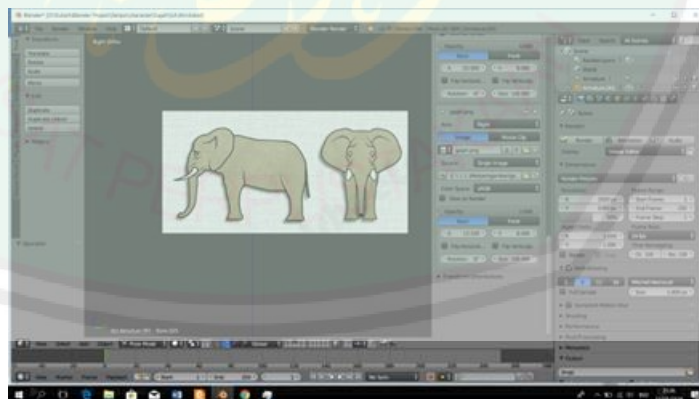
Gambar 3.4 Desain kerangka 2D.

Desain kerangka 2D dibuat dengan menggunakan *software* Photoshop CS6. Kerangka dari desain 2D diambil dari internet

dan merubah gambar tersebut agar terlihat dari depan dan samping. Desain kerangka karakter 2D dilakukan dengan menyesuaikan dari kebutuhan untuk membuat karakter gajah yang berbentuk 3D dengan cara membuat kerangka karakter gajah 2D seperti tampak dari depan dan tampak dari samping. Tahapan selanjutnya yaitu melakukan *ekspor* gambar dari desain kerangka karakter gajah 2D ke dalam format PNG atau JPG untuk mempermudah melakukan *import* pada aplikasi blender.

3.1.3 Modeling Karakter 3D

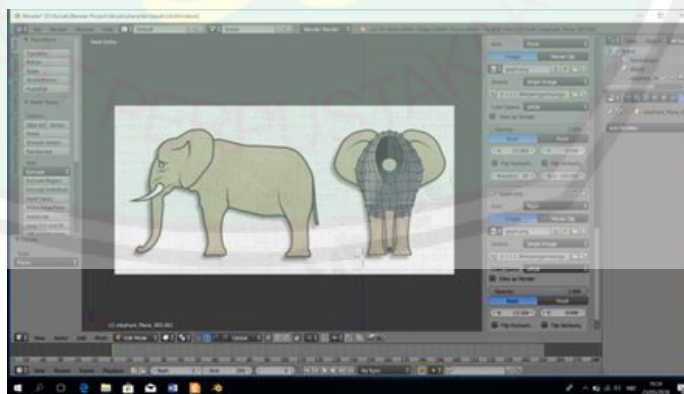
Pembuatan *Modeling* karakter 3D dilakukan dengan menggunakan aplikasi blender. Langkah pertama yang dilakukan dalam pembuatan model karakter gajah 3D adalah *import* desain kerangka 2D yang sebelumnya sudah diekspor menjadi gambar PNG atau JPG dari Photoshop CS6.



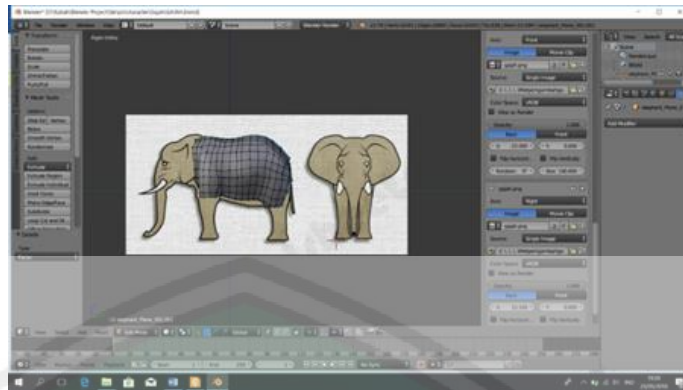
Gambar 3.5 *Import* desain 2D ke blender.

Proses pembuatan karakter 3D dilakukan dengan cara *tracing* pada gambar kerangka karakter 2D yang telah dimasukkan ke dalam aplikasi blender. Ukuran gambar *tracing* disesuaikan dengan merubah *size* dan menetapkan koordinat lokasi dari gambar tersebut . Pembuatan karakter

3D mempunyai 3 sumbu dalam halaman kerjanya yaitu sumbu x, y, dan sumbu z. Tujuan utama pembuatan kerangka 2D dengan gambar karakter gajah tampak depan dan samping yaitu untuk mempermudah dalam pembuatan karakter 3D dengan sudut pandang *orthogonal view* sehingga *modeler* dapat bekerja dengan cara *tracing* model berdasarkan sumbu x,y dan z. Keunikan dalam dunia desain 2D maupun 3D adalah saat pembuatan suatu karya setiap desainer memiliki cara tersendiri, meskipun memiliki hasil akhir yang sama. Dalam proses pembuatan karakter 3D, ada desainer yang membuat model kepala terlebih dahulu dengan cara *tracing*, ada juga yang terlebih dahulu membuat bagian badan dahulu. Penulis mendahulukan model dari badan dalam proses pembuatan *modeling* karakter ini. Proses pembuatan karakter 3D dilakukan dengan *tracing* pada bagian depan dari badan gajah selanjutnya dilanjutkan dengan melihat dari tampak samping, sehingga karakter memiliki ketiga sumbu x, y, dan z.



Gambar 3.6 *Modeling* badan tampak depan.

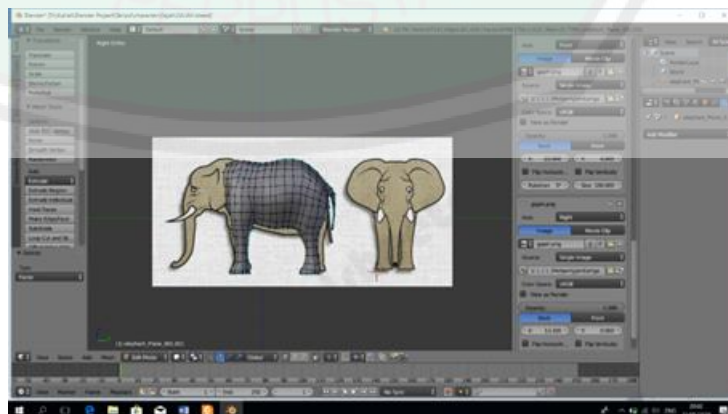


Gambar 3.7 *Modeling* badan tampak samping.

Tahap selanjutnya setelah bagian tubuh karakter selesai dibuat, dilanjutkan dengan membuat bagian kaki dari karakter gajah. Ukuran dari kaki menyesuaikan dengan kerangka 2D yang telah dibuat sebelumnya.

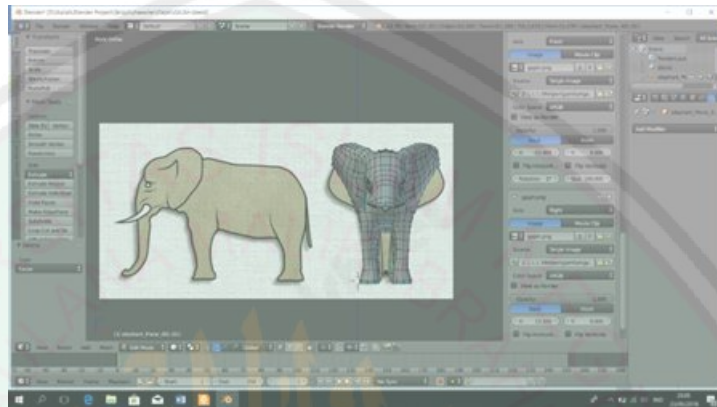


Gambar 3.8 *Modeling* kaki tampak depan.

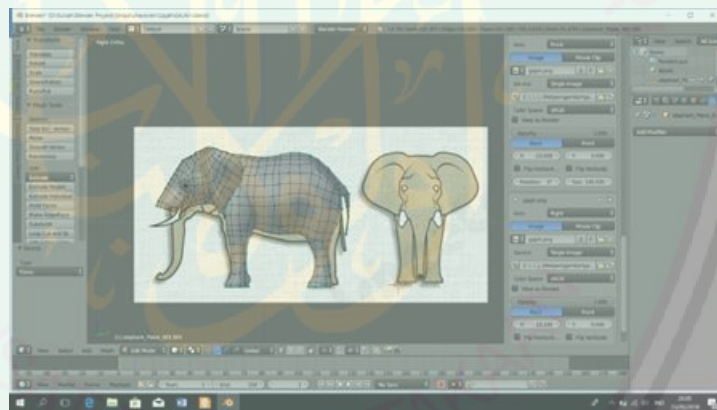


Gambar 3.9 *Modeling* kaki tampak samping.

Tahapan selanjutnya yaitu membuat model kepala karakter gajah dengan cara *tracing* tampak depan dan tampak samping. Pembuatan kepala dilakukan sedemikian rupa sehingga bentuk dari kepala dalam proses *modeling* karakter 3D menyerupai dengan kerangka 2D.

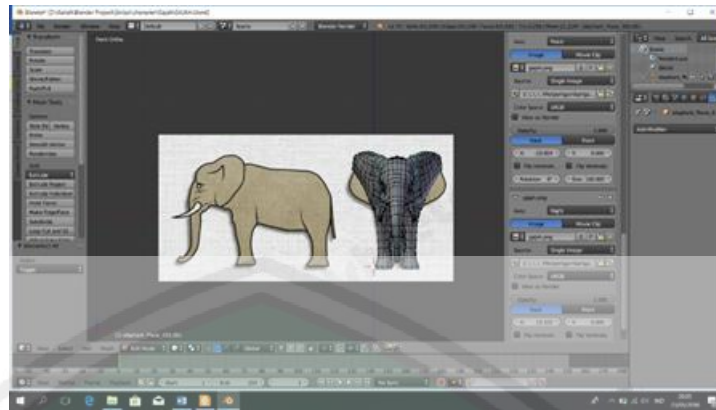


Gambar 3.10 *Modeling* kepala tampak depan.

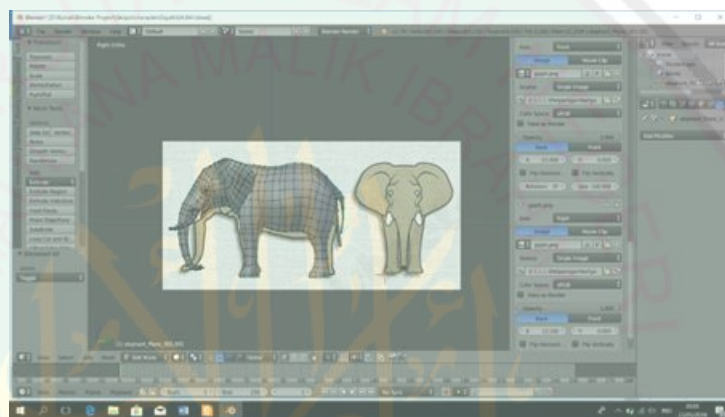


Gambar 3.11 *Modeling* kepala tampak samping.

Tahapan terakhir pada proses *modeling* adalah membuat bagian belalai dari gajah, sekaligus dilakukakn penyempurnaan *modeling* agar karakter gajah 3D terlihat menarik. Penyempurnaan dilakukan dengan cara memperhalus dengan menambahkan *add modifier* sehingga permukaan objek karakter gajah menjadi lebih halus dan tampak menarik.

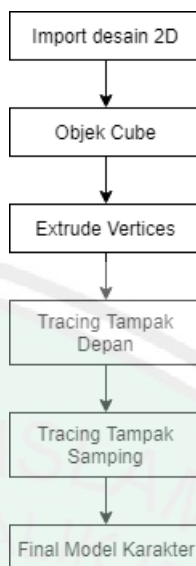


Gambar 3.12 *Modeling* Belalai dan Gading Tampak Depan.



Gambar 3.13 *Modeling* belalai dan gading tampak samping.

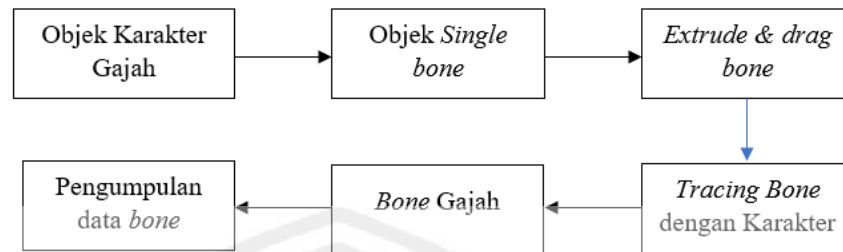
Bentuk ukuran yang dibuat dalam proses membuat karakter animasi yang dimulai dari tahap pembuatan desain kerangka karakter 2D sampai dengan pembuatan *modeling* karakter 3D tentunya pasti tidak persis dengan ukuran karakter gajah di dunia nyata. Terdapat sebuah asas ketidakpastian atau *uncertainty principle* dalam sebuah ilmu fisika. Asas tersebut dikenalkan oleh seseorang bernama Warner Heisenberg yang dikenal sebagai seorang ilmuwan fisikan dari Jerman. Prinsip asas ini menjelaskan tentang kemustahilan manusia dalam mengetahui ukuran posisi dan kecepatan dari suatu material dengan akurasi yang tepat (Hilgevoord, 2006).



Gambar 3.14 Blok diagram pembuatan karakter 3D.

3.1.4 Rigging Karakter 3D

Proses *rigging* karakter merupakan tahapan yang akan membuat sebuah model karakter terlihat hidup dengan pergerakan dilakukan oleh *animator*. Untuk dapat menggerakkan sebuah model karakter, *animator* juga membutuhkan *setup controller* tulang dan *manipulator* pada obyek karakter. Pada tahapan rigging karakter peneliti menggunakan objek *single bone* yang ada pada *software* Blender. *Single bone* pada Blender berfungsi untuk membuat 1 *bone*. *Bone* tersebut akan dibentuk sesuai dengan objek gajah yang telah dibuat sebelumnya sehingga objek tersebut dapat bergerak saat proses *animating*. Proses selanjutnya adalah membuat struktur hubungan antar tulang sesuai dengan desain struktur yang telah dibuat. Setelah dilakukan pembuatan *bone* gajah, akan dilakukan pengambilan data *bone* guna pembuatan *add on*. Data *bone* berupa nama *bone*, koordinat dari *head* dan *tail bone*, *parent bone*, dan koneksi *bone*.



Gambar 3.15 Blok diagram pembuatan *bone* gajah

Objek Karakter Gajah

Bertujuan untuk mempermudah dalam pembuatan *bone* yang berfungsi untuk *tracing* saat memodifikasi *single bone*.

Objek *single bone*

Bahan dasar yang disediakan oleh *software* Blender untuk pembuatan *bone* yang berfungsi dalam menggerakkan objek saat proses *animating*.

Extrude & drag bone

Salah satu cara dalam memodifikasi objek dalam *software* Blender yang berfungsi untuk menyesuaikan *primitive object* agar sesuai dengan keinginan *animator*.

Bone Gajah

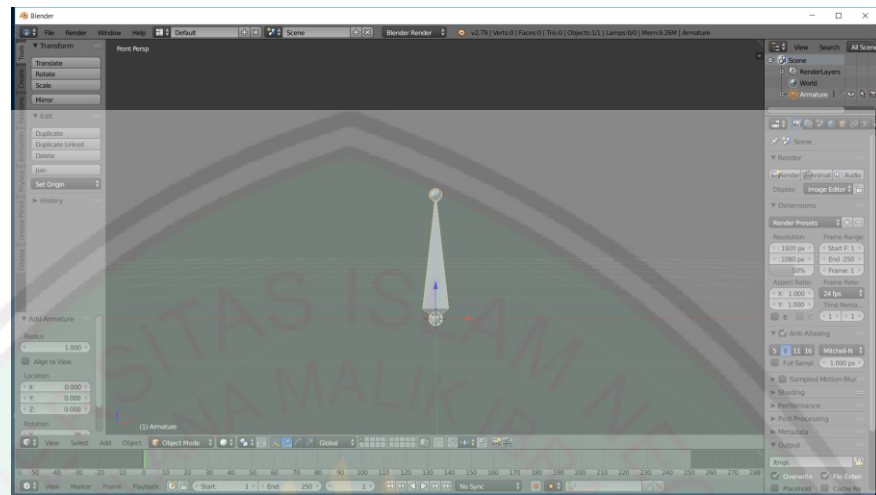
Hasil akhir dalam pembuatan *bone* yang bertujuan untuk dijadikan *add on* dan berfungsi sebagai objek penggerak karakter gajah.

Pengumpulan data *bone*

Bertujuan untuk data dalam pembuatan *add on*.

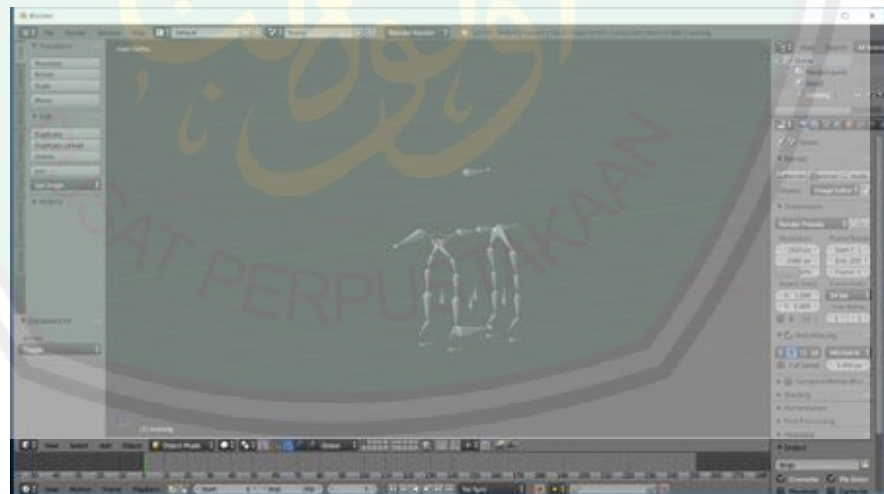
Setelah pembuatan karakter 3D langkah selanjutnya yaitu pembuatan *bone* gajah atau bisa disebut dengan *armature*. Tahapan

pertama dalam pembuatan *bone* yaitu dengan membuat *single bone* yang sudah ada dalam *software* blender.



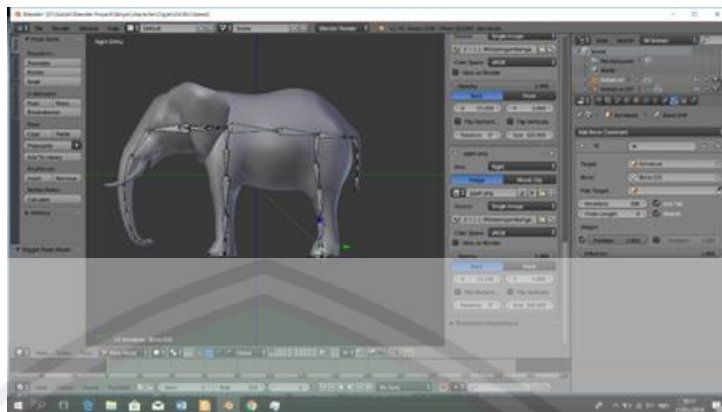
Gambar 3.16 *single bone*

Selanjutnya memodifikasi *single bone* sesuai dengan karakter gajah yang telah dibuat sebelumnya dengan cara *extrude & drag* pada *edit mode* sehingga *single bone* menyerupai bentuk karakter gajah.



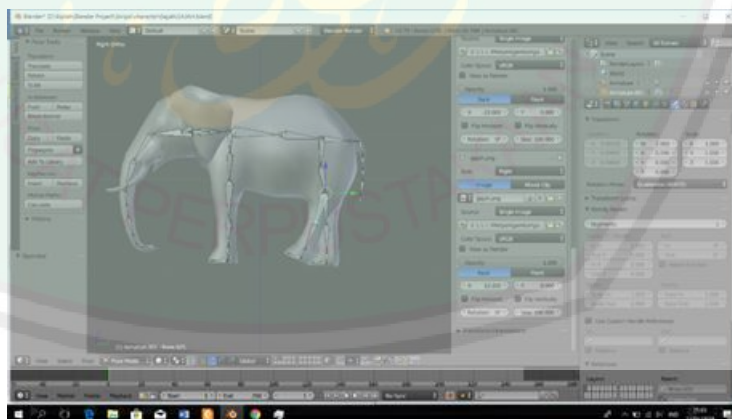
Gambar 3.17 Kerangka *bone* gajah.

Proses berikutnya adalah implementasi susunan struktur tulang menggunakan *inverse kinematics* pada model gajah. Pemberian tulang dilakukan dengan satu bidang pada sisi kanan kemudian digandakan dan di-*mirror* agar simetris antara tulang kiri dan kanan.



Gambar 3.18 Setting bone menggunakan *inverse kinematics*.

Tahapan selanjutnya adalah pengaturan tulang (*setup bone*) menggunakan metode *inverse kinematics*. Pengaturan tulang model Gajah pada *software* blender, peneliti menggunakan *tracking inverse kinematik* pada *bone constraint* di *software* blender. Penerapan *inverse kinematik* dilakukan pada model tulang kaki yaitu dengan cara memilih *tracking inverse kinematics* pada *bone constraint* dilanjutkan dengan menyeleksi parent pada tulang tersebut.



Gambar 3.19 Mengunci gerakan *bone* pada *keyframe*.

Proses *rigging* karakter dilanjutkan dengan penerapan ukuran derajat kebebasan tulang (*degree of freedom*) pada bagian pangkal sendi model gajah. Penerapan sudut derajat kebebasan tulang berdasarkan hasil analisis yang telah peneliti lakukan sebelumnya. Penerapan sudut derajat

kebebasan tulang menggunakan *tools* Armature pada *software* Blender, dengan pengaturan pada sisi tulang dilakukan pada bagian ke empat kaki Gajah. Setelah pembuatan 1 kali gerakan Gajah berjalan lalu dilakukan dengan *shape keys* sehingga membentuk sebuah pola *walk cycle*.

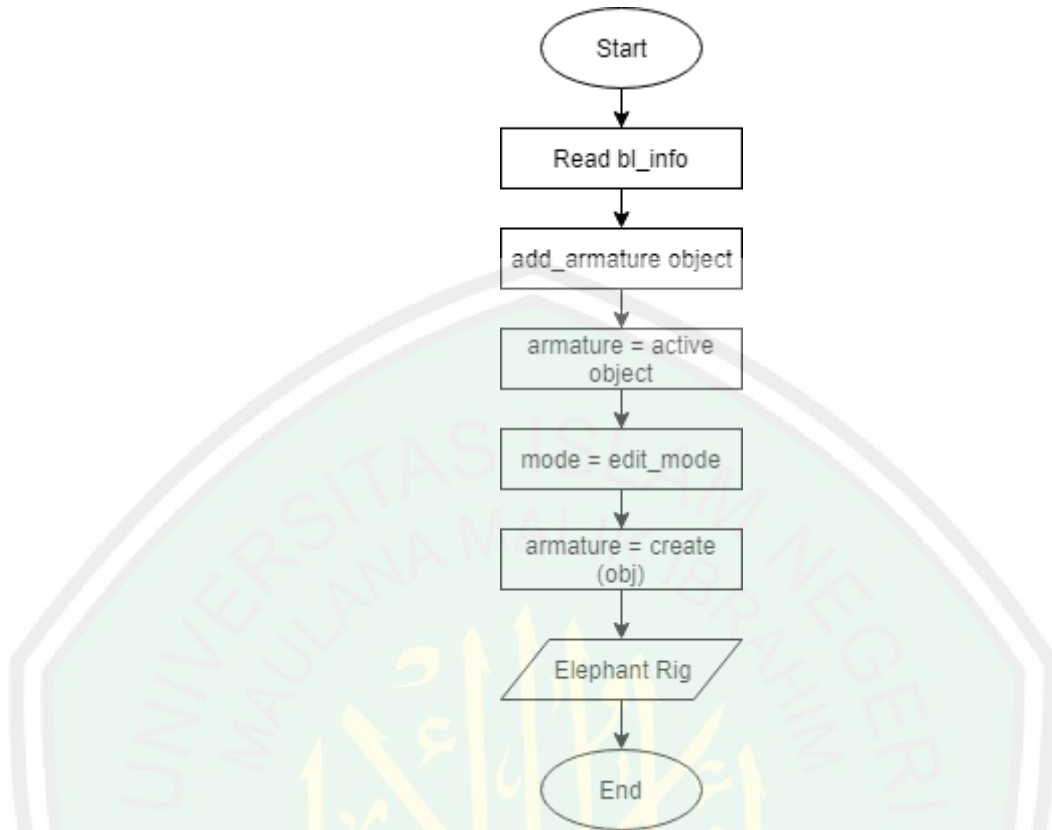
3.2 Penulisan Kode Program

Aplikasi blender memiliki sebuah fitur untuk tujuan mengembangkan aplikasi tersebut. Fitur tersebut yaitu *text editor* yang berfungsi untuk memudahkan *developer* dalam pengembangan aplikasi tersebut. *Text editor* dalam aplikasi blender memiliki fungsi Fungsi seperti *text editor* lainnya seperti sublime dan notepad++. Seluruh tahapan penulisan kode program untuk pembuatan *add on* dilakukan dalam aplikasi blender menggunakan *text editor*.

Penulisan kode dalam proses pembuatan *add on*, sebenarnya bisa dilakukan dengan menggunakan diluar aplikasi blender menggunakan *text editor* lain, namun *text editor* pada aplikasi blender memiliki keunggulan yaitu memiliki fitur *running script* sehingga *source code* yang ditulis dalam *text editor* dalam aplikasi blender dapat langsung dijalankan dan mengetahui hasil dari *source code* yang telah dibuat.

Sebelumnya sudah ada *add on* yang bisa generate *bone* manusia yaitu *rigify*. Penulis akan mengembangkan *add on* tersebut untuk generate *bone* gajah.

Berikut ini merupakan rincian dari alur penulisan *source code* dalam proses pembuatan *add on* blender yang bertujuan untuk generate *bone* gajah yang sudah dibuat sebelumnya.



Gambar 2.20 *Flowchart add on*

Add on pada blender bekerja seperti pada gambar diatas

Read bl_info

Berfungsi untuk membaca identitas dari *add on*, mulai dari identitas sampai dengan memanggil fungsi dari *add on*.

Add_armature object

Berfungsi untuk membuat satu objek *armature* atau *bone*.

Armature = active_object

Berbeda dengan *primitive object*, *armature object* adalah suatu objek non aktif dalam blender. Sebelum memodifikasi atau merubah *armature*, *armature object* harus dirubah menjadi *active object*

Mode = edit_mode

Berfungsi untuk merubah lembar kerja pada blender menjadi *edit mode*. Dalam blender terdapat beberapa *mode* untuk pembuatan animasi seperti *object mode*, *pose mode*, *edit mode* dan lain-lain. Fungsi merubah lembar kerja menjadi *edit mode* adalah untuk dapat merubah suatu objek seperti *primitive object* dan *armature object*.

Armature = create (obj)

Berfungsi untuk memodifikasi *armature object* seperti data yang ada dalam *class create (obj)* yang nantinya akan dibahas pada bab selanjutnya.

Elephant Rig

Output dari *add on* ini adalah sebuah *rigging* gajah.

3.2.1 Add On Blender

Langkah pertama dalam penulisan *source code* proses pembuatan *add on* adalah membuat identitas data dari *add on* yang akan dibuat. Identitas data tersebut terdiri dari nama *add on*, *developer*, versi dari *add on*, versi blender yang dapat menggunakan *add on* tersebut, dan lain-lain.

```
bl_info = {
    "name": "Add Elephant Rig",
    "author": "M Bagus Syaifullah",
    "version": (1, 0),
    "blender": (2, 75, 0),
    "location": "View3D > Add > Armature > New Object",
    "description": "Adds a Elephant Rig IK",
    "warning": "",
    "wiki_url": "",
    "category": "Rigging",
}
```

Selanjutnya, dilakukan *import* modul-modul yang dibutuhkan dalam proses pembuatan *add on*.

```
import bpy
from bpy.types import Operator
from bpy_extras.object_utils
import AddObjectHelper, object_data_add
```

Sebelum mengambil sampel dari data *bone* gajah, untuk membuat sebuah *add on* yang dapat *generate bone* tersebut memerlukan penulisan *source code* yang berfungsi sebagai operator untuk mengintruksikan pada aplikasi blender bahwa *source code* yang dibuat merupakan sebuah *add on*, agar aplikasi blender dapat memunculkannya pada menu sesuai dengan yang dibuat dala *source code* untuk memunculkan tempat *add on* tersebut muncul, sehingga pengguna dapat menggunakan *add on* tersebut.

```
class OBJECT_OT_add_object(Operator, AddObjectHelper):
    bl_idname = "armature.elephant_bone"
    bl_label = "Add Elephant"
    bl_options = {'REGISTER', 'UNDO'}

    def execute(self, context):
        add_object(self, context)
        return {'FINISHED'}
```

Fungsi dari operator dalam aplikasi blender berisi tentang beberapa aktifitas modifikasi objek yang diperlukan untuk pembuatan *add on*. Penulisan dalam operator ditulis dengan menggunakan id name, label serta *execution function* yang bertujuan untuk memunculkan fungsi perintah pada aplikasi blender.

Selanjutnya yaitu *function definition* yang berfungsi untuk mendefinisikan kategori dari *add on* dan memunculkannya pada bar sesuai dengan lokasi dan kriteria objek dari *add on*. Dari fungsi

tersebutlah yang nantinya akan dilanjutkan dengan pemberian data dari *bone* yang telah didapatkan.

```
def add_object(self, context):
    # Add armature object
    bpy.ops.object.armature_add()
    obj = context.active_object
    obj.name = "metarig"
    obj.data.name = "metarig"

    # Remove default bone
    bpy.ops.object.mode_set(mode='EDIT')
    bones = context.active_object.data.edit_bones
    bones.remove(bones[0])

    # Create metarig
    create(obj)

    # Rotation Bone
    bpy.context.object.rotation_euler[0] = -1.5708
    bpy.ops.object.mode_set(mode='OBJECT')

    return {'FINISHED'}
```

Tahapan selanjutnya yaitu membuat *source code* yang mempunyai fungsi untuk menampilkan *button* dari *add on* pada *layout* dalam aplikasi blender sehingga blender dapat menjalankan perintah dari *source code* syang telah dibuat sebelumnya.

```
def menu_entries(menu, context):
    menu.layout.operator(OBJECT_OT_add_object.bl_idname,
        text="Elephant Rig", icon='POSE_HLT')
```

Langkah terakhir yaitu menuliskan *source code register* yang berfungsi untuk memberikan informasi pada blender perintah mana saja yang dapat diaktifkan atau di non aktifkan oleh blender.

```

def menu_entries(menu, context):
    menu.layout.operator(OBJECT_OT_add_object.bl_idname,
        text="Elephant Rig", icon='POSE_HLT')

def register():
    bpy.types.INFO_MT_armature_add.append(menu_entries)
    bpy.utils.register_class(OBJECT_OT_add_object)

def unregister():
    bpy.utils.unregister_class(OBJECT_OT_add_object)
    bpy.types.INFO_MT_armature_add.remove(menu_entries)

if __name__ == "__main__":
    register()

```

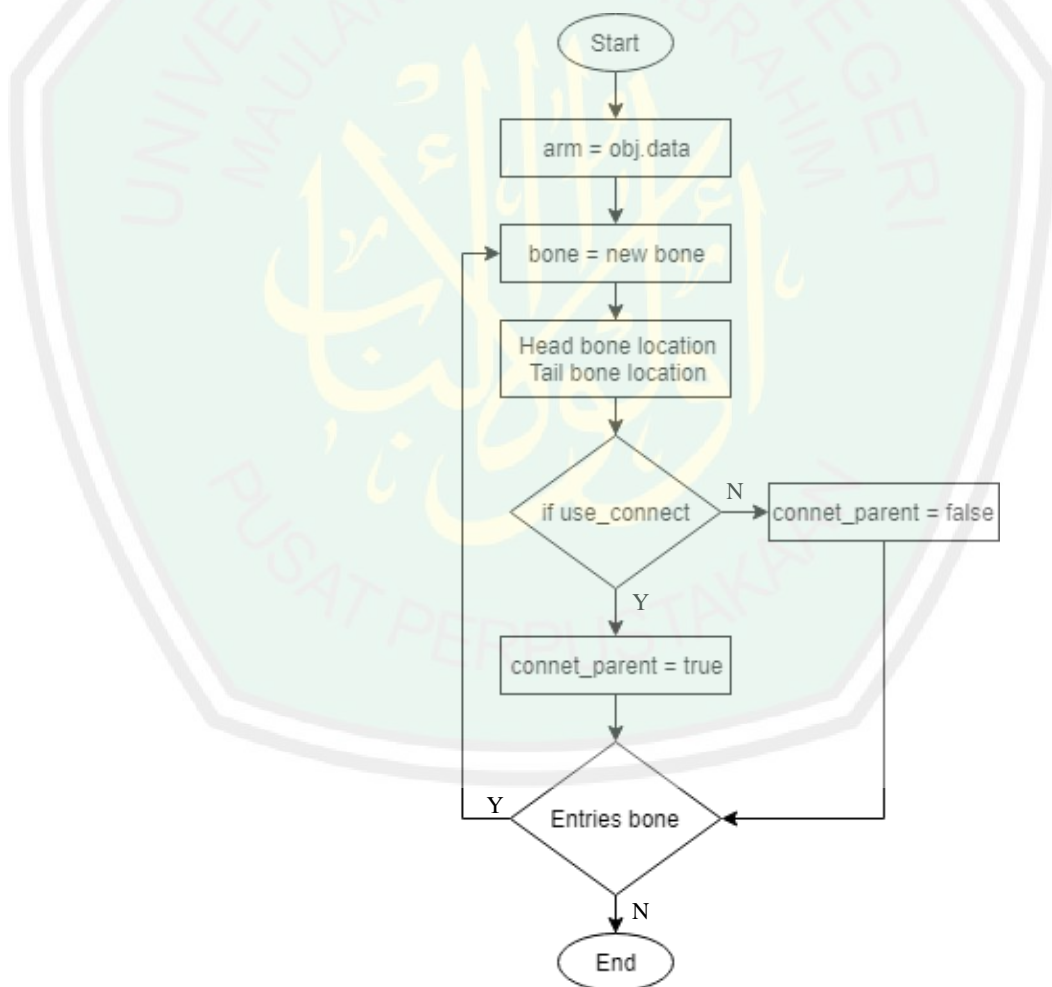
Model *bone* pada halaman kerja dalam aplikasi blender sebenarnya adalah hasil dari *compile* dari perintah-perintah yang ada dalam modul dan *add on* blender. Modul dan *add on* tersebut menggunakan bahasa pemrograman python 3.3.0. Oleh karena itu dalam pembuatan *add on* memerlukan sebuah referensi berupa model *bone* dari halaman kerja blender. Pembuatan *add on* dengan cara menuliskan *source code* secara langsung pada *text editor* sangatlah tidak mungkin, karena pembuatan *add on* memerlukan sebuah value data *bone* yang akan dibuat.

3.2.2 Data Bone

Sebuah model *bone* pada halaman kerja dalam aplikasi blender sebenarnya adalah hasil dari *compile* dari perintah-perintah yang ada dalam modul dan *add on* blender. Modul dan *add on* tersebut menggunakan bahasa pemrograman python 3.3.0. Oleh karena itu dalam pembuatan *add on* memerlukan sebuah referensi berupa model *bone* dari halaman kerja blender. Pembuatan *add on* dengan cara menuliskan *source code* secara langsung pada *text editor* sangatlah tidak mungkin, karena pembuatan *add on* memerlukan sebuah value data yang meliputi

koordinat *head* dan *tail* ataupun *parent* dari *bone* tersebut yang menghubungkan *bone* satu dengan yang lainnya.

Object *bone* gajah yang terbentuk dari beberapa *single bone* yang terhubung antara satu dengan yang lainnya, jika kita menyeleksi salah satu *bone*, maka *bone* tersebut akan menunjukkan nama *bone*, 6 titik koordinat dari *head* dan *tail* yang dipengaruhi oleh sumbu XYZ, *parent*, dan derajat *roll* dari *bone* tersebut. Data tersebut yang nantinya akan dijadikan acuan untuk pembuatan *add on*.



Gambar 3.21 Flowchart class crate (obj)

Arm = obj data

Berfungsi untuk membuat variabel *arm* menjadi objek data.

Bone = new.bone

Berfungsi untuk membuat *bone* baru dalam pembuatan sebuah *rig*.

Head bone location dan *tail bone location*

Berfungsi untuk merubah koordinat dari kepala dan ekor dari *single bone* yang dibuat sehingga dapat membentuk sebuah *rig*.

Use_connect

Berfungsi untuk koneksi antar *bone* tersebut dengan *parent bone* tersebut. Jika *bone* tersebut terkoneksi maka *bone* tersebut tidak dapat terpisah dengan *parent bone* tersebut. Sedangkan jika tidak terkoneksi maka *bone* tersebut dapat bergerak bebas dan dapat terpisah dari *parent bone*.

Entries bone

Berfungsi untuk membaca *entries* dari pembuatan sebuah *bone* baru. Jika sudah tidak ada *entries* maka pembuatan *bone* dianggap selesai.

Langkah selanjutnya adalah pembuatan *source code* dari *class creat (obj)*. *Class* tersebut berisikan data *bone* yang diambil dari pembuatan *riging* dari karakter hewan gajah. *Source code* tersebut berisikan data-data yang dibutuhkan dalam pembuatan *rig* karakter mulai dari koordita kepala dan ekor *bone*, koneksi *parent bone* dan lain-lai.

```

def create(obj):
    bpy.ops.object.mode_set(mode='EDIT')
    arm = obj.data
    bones = {}
    bone = arm.edit_bones.new('spinectrl')
    bone.head[:] = 0.0000, 0.930, 0.220
    bone.tail[:] = 0.0000, 0.930, 0.667
    bone.roll = -0.0000
    bone.use_connect = False
    bones['spinectrl'] = bone.name

    bone = arm.edit_bones.new('foot_r_front.IKctrl')
    bone.head[:] = -0.209, 1.058, -0.041
    bone.tail[:] = -0.209, 1.058, 0.109
    bone.roll = -0.0000
    bone.use_connect = False
    bones['foot_r_front.IKctrl'] = bone.name
    bone.parent = arm.edit_bones[bones['spinectrl']]

```

Source code diatas dituliskan dalam *text editor* yang bertujuan untuk membuat fungsi dari kumpulan beberapa data *bone* sehingga membentuk *Elephant Rig*. Data tersebut merupakan data dari nama *bone*, *connect bone*, *bone roll*, *bone parent*, koordinat *head* dan *tail* dari *bone* tersebut.

3.3 Perancangan Pengujian Metode *Inverse Kinematics* dan *Add On*

Rig karakter adalah sebuah kerangka yang berbentuk digital dan terhubung dengan objek 3D layaknya kerangka pada dunia nyata. Terdapat dua bagian dari *rig* karakter yaitu *join* dan *bone*, yang memiliki tugas masing-masing agar dapat menggerakkan objek saat proses *animating*. Gerakan tersebut yang akan di olah oleh *animator* untuk menggerakkan objek karakter. Fungsi utama dari *rig* adalah untuk memudahkan *animator* dalam memberikan gerakan dari objek karakter agar sesuai dengan keinginan. *Rig* dapat memiliki bentuk yang sederhana sampai dengan bentuk yang rumit, tergantung dari gerakan yang akan dilakukan pada objek karakter tersebut. Sebuah *rig* dapat dibuat dalam waktu singkat jika gerakan tersebut adalah sebuah gerakan yang sederhana. Namun akan memakan waktu sehari-hari bahkan berminggu-minggu jika *rig*

digunakan dalam sebuah objek karakter yang memiliki gerakan yang rumit (Syalabi, 2014).

Tantangan terbesar dalam *animator* adalah menciptakan sebuah karakter yang bisa mengeluarkan gerakan terbaik dari karakter tersebut, namun juga mengeluarkan gerakan yang natural dan terlihat hidup. Membuat gerakan sesuai kemampuan karakter sebenarnya juga sangat penting. Supervisor 3D Fabian Frank menjelaskan “Mengonsep *bone* dalam karakter merupakan suatu hal yang sangat penting dalam *rig* karakter. Sebuah *pra-visual* digunakan untuk menentukan dasar gerakan karakter kemudian digunakan sebagai panduan untuk *rig* karakter.” (VFX, 2014).

Pengujian akan dilakukan dengan cara menguji menerapkan metode *inverse kinematics* dengan *add on* yang telah dibuat pada *software* blender 2.79b. Selanjutnya akan dilakukan pembuatan simulasi gerak gajah saat berjalan. Pembuatan simulasi akan dilakukan dengan *tracking* pada *pose* yang sudah ditentukan sebelumnya. Setelah pembuatan simulasi gerak gajah saat berjalan selesai selanjutnya dilakukan perhitungan persentase kesalahan sumbu derajat kebebasan tulang antara data yang sudah diperoleh dan hasil dari simulasi.

Bentuk pengujian yang akan dilakukan adalah :

1. Apakah *add on generate elephant rig* jika implementasikan dengan *inverse kinematics* bekerja dengan baik ?
2. Berapa presentase kesalahan derajat dari simulasi gerak gajah berjalan menggunakan metode *inverse kinematics*?

3.4 Penerapan Metode Inverse Kinematics dan Add On

Penerapan metode akan dilakukan dengan mengimplementasikan metode *inverse kinematics* dengan *add on* telah dibuat sebelumnya. Pembuatan *inverse kinematics armature* dilakukan di mode *pose* dan memilih *tail bone* dari hasil *generate add on*.

Algoritma *inverse kinematics* :

1. Masukkan objek IK target
2. Jika tipe objek = “ARMATURE” maka pilih *sub target*
3. Masukkan objek pole target
4. Jika tipe objek = “ARMATURE” maka pilih *pole bone*
5. Masukkan derajat pole angle
6. Masukkan iterasi.
7. Masukkan *chaint_length*.
8. Masukkan *position*.
9. Masukkan *influence*.

Algoritma tersebut berfungsi sebagai pembuatan IK *armature* dalam *bone*. Pembuatan IK *armature* membutuhkan data *bone* agar gerakan yang dihasilkan sesuai dengan keinginan. Data dari pembuatan IK *armature* yang meliputi derajat *pole angle*, iterasi, *chaint_length*, *position* dan *influence* dapat dimasukkan sesuai kebutuhan, tergantung dari objek dan gerakan yang diinginkan. Sedangkan fungsi dari IK target yaitu untuk membuat *head bone* yang akan diikuti oleh *tail bone*. *Pole bone* berfungsi sebagai pembatas dari gerak IK *armature* untuk mempermudah dalam proses *animate*.



Gambar 3.22 Flowchart *inverse kinematics*

Langkah pertama yang dilakukan yaitu dengan memasukkan objek *armature*. Setelah itu memilih *bone* dari bagian *armature* untuk dijadikan *sub target*. Langkah selanjutnya yaitu memasukkan objek *armature* untuk dijadikan *pole target*, lalu memilih *bone* dari bagian *armature*. *Pole target* berfungsi sebagai pembatas rotasi untuk *bone* saat proses *animate*. Selanjutnya mengatur nilai dari *pole angle* sesuai dengan kebutuhan dari bentuk objek *armature*. Setelah itu dilanjutkan dengan memasukkan nilai iterasi. Nilai iterasi berfungsi sebagai batasan pemecahan IK untuk mendapatkan jarak toleransi dari IK target. Selanjutnya mengatur *chain length* yang berfungsi untuk menentukan jumlah *bone* yang diberikan efek IK. Selanjutnya memasukkan nilai *position* yang berfungsi sebagai bobot kendali IK. Langkah terakhir yaitu memasukkan nilai *influence* yang berfungsi untuk membuat *bone* yang diberikan efek IK mengikuti *sub target*. (Blender, 2016)

Pembuatan IK dapat dilakukan dengan menuliskan pada *text editor* dengan *source code* sebagai berikut .

```
bpy.data.objects["metarig"].data.bones.active=
bpy.data.objects["metarig"].pose.bones['foot_1_front.upper']
.bone
bpy.ops.pose.constraint_add(type='IK')
bpy.context.object.pose.bones["foot_1_front.upper"].constrain
ts["IK"].target = bpy.data.objects["metarig"]
bpy.context.object.pose.bones["foot_1_front.upper"].constrain
ts["IK"].subtarget = "foot_1_front.IKtrgt"
bpy.context.object.pose.bones["foot_1_front.upper"].constrain
ts["IK"].pole_target = bpy.data.objects["metarig"]
bpy.context.object.pose.bones["foot_1_front.upper"].constrain
ts["IK"].pole_subtarget = "foot_1_front.pole"
bpy.context.object.pose.bones["foot_1_front.upper"].constrain
ts["IK"].chain_count = 2
bpy.context.object.pose.bones["foot_1_front.upper"].constrain
ts["IK"].pole_angle = 1.5708
bpy.context.object.pose.bones["foot_1_back.upper"].constrain
ts["IK"].influence = 1
bpy.context.object.pose.bones["foot_1_back.upper"].constrain
ts["IK"].weight = 1
```

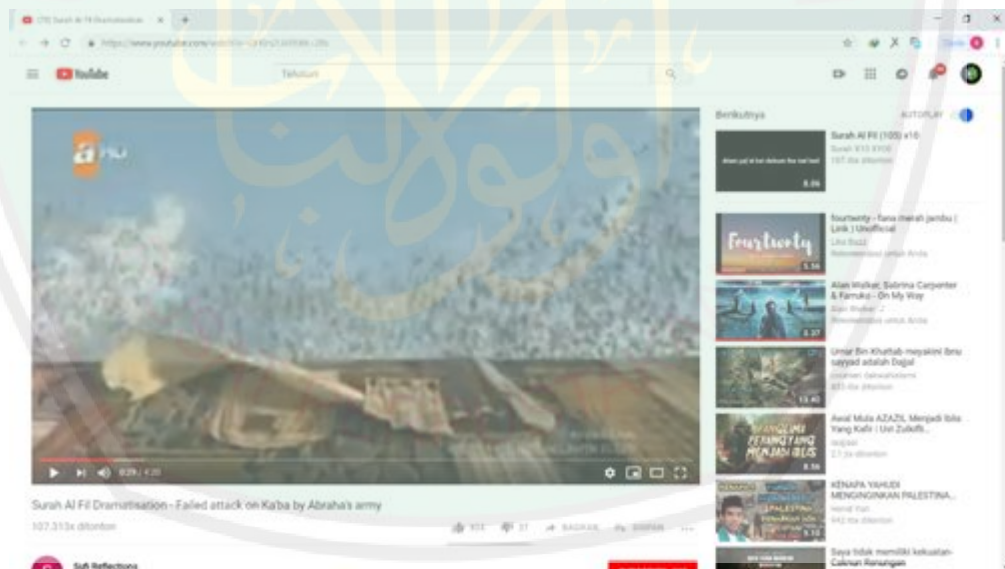
BAB IV

HASIL DAN PEMBAHASAN

Pada bab ini akan dibahas mengenai hasil pengerjaan yang telah dilakukan sesuai dengan perancangan pada bab sebelumnya. Pengujian yang akan dilakukan adalah tentang bagaimana hasil dan dampak yang diberikan oleh *add on elephant rig* dengan mengombinasikan *add on* bawaan blender yang menggunakan metode *inverse kinematics*.

4.1 Pra Produksi

Ide cerita dalam pembuatan animasi *asbabun nuzul Al-Fiil* diambil dari sebuah *scene* film yang berjudul *Muhammad, The Messenger of God*.



Gambar 4.1 Ide cerita animasi
(<https://www.youtube.com/watch?v=GFKtn2LkiYE&t=28s>).

4.1.1 Storyboard

Langkah setelah menentukan ide cerita dan konsep karakter selanjutnya membuat alur cerita dengan kerangka konsep yang dibentuk dalam *storyboard*.

Tabel 4.1 Storyboard

	<p>-Opening</p> <p>-Kamera memperlihatkan Kota Makkah</p> <p>-Memperlihatkan Ka'bah</p>
	<p>-Saat Pasukan Raja Abrahah sudah mendekati kota Makkah, Kakek Nabi Muhammad SAW (Abdul Muthalib) Berdo'a didepan Ka'bah.</p>
	<p>-Pasukan Gajah raja Abrahah berjalan menuju kota Makkah</p>
	<p>-Pasukan Ababil muncul dan terbang menuju pasukan Gajah raja Abrahah</p>

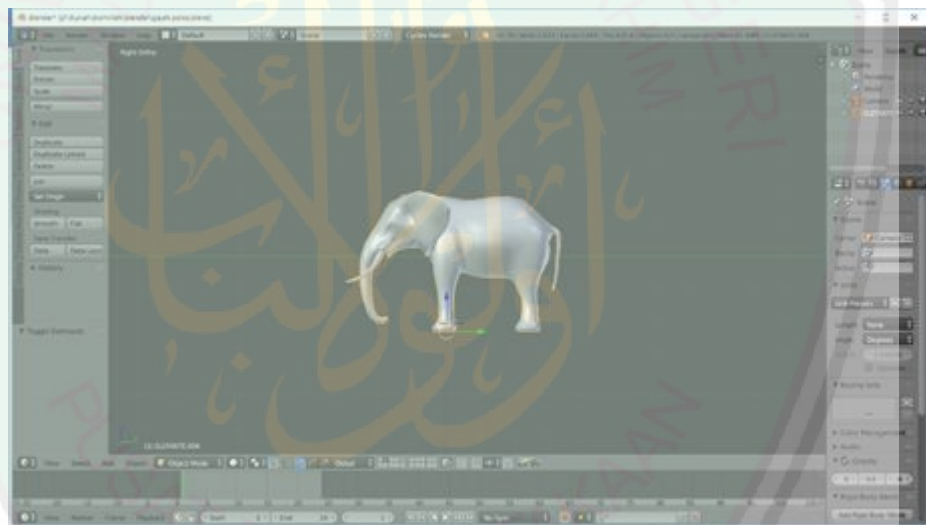
	<p>-Burung Ababil menjatuhkan batu api dari neraka</p>
	<p>-Hujan batu yang dijatuhkan oleh burung Ababil</p>
	<p>-Pasukan Gajah Abrahah mencoba lari dari hujan batu namun mereka tetap mati karena diserang oleh pasukan burung Ababil dengan batu api.</p>

4.2 Produksi

4.2.1 Modeling

Pembuatan *modeling* dalam pembuatan film animasi pendek yang berjudul *asbabun nuzul Al-Fiil* menggunakan *primitive object* yang sudah tersedia dalam *software* blender. *Primitive object* akan digunakan

dalam pembuatan model yang diperlukan dalam pembuatan animasi 3D dengan cara *extrude* yang berfungsi untuk menambahkan *vertex* yang diseleksi, *scale* yang berfungsi untuk merubah bentuk *vertex* yang diseleksi, *subdivide* yang berfungsi untuk membuat *vertex* sesuai dengan jumlah objek sebelumnya, dan fungsi operator lainnya sesuai dengan kebutuhan untuk membentuk suatu objek. Proses *modeling* juga membutuhkan beberapa *add on* seperti *cloth simulation* yang berfungsi untuk membuat objek memiliki efek seperti kain, *wind effect* yang berfungsi untuk memberikan efek angin pada objek, dan *add on* lainnya.

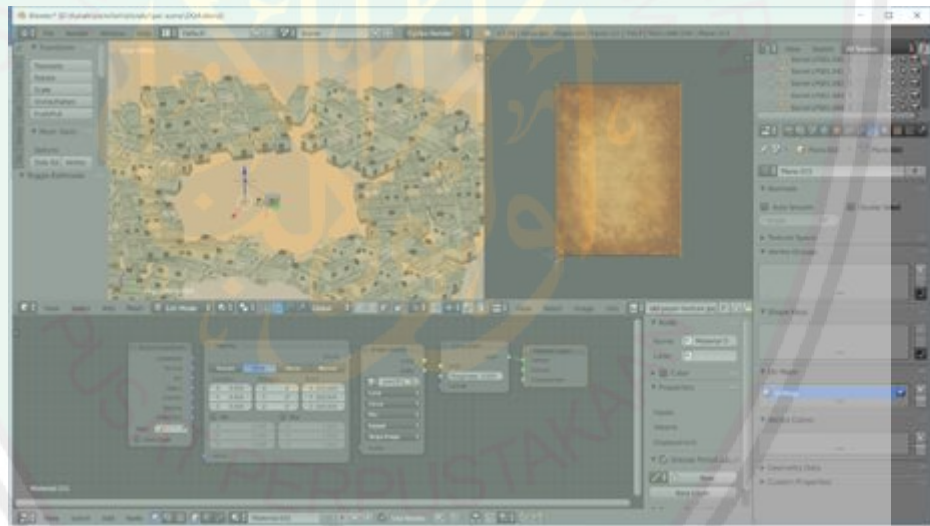


Gambar 4.2 *Modeling* gajah.

Model karakter 3D gajah yang sudah dibuat dan dibahas dalam bab sebelumnya. Selanjutnya pembuatan *modeling* objek *environment* yang di butuhkan dalam pembuatan animasi.

4.2.2 Texturing

Tahap *texturing* berfungsi untuk memberi warna dari objek-objek yang telah dibuat sebelumnya. Pada tahap ini pemberian warna menggunakan beberapa material warna sesuai dengan kebutuhan. Selain itu pemberian warna juga membutuhkan *texture mapping* yang berfungsi untuk memberikan warna sesuai keinginan. Pembuatan *texture mapping* menggunakan tools pada *software* blender seperti *UV map* dan *node editor* untuk memberikan efek warna yang maksimal. Selain itu *software* pendukung lainnya juga digunakan dalam pembuatan bahan warna sebagai *texture* seperti photoshop.

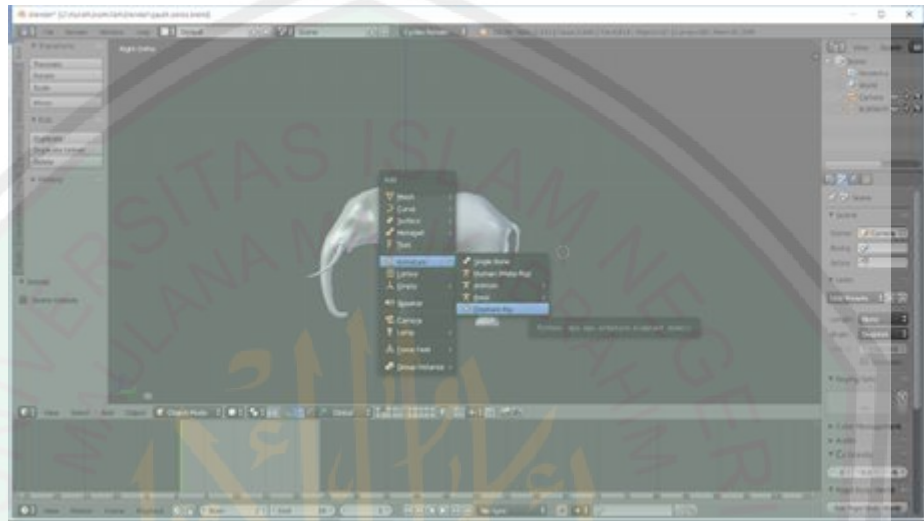


Gambar 4.5 *Texture environment* menggunakan *node editor*

Pada Gambar 4.5 menunjukkan hasil dari *texturing* menggunakan tools *UV map* dan *node editor*. Selanjutnya akan dilakukan *texturing* pada objek lainnya sesuai dengan kebutuhan agar objek terlihat lebih bagus.

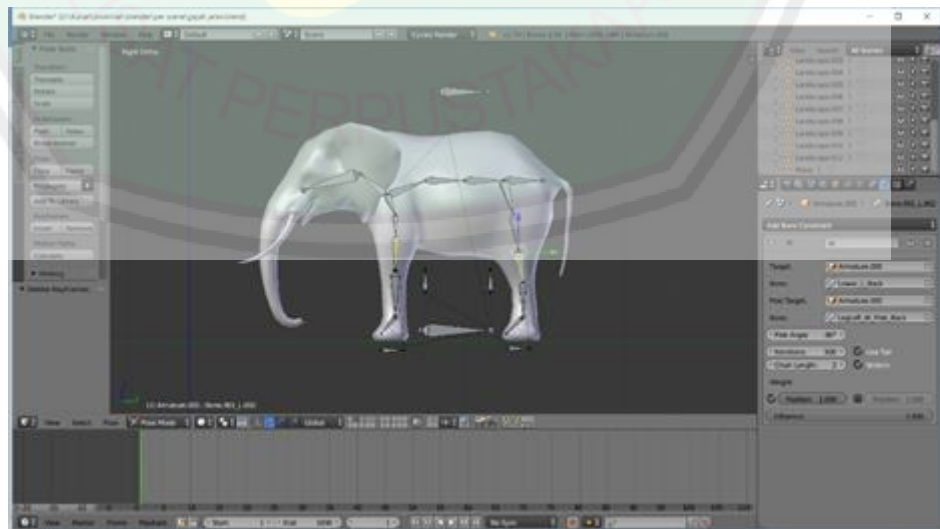
4.2.3 Rigging

Proses *rigging* dilakukan dengan *generate bone* dari *add on* yang telah dibuat sebelumnya. *Generate bone* dilakukan dengan cara menekan *shortcut Shift + A + Armature + Elephant Rig*.



Gambar 4.6 *Generate bone* dengan *add on*

Selanjutnya mengkombinasikan *bone* yang sudah digenerate dari *add on* yang dibuat sebelumnya dengan *source code* yang telah dibuat sebelumnya.

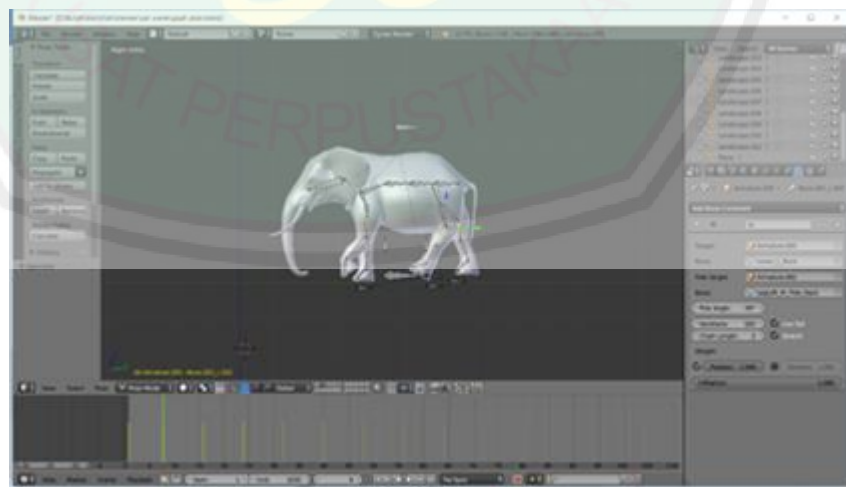


Gambar 4.7 Mengkombinasikan *bone* dengan *source code*.

Pengkombinasian *bone* gajah dengan *add on inverse kinematics* dapat dilakukan dengan mudah karena terdapat tambahan *bone* untuk menjadi IK target dan pole target.

4.2.4 Animating

Seluruh tahapan dalam proses *animating* dilakukan menggunakan cara menyimpan *keypose* pada *frame*. Penyimpanan *keypose* pada *frame* dilakukan dengan cara *frame by frame* yang artinya menyimpan *pose* dari *frame* 1 ke *frame* selanjutnya sesuai dengan sumbu derajat kebebasan tulang gajah yang sudah didapat. Proses *animating* juga dilakukan dengan *add on* bawaan blender yang menggunakan metode *inverse kinematics*. Fungsi dari *add on* tersebut untuk mengurangi kesalahan *pose* saat proses *animating*. Selain itu *add on* tersebut juga berfungsi untuk memudahkan animator dalam tahap *animating* karena *add on* tersebut hanya perlu menggerakkan *bone* *Ikctrl* untuk membuat pola berjalan (*walk cycle*).



Gambar 4.8 Animating menggunakan metode *inverse kinematics*

Pada proses *animating* objek gajah dilakukan beberapa rotasi sesuai kebutuhan. Rotasi tersebut mengacu pada sumbu kebebasan tulang yang sudah didapatkan sebelumnya.

4.2.5 Lighting

Proses *lighting* dibutuhkan dalam pembuatan film animasi agar saat *rendering* objek tidak terlihat terlalu gelap atau terlalu terang. Proses *lighting* dilakukan dengan objek *sun* yang sudah ada dalam *software* blender dengan pengaturan efek *emission*, *strength* cahaya 2.000 agar sesuai dengan suasana agak gelap.

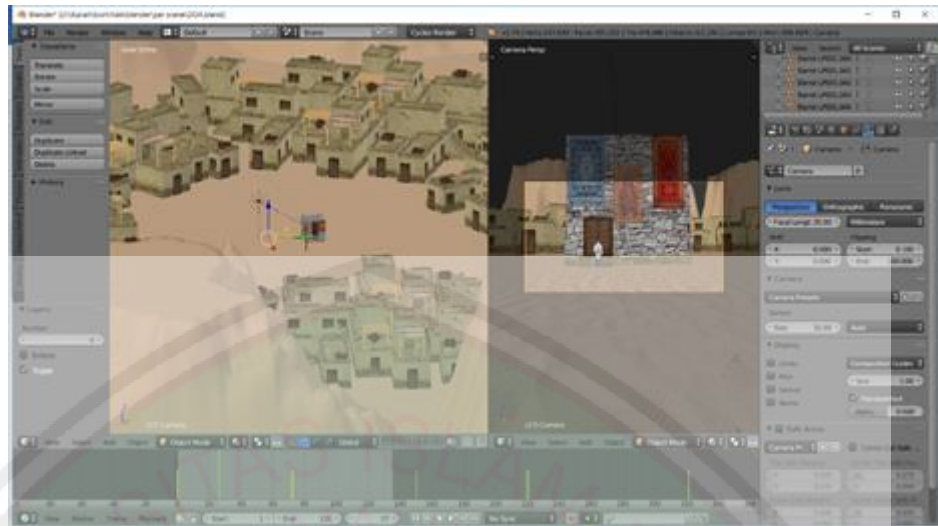


Gambar 4.9 Setting lighting

Proses *lighting* dilakukan dengan cara yang sederhana namun dengan efek yang diatur sesuai dengan suasana alur cerita.

4.2.6 Camera Operating

Proses pembuatan kamera dilakukan dengan objek kamera yang sudah disediakan dalam *software* blender. Dalam pembuatan *camera operating* dilakukan dengan cara merubah koordinat kamera sesuai kebutuhan dan menyimpannya pada *frame* dengan acuan *storyboard* yang sudah ada.

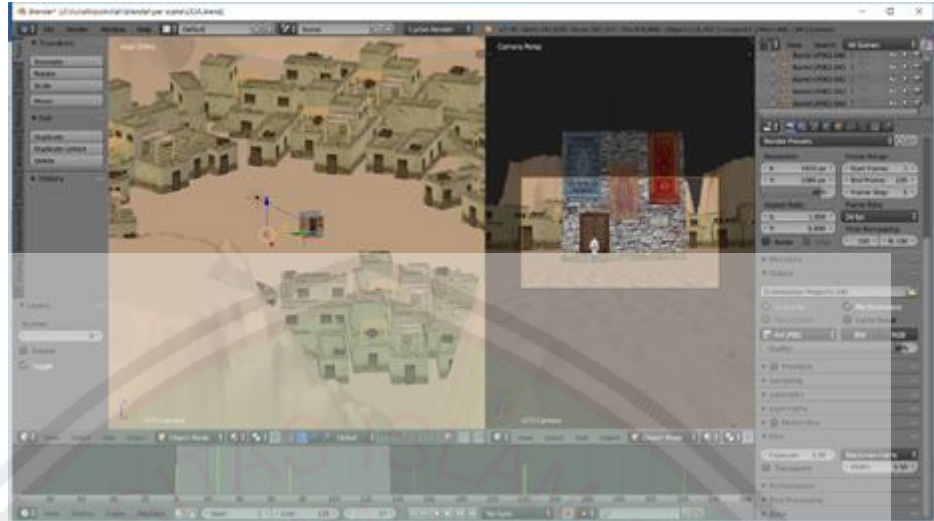


Gambar 4.10 *Camera operating.*

Sudut pandang kamera bertujuan untuk menampilkan objek saat di-render.

4.2.7 *Rendering*

Proses *rendering* berfungsi untuk menampilkan objek dari sudut pandang kamera yang telah dibuat sebelumnya. Hasil *rendering* dapat berupa gambar maupun video. Pada tahapan *rendering* hasil yang dimunculkan berupa video untuk pembuatan film animasi *asbabun nuzul Al-Fiil*. *Rendering* dilakukan dengan cara memilih *cycles render* yang bertujuan untuk memaksimalkan hasil dari proses *texturing*. Sedangkan hasil dari *rendering* menggunakan perbandingan 24 fps yang artinya setiap 24 frame akan menjadikan video dengan durasi 1 detik. Terdapat dua cara dalam melakukan proses *rendering*, yaitu *CPU rendering* yang memproses render dengan mengandalkan *processor*. *GPU compute* yang mengandalkan VGA untuk proses *render*.



Gambar 4.11 Pengaturan *rendering*

Proses *rendering* menggunakan pengaturan seperti pada gambar 4.10 dengan dimensi 1920 piksel x 1080 piksel yang beresolusi 80% dengan kualitas 90% dan hasil video bertipe AVI JPEG.

4.1 Pasca Produksi

Tahapan terakhir dalam pembuatan film animasi pendek yang berjudul *asbabun nuzul Al-Fiil* yaitu *final editing*. Proses ini dilakukan dengan mengedit video yang didapat dari proses *rendering* menggunakan aplikasi Vegas 14.



Gambar 4.12 Proses *final editing*

Final editing yang dilakukan untuk membuat film animasi sesuai dengan *storyboard* yang telah dibuat. Hasil dari *final editing* yaitu berupa video yang sudah diberikan efek video, efek suara dan penggabungan beberapa *scene* yang dari proses *rendering* agar sesuai dengan ide cerita yang sudah dibuat dalam *storyboard*.

4.3 Pengujian Add On

4.3.1 Importing Add On

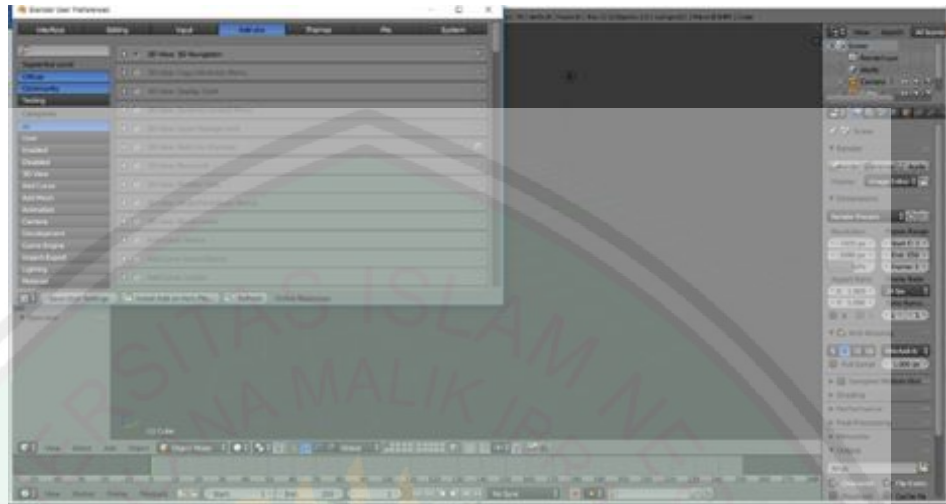
Importing add on dilakukan dengan cara menambahkan *add on* telah dibuat sebelumnya ke dalam aplikasi blender dengan cara menyimpan *source code* dari pembuatan *add on* dengan format *.py*. selanjutnya memilih file tersebut untuk di *import* ke dalam aplikasi blender.



Gambar 4.13 Membuka *user preferences*

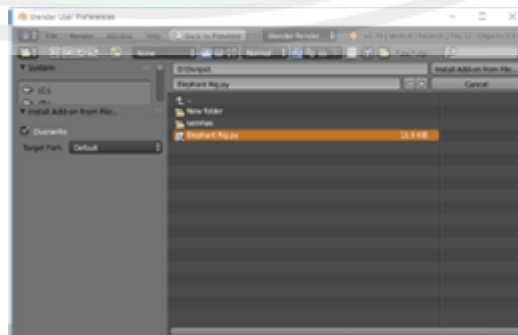
Langkah pertama yaitu dengan membuka menu bar *file* bagian ini akan menampilkan menu dari *user preferences*, yaitu tempat pengguna untuk mengatur ulang blender sesuai dengan keinginan dan kebutuhan. Pengaturan tersebut dapat dirubah sesuai dengan kebutuhan pengguna,

mulai dari pengaturan tombol hingga pengaturan *add on* yang ingin dimunculkan dala halaman kerja pada aplikasi blender.



Gambar 4.14 *Install add on*

Langkah selanjutnya setelah menu *user preference* terbuka kita memilih sub menu *add on* selanjutnya pilih menu *install add on from file*. Aplikasi blender menyediakan *add on* yang sangat banyak dan beberapa dari *add on* tersebut telah aktif secara otomatis dari sistem *default* dari blender. Tanda centang dari *tools box* merupakan tanda dari *add on* yang saat ini aktif. *Add on* dari luar aplikasi blender keadaannya belum aktif meskipun sudah terinstal, maka dari itu pengguna harus memberikan tanda centang dengan menekan *tools box add on* yang ingin diaktifkan.



Gambar 4.15 *Open add on file*

Langkah selanjutnya setelah memasuki halaman pemilihan folder *add on*, pilih file *add on* yang ingin diinstal.



Gambar 4.16 *Save user setting*

Kemudian, *file add on* yang dipilih tadi akan muncul pada tampilan *user preference*. Setelah itu centang pada *tools box add on* untuk mengaktifkan *add on* tersebut.

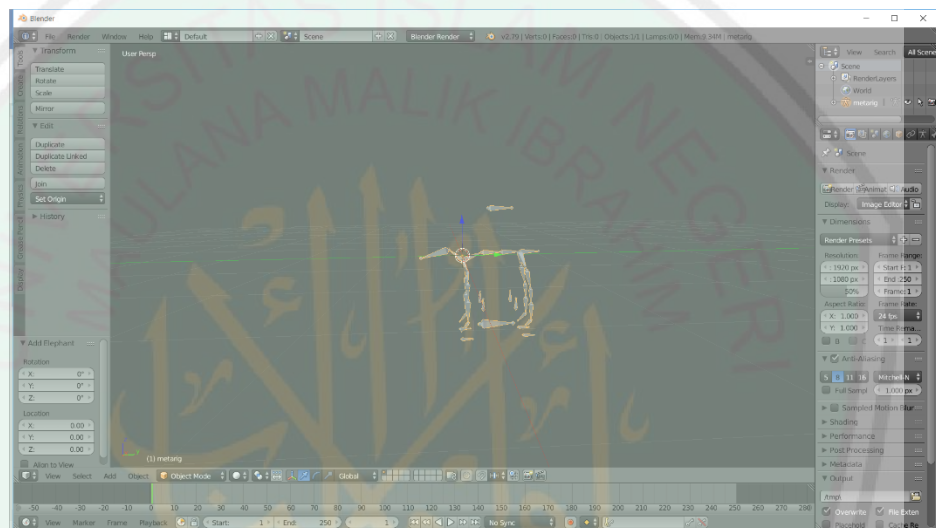


Gambar 4.17 *Generating bone*

Jika *add on* berhasil diaktifkan maka *add on* akan muncul di halaman kerja sesuai dengan kategori dan penempatan *add on* yang telah ditulis dalam *source code* pada *add on* tersebut.

4.3.2 Generate Bone

Langkah selanjutnya setelah *add on* berhasil di *import* dan diaktifkan *add on* tersebut akan berada pada *tools add object* pada menu bar yang berada di atas *timeline* atau juga bisa menggunakan *shourtcut* dengan menekan *shift+ A* selanjutnya pilih menu *armature* kemudian pilih *Elephant Rig* untuk memunculkan *bone* karakter gajah.

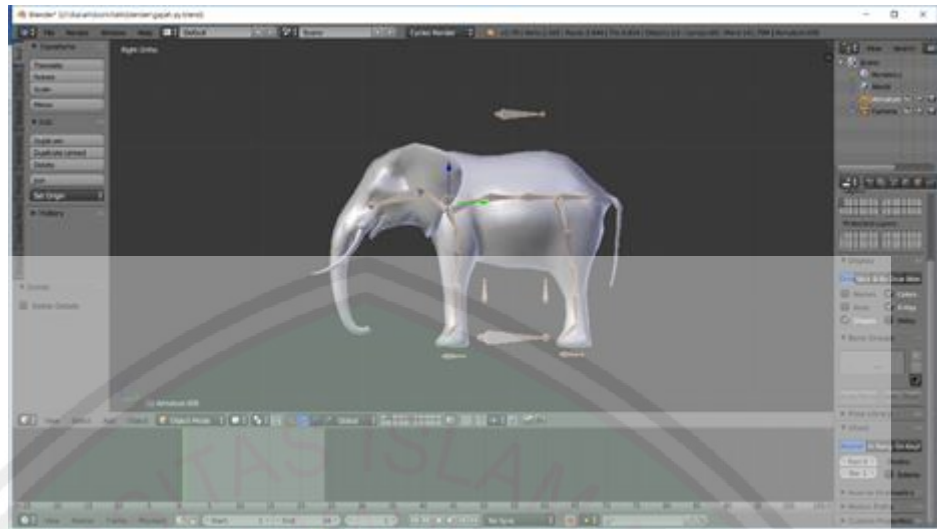


Gambar 4.18 Generate bone

Berdasarkan gambar diatas percobaan fungsi dari *add on Elephant Rig* yang berfungsi untuk membuat *bone* gajah dengan *generate* otomatis berhasil dengan baik pada aplikasi blender 2.79b .

4.3.3 Rigging

Proses pembuatan *rigging* dalam animasi 3D memiliki 2 unsur yang saling berhubungan yaitu *joint* dan *bone*. Pada pembuatan *rigging* dilakukan secara manual dan membuat kerangka sesuai kebutuhan alat gerak dari objek dan karakter 3D. Proses pembuatan *rigging* saat ini dilakukan adalah dengan menggunakan *add on elephant rig* yang telah dibuat sebelumnya.

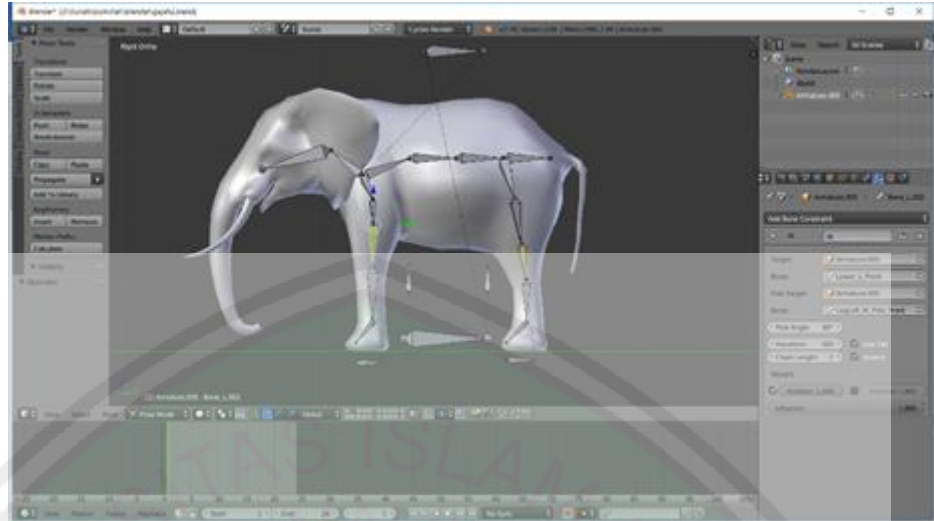


Gambar 4.19 *Add on rigging*

Pada gambar 4.19 menunjukkan bahwa *ringing* dari 3D karakter dan *add on elephant rig* berfungsi dengan baik.

4.3.4 *Pengujian Add On dengan Metode Inverse Kinematics*

Salah satu tahapan dalam membuat animasi adalah *rigging* dan *animate*. Dalam *software* blender terdapat *add on* yang menggunakan metode *inverse kinematics* sebagai alat untuk memudahkan proses *animate*. Kelebihan dari *add on* tersebut adalah mengurangi kesalahan dalam menyimpan *pose* saat dimasukkan ke dalam frame. Dalam *software* blender juga terdapat *add on Rigify* yang dapat *generate bone* manusia. Namun kita harus menambahkan beberapa *bone* untuk mengkombinasikan dengan *add on inverse kinematics*. Maka dari itu kelebihan dari *add on Elephant Rig* yaitu terdapat *bone* tambahan untuk mengkombinasikan dengan *add on* bawaan dari *software* blender yang menggunakan metode *inverse kinematics*.



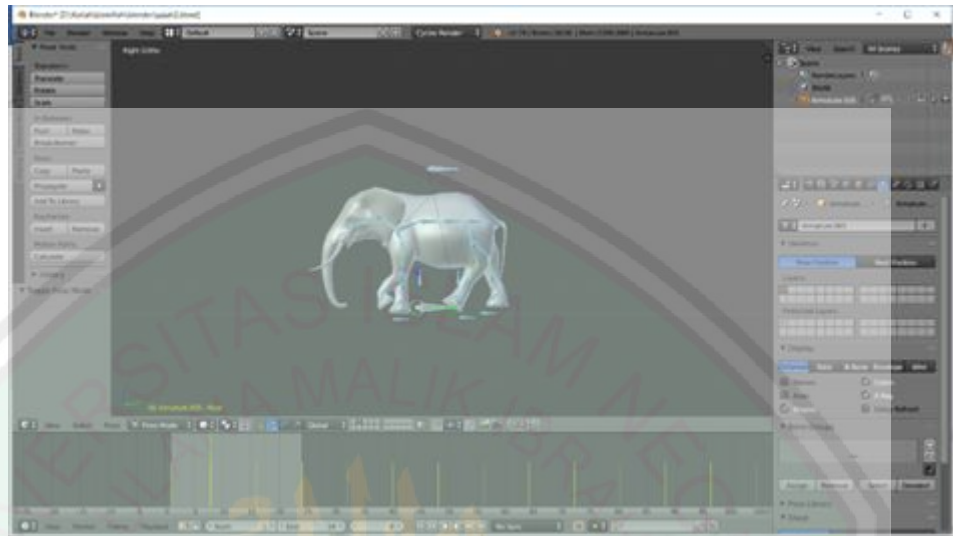
Gambar 4.20 Add on inverse kinematics.

Pada gambar 4.20 menunjukkan bahwa *add on Elephant Rig* dapat dengan mudah dikombinasikan dengan *add on inverse kinematics* yang ada dalam *software* blender karena pada *add on Elephant Rig* sudah dilengkapi dengan *bone inverse kinematics* (IK) target dan *pole target*.

4.3.5 Pembuatan Simulasi Gerak Gajah

Pembuatan simulasi gerak gajah akan dilakukan dengan *add on Elephant Rig* dan *add on inverse kinematics*. Simulasi gerak gajah akan dilakukan dengan mengacu pada data sumbu derajat kebebasan tulang gajah yang sudah didapatkan. Pembuatan simulasi gerak gajah saat berjalan akan dilakukan dengan mengganti koordinat lokasi dan rotasi dari *bone* sesuai kebutuhan. Rotasi yang digunakan dalam pembuatan simulasi gerak gajah saat berjalan yaitu dengan pengaturan rotasi mode Quaterion (WXYZ). Simulasi gerak gajah saat berjalan akan dilakukan dengan cara *tracking* pada gambar yang telah dilakukan pengukuran sumbu derajat kebebasan tulang pada bab sebelumnya. Pembuatan

simulasi dengan cara *tracking* bertujuan untuk memudahkan menemukan koordinat *bone*.



Gambar 4.21 Simulasi Gerak Gajah

Pada gambar 4.20 menunjukkan simulasi gajah saat berjalan dengan *add on Elephant Rig* yang telah dikombinasikan dengan *add on inverse kinematics* yang sudah ada didalam *software* blender. Dari pembuatan simulasi gerak gajah yang mengacu pada sumbu derajat kebebasan tulang yang sudah didapatkan, diperoleh koordinat tulang sebagai berikut :

Tabel 4.2 Koordinat *bone* kaki kanan.

Frame	Bone Kaki Kanan						
	Sumbu	Depan atas	Depan bawah	Belakang atas	Belakang bawah	Ikctrl depan	Ikctrl Belakang
		Rotation	Rotation	Rotation	Rotation	Location	Location
8	W	0.000	0.000	0.985	0.985	-	-
	X	0.000	0.000	0.174	-0.174	0.00000	0.00000
	Y	0.000	0.000	0.000	0.000	0.00000	0.14368
	Z	0.000	0.000	0.000	0.000	0.00000	0.05923
16	W	0.985	0.985	0.956	0.000	-	-
	X	0.174	-0.174	0.294	0.000	0.00000	0.00000
	Y	0.000	0.000	0.000	0.000	0.00000	0.15475
	Z	0.000	0.000	0.000	0.000	0.00000	0.06336

Tabel 4.3 Koordinat *bone* Kaki Kiri.

Frame	Bone Kaki Kiri						
	Sumbu	Depan atas	Depan bawah	Belakang atas	Belakang bawah	Ikctrl depan	Ikctrl Belakang
		Rotation	Rotation	Rotation	Rotation	Location	Location
8	W	0.924	0.766	0.966	0.000	-	-
	X	-0.383	0.643	-0.259	0.000	0.00000	0.00000
	Y	0.000	0.000	0.000	0.000	0.00000	-0.08985
	Z	0.000	0.000	0.000	0.000	0.00000	0.00432
16	W	0.966	0.000	0.991	0.000	-	-
	X	-0.259	0.000	-0.131	0.000	0.00000	0.00000
	Y	0.000	0.000	0.000	0.000	-0.19049	-0.05219
	Z	0.000	0.000	0.000	0.000	-0.03810	-0.00370

Pada tabel diatas menunjukkan koordinat dari *bone* gajah dalam pembuatan 1 gerakan gajah berjalan. Selanjutnya akan dilakukan perulangan saat gajah berjalan guna membentuk gerakan *walk cycle*.

4.3.6 Hasil Uji Coba

Berdasarkan rancangan pengujian yang telah dijabarkan pada bab sebelumnya, maka akan dilakukan pengujian terhadap nilai kesalahan derajat kebebasan tulang pada simulasi gerak gajah saat berjalan untuk mengetahui kesalahan derajat dari simulasi gerak gajah saat berjalan yang diimplementasikan dengan metode *inverse kinematics* dengan menggunakan rumus :

$$error (\%) = \frac{\text{nilai data} - \text{nilai hasil}}{\text{nilai data}} \times 100\% \quad (4.1)$$

Hasil derajat tulang yang didapatkan setelah pembuatan simulasi gerak gajah saat berjalan sebagai berikut :

Tabel 4.4 Pengujian simulasi gerak gajah saat berjalan.

No	Kaki		Pose	Data	Hasil
	Depan	Belakang			
1	Sendi Atas		1	50	45
2	Sendi Bawah		1	70	80
3	Sendi Atas		2	40	35
4	Sendi Bawah		2	0	0
5	Sendi Atas		3	0	0
6	Sendi Bawah		3	0	0
7	Sendi Atas		4	20	20
8	Sendi Bawah		4	20	20
9		Sendi Atas	1	25	30
10		Sendi Bawah	1	0	0
11		Sendi Atas	2	20	15
12		Sendi Bawah	2	0	0
13		Sendi Atas	3	20	20
14		Sendi Bawah	3	0	0
15		Sendi Atas	4	30	35
16		Sendi Bawah	4	0	0

Berikut ini perhitungan kesalahan sumbu derajat kebebasan tulang bagian depan dari simulasi gerak gajah saat berjalan menggunakan metode *inverse kinematics* :

Error (%) sendi atas *pose 1* :

$$\begin{aligned} \text{error} (\%) &= \frac{50 - 45}{50} \times 100\% \\ &= 10\% \end{aligned}$$

Error (%) sendi bawah *pose 1* :

$$\begin{aligned} \text{error} (\%) &= \frac{70 - 80}{70} \times 100\% \\ &= 14,3\% \end{aligned}$$

Error (%) sendi atas *pose 2* :

$$\begin{aligned} \text{error} (\%) &= \frac{40 - 35}{40} \times 100\% \\ &= 12,5\% \end{aligned}$$

Error (%) sendi atas *pose* 4 :

$$\begin{aligned} \text{error} (\%) &= \frac{20 - 23}{20} \times 100\% \\ &= 15\% \end{aligned}$$

Error (%) sendi bawah *pose* 4 :

$$\begin{aligned} \text{error} (\%) &= \frac{20 - 22}{20} \times 100\% \\ &= 10\% \end{aligned}$$

Berikut ini perhitungan kesalahan sumbu derajat kebebasan tulang bagian belakang dari simulasi gerak gajah saat berjalan menggunakan metode *inverse kinematics* :

Error (%) sendi atas *pose* 1 :

$$\begin{aligned} \text{error} (\%) &= \frac{25 - 30}{25} \times 100\% \\ &= 20\% \end{aligned}$$

Error (%) sendi atas *pose* 2 :

$$\begin{aligned} \text{error} (\%) &= \frac{20 - 15}{20} \times 100\% \\ &= 25\% \end{aligned}$$

Error (%) sendi atas *pose* 3 :

$$\begin{aligned} \text{error} (\%) &= \frac{20 - 21}{20} \times 100\% \\ &= 5\% \end{aligned}$$

Error (%) sendi atas *pose* 4 :

$$\begin{aligned} \text{error} (\%) &= \frac{30 - 35}{30} \times 100\% \\ &= 16,7\% \end{aligned}$$

Tabel 4.5 Tingkat kesalahan derajat pada simulasi.

No	Kaki		Pose	Hasil perhitungan
	Depan	Belakang		
1	Sendi Atas		1	10%
2	Sendi Bawah		1	14,3%
3	Sendi Atas		2	12,5%
4	Sendi Bawah		2	0
5	Sendi Atas		3	0
6	Sendi Bawah		3	0
7	Sendi Atas		4	15%
8	Sendi Bawah		4	10%
9		Sendi Atas	1	20%
10		Sendi Bawah	1	0
11		Sendi Atas	2	25%
12		Sendi Bawah	2	0
13		Sendi Atas	3	5%
14		Sendi Bawah	3	0
15		Sendi Atas	4	16,7%
16		Sendi Bawah	4	0

Dari tabel tingkat kesalahan , dapat diamati bahwa nilai kesalahan atau *error* tidak bergantung pada *input*, sehingga nilai kesalahan ini termasuk dalam galat *nonrelative*. Untuk menentukan nilai rata-rata tingkat kesalahan *output* terhadap *input* metode perhitungan *invers kinematics* dengan rumus rata-rata sederhana dapat menggunakan jumlah nilai *error* yang muncul dibagi dengan jumlah data :

$$Rata = \frac{\sum error}{\sum data} = \frac{128,5}{16} = 8,03\%$$

4.4 Integrasi Dalam Islam

Dalam animasi *asbabun nuzul* Al-Fiil bercerita tentang kesombongan raja Abrahah yang berniat menyerang Ka'bah. Namun penyerangan tersebut gagal karena Allah SWT mengirimkan pasukan burung Ababil yang membawa batu api dari neraka. Sungguh hal yang tidak masuk akal jika pasukan gajah yang besar dan kuat dapat dikalahkan oleh hewan yang kecil dan tidak berbahaya.

Maka dari itu itu sebaik-baik kisah sejarah yang dapat diambil pelajaran dan hikmah yang berharga adalah kisah-kisah yang terdapat dalam ayat-ayat al-Qur'an dan hadits. Seperti yang dijelaskan dalam Al-Qur'an surat Yusuf ayat 111 yang berbunyi :

لَقَدْ كَانَ فِي قَصَصِهِمْ عِبْرَةً لِأُولِي الْأَلْبَابِ مَا كَانَ حَدِيثًا يُفْتَرَىٰ وَلَٰكِن تَصَدِيقَ
الَّذِي بَيْنَ يَدَيْهِ وَتَفْصِيلَ كُلِّ شَيْءٍ وَهُدًى وَرَحْمَةً لِّقَوْمٍ يُؤْمِنُونَ ١١١

Artinya : “*Sesungguhnya pada kisah-kisah mereka itu terdapat pengajaran bagi orang-orang yang mempunyai akal. Al Quran itu bukanlah cerita yang dibuat-buat, akan tetapi membenarkan (kitab-kitab) yang sebelumnya dan menjelaskan segala sesuatu, dan sebagai petunjuk dan rahmat bagi kaum yang beriman*”.

Salah satu hikmah yang bisa diambil dari animasi tersebut bahwa semua hal sudah mempunyai ukuran. Oleh karena itu janganlah bersikap sombong dan semena-mena karena mempunyai kekuatan yang besar. Seperti halnya yang sudah dijelaskan dalam Al-Qur'an surat Al-Qamar ayat 49 yang berbunyi :

إِنَّا كُلَّ شَيْءٍ خَلَقْنَاهُ بِقَدَرٍ ٤٩

Artinya : “*Sesungguhnya Kami menciptakan segala sesuatu menurut ukuran*”.

Dalam surat Al-Qamar ayat 49 menjelaskan bahwa segala sesuatu sudah ada ukurannya. Begitupun dengan anatomi dan cara berjalan hewan gajah. Dan segala ilmu yang ada di dunia semunya memiliki porsinya masing – masing.

Dalam surat An-Nisa' ayat 66 juga dijelaskan tentang mengamalkan ilmu yang berbunyi :

وَلَوْ أَنَّا كَتَبْنَا عَلَيْهِمْ أَنْ اقْتُلُوا أَنْفُسَكُمْ أَوْ اأَخْرَجُوا مِنْ دِيَارِكُمْ مَا فَعَلُوهُ إِلَّا قَلِيلًا
مِّنْهُمْ وَلَوْ أَنَّهُمْ فَعَلُوا مَا يُوعَظُونَ بِهِ لَكَانَ خَيْرًا لَّهُمْ وَأَشَدَّ تَنبِيئًا ٦٦

Artinya : “*Dan sesungguhnya kalau Kami perintahkan kepada mereka: "Bunuhlah dirimu atau keluarlah kamu dari kampungmu", niscaya mereka tidak akan melakukannya kecuali sebagian kecil dari mereka. Dan sesungguhnya kalau mereka melaksanakan pelajaran yang diberikan kepada*

mereka, tentulah hal yang demikian itu lebih baik bagi mereka dan lebih menguatkan (iman mereka)”.

Surat diatas menjelaskan bahwa mengamalkan ilmu itu lebih baik karena dapat menguatkan iman mereka. Oleh karena itu mengamalkan dan mengembangkan ilmu yang sudah didapat sangatlah diutamakan. Maka dalam skripsi ini dapat mempelajari tentang cerita yang ada dalam kandungan surat Al-Fiil, baik dari segi cerita, maupun tokoh yang berada didalam animasi tersebut.



BAB V

KESIMPULAN DAN SARAN

5.1 Kesimpulan

Kesimpulan dari hasil penelitian yang membahas pembuatan *add on Elephant Rig* yang diimplementasikan dengan metode *inverse kinematics* pada *software* blender dapat berjalan dengan baik dengan rata-rata *error* 8,03% dari 16 data. Data tersebut berupa 2 *bone* dari masing-masing kaki dari gajah dengan 2 *pose* yang membentuk pola berjalan gajah. Hasil dari pembuatan *add on Elephant Rig* juga dapat diterapkan dalam pembuatan animasi dengan objek gajah seperti yang sudah dibuat dalam penelitian ini. Hasil dari animasi yang dibuat yaitu animasi *asbabun nuzul* Al-Fiil.

5.2 Saran

Hasil *add on* pada penelitian ini hanya meliputi proses *rigging* dalam pembuatan animasi. Penelitian selanjutnya diharapkan dapat dikembangkan dengan pembuatan *add on* yang dapat *generate* gerakan pola gajah berjalan di dalam *keyframe* pada proses *animate*. Sehingga dapat membantu dalam pembuatan proses pembuatan film animasi.

DAFTAR PUSTAKA

- Beane, a. (2012). *3D Animation ESSENTIALS*. indianapolis: John Wiley & Sons, Inc.
- Bhatti, Z. (2013). Forward and Inverse Kinematics Seamless Matching Using Jacobian.
- Blender. (2016, 07 28). Diambil kembali dari https://en.wikibooks.org/wiki/Blender_3D:_Noob_to_Pro/Advanced_Tutorials/Advanced_Animation/Guided_tour/Const/ik
- CGIMeetup. (2016, january 4). Diambil kembali dari [www.youtube.com: https://www.youtube.com/watch?v=ru0tQRJ4qKs](http://www.youtube.com/watch?v=ru0tQRJ4qKs)
- Foundation, B. (2017, Oktober 1). www.blender.org. Diambil kembali dari <https://www.blender.org/about/>
- Foundation, P. (2017). <https://www.python.org/about>.
- Hasibuan, Z. A. (2007). *METODOLOGI PENELITIAN PADA BIDANG ILMU KOMPUTER DAN TEKNOLOGI INFORMASI*.
- Hilgevoord. (2006). American Journal of Physics. *The Uncertainty Principle*.
- J. Huang, M. F. (2016). *Inverse kinematics using dynamic joint parameters: inverse kinematics animation synthesis learnt from sub-divided motion micro-segments*.
- Jing Huang, C. P. (2005). *An Efficient Energy Transfer Inverse Kinematics*.
- Lei, V. D. (2013). *Inverse Kinematics Techniques in the BirthPlay Application*.
- M Hasan, I. (2002). *pokok-pokok materi : teori pengambil keputusan*. jakarta: Ghalia Indonesia.
- MPC. (2014, November 17). Diambil kembali dari [www.youtube.com: https://www.youtube.com/watch?v=qsZbtfQmRCc&t=21s](http://www.youtube.com/watch?v=qsZbtfQmRCc&t=21s)
- Nilsson, R. (2009). Mater Thesis. *Inverse Kinematics*.
- Pcmag. (2018, 1 17). <https://www.pcmag.com>. Diambil kembali dari <https://www.pcmag.com/encyclopedia/term/37067/3d-animation>
- Siagian. (1987). *Penelitian operasional: teori dan praktek*. Jakarta: Penerbit Universitas Indonesia (UI Press).

Sukintaka. (2004). *Teori Pendidikan Jasmani (Filosofi, Pembelajaran dan Masa Depan)*. Bandung: Nuansa.

Syalabi, L. A. (2014). *ANALISIS DAN PEMBUATAN RIG KARAKTER 3D BERDASARKAN STANDAR PRODUKSI FILM ANIMASI DI PT. MSV PICTURES*.

VFX, M. (2014, November 17). *Moving-Picture*. Diambil kembali dari [www.moving-picture.com](https://www.moving-picture.com/advertising/work/moneysupermarket-elephunk/): <https://www.moving-picture.com/advertising/work/moneysupermarket-elephunk/>

