



### **Hak cipta dan penggunaan kembali:**

Lisensi ini mengizinkan setiap orang untuk mengubah, memperbaiki, dan membuat ciptaan turunan bukan untuk kepentingan komersial, selama anda mencantumkan nama penulis dan melisensikan ciptaan turunan dengan syarat yang serupa dengan ciptaan asli.

### **Copyright and reuse:**

This license lets you remix, tweak, and build upon work non-commercially, as long as you credit the origin creator and license it on your new creations under the identical terms.

## BAB II

### TINJAUAN PUSTAKA

#### 2.1. Animasi

Animasi merupakan sebuah manipulasi dari beberapa gambar yang berurutan dan jika dimainkan dengan kecepatan tinggi akan menimbulkan ilusi. Para animator juga bisa disebut sebagai *illusionists*. Di zaman sekarang, terdapat *digital animation* yang telah membantu para animator untuk menciptakan gambar-gambar luar biasa yang sulit untuk dicapai pada zaman dahulu.

*Digital animation* telah mempengaruhi setiap aspek dari produksi animasi modern sekarang ini. *Digital animation tools* sudah tidak lagi hanya tersedia pada studio-studio profesional, sekarang bisa didapatkan dengan mudah (Wyatt, 2010). Ada banyak spesialis pada setiap teknik di dalam *digital animation*, tetapi semua ini dapat dimasukkan ke dalam empat kategori utama, yaitu:

- *Stop Motion*
- *Digital 2D Animation*
- *Digital 3D Animation*
- *Compositing*
- *Digital 3D Animation* menggunakan komputer untuk membuat sebuah geometris yang muncul di dalam dunia 3D, dapat di animasikan ke berbagai arah dan dapat di render dari manapun (Watkins, 2012).

Membuat *Digital 3D Animation* di dalam layar komputer yang 2D merupakan sebuah tantangan, tetapi ketika semua dikuasai akan terbuka *world of possibilities*. Di dalam sebuah animasi tentu terdapat sebuah tokoh yang menjadi penggerak sebuah cerita di dalamnya.

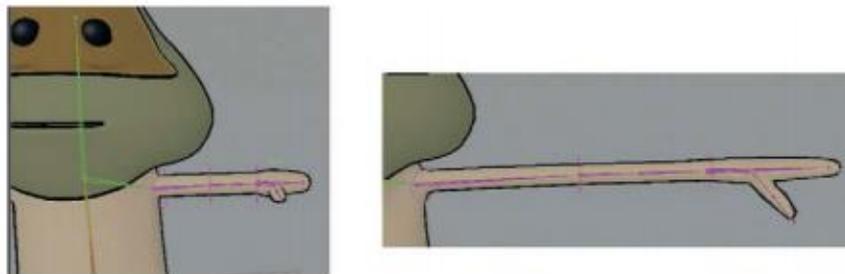
Pada *3D Animation* terdapat sebuah model yang nantinya bisa dijadikan sebagai tokoh ataupun *environment* atau *props*. Sebuah model tersebut dapat diubah bentuknya sebagaimana rupa untuk mendapatkan bentuk seperti yang diinginkan bagaikan sebuah tanah liat yang diubah bentuknya menjadi sebuah bentuk lainnya, yang berbentuk manusia, hewan, alat, dan lain-lain (Wyatt, 2010).

## **2.2. Exaggeration**

Terdapat 12 prinsip animasi yang membuat suksesnya pembuatan animasi, salah satunya adalah *Exaggeration*. *Exaggeration* memiliki peran yang sangat penting namun sering tidak diperhatikan oleh pembuat animasi. Tokoh yang dianimasikan sering terlihat kaku dan seperti robot. Untuk mempelajari *exaggeration* dapat dicapai dengan mencari arti dibalik dari sebuah tindakan. Ketika sudah mengetahui hal tersebut, maka pembuat akan paham bagaimana cara berkomunikasi dengan penonton nantinya. Jika tokoh dalam film mengalami kebahagiaan, maka buatlah tokoh itu terlihat sangat bahagia, sedangkan tokoh yang sedih, buatlah tokoh itu terlihat sangat sedih. Pembuat animasi harus dapat membuat penontonnya merasa tindakan yang dilakukan oleh tokoh dalam film adalah sesuatu tindakan yang nyata. Jika tidak, maka gerakan yang akan muncul terlihat *unnatural* dan kepribadian tokoh akan berkurang (Ratner, 2009).

Semua animasi dibuat demikian rupa untuk dapat dipercaya oleh penontonnya. Tindakan yang terjadi di dunia nyata berbeda dengan tindakan yang ada di dalam animasi. Jika tindakan dalam dunia nyata dibuat dalam animasi tanpa merombaknya, maka hasilnya akan terlihat polos. Oleh karena itu, exaggeration dibutuhkan dalam pembuatan animasi dan tingkatnya bergantung pada seberapa realistis karakter anda. Aktor yang bermain pada live-action tidak berperilaku seperti orang-orang di kehidupan nyata, dan juga tidak seperti animasi (Roberts, 2011).

Salah satu yang mendukung exaggeration adalah tokoh yang dapat stretchy. Stretchy adalah bagian yang penting dalam animasi namun tidak semua animasi membutuhkannya. Stretchy dapat menjadi sulit dan terkadang 1 teknik tidak masuk ke dalam kebutuhan animasi.



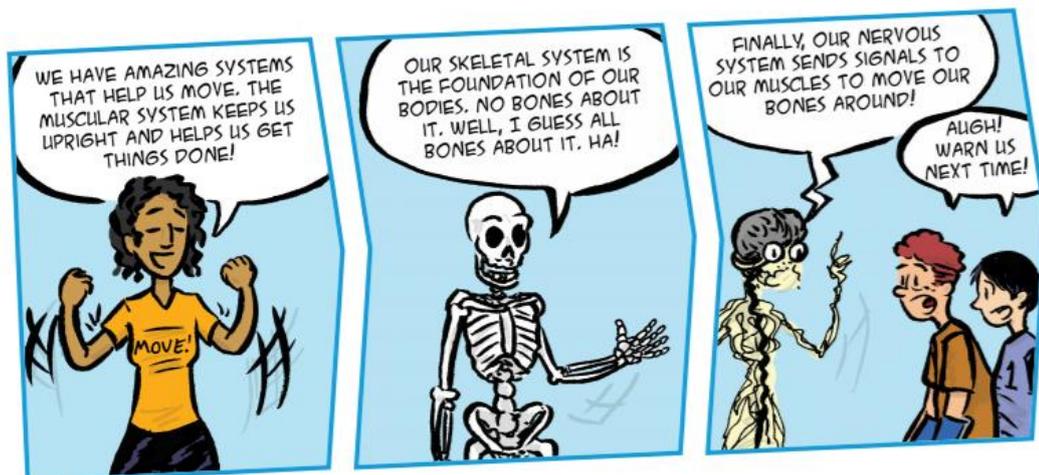
Gambar 2.1. Contoh Gambar *Stretch Arm*  
(Rig it right! Maya Animation Rigging Concepts, 2013)

### **2.3. *Human Body Movement***

Tubuh manusia terbuat untuk menciptakan adanya gerakan. Setiap hari kita menggerakkan bagian-bagian dari tubuh kita. Untuk bergerak, tubuh manusia melewati serangkaian interaksi yang kompleks yang melibatkan sistem tubuh

(Mooney, 2017). Gerakan yang kecil pun membutuhkan adanya koordinasi, komunikasi antar otot, sistem kerangka, dan saraf.

- Sistem Kerangka, kerangka manusia adalah kerangka yang bekerja mendukung tubuh manusia. Tanpa tulang, manusia tidak dapat berdiri, duduk, atau berjalan.
- Sistem Otot, otot terhubung dengan tulang. Ketika otot berkontraksi, otot menghasilkan gerakan di sekitar sendi atau sebaliknya mendukung tubuh dan meningkatkan stabilitas.
- Sistem Saraf, setiap yang dilakukan manusia dikendalikan oleh saraf. Saraf merupakan koneksi antara otak dengan tubuh dengan memberikan sinyal ke otot.

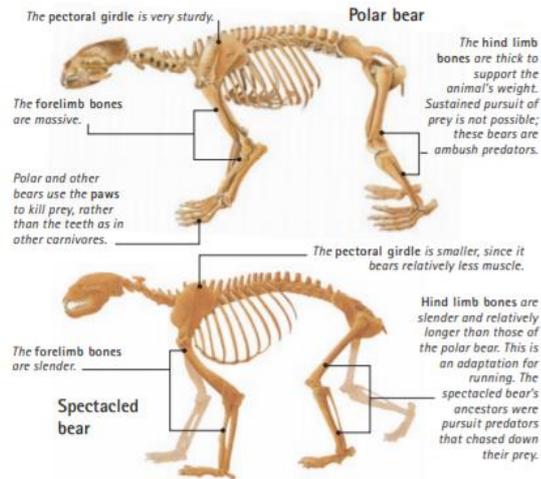


Gambar 2.2. Penjelasan Pegerakan Manusia  
(Human Movement, 2017)

Tulang pada manusia sangat kuat, akan tetapi jika setiap tulang bergabung menjadi satu maka manusia akan susah untuk bergerak. Untuk mengatasi masalah ini maka tulang memiliki sendi, tempat dimana dua tulang bertemu. Dengan adanya sendi, tubuh manusia dapat bergerak dengan berbagai cara seperti, mengangkat tangan, menekuk lengan atau jari-jari, mengangkat kaki, menundukkan kepala, dan lain-lain (Mooney, 2017).

#### **2.4. *Bear Skeletal System***

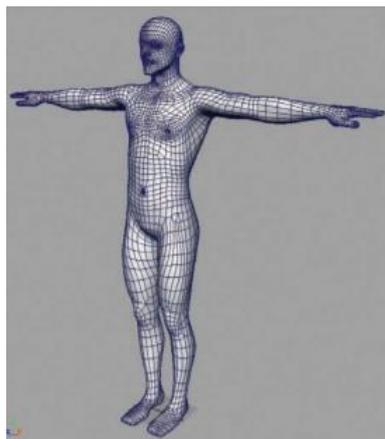
Beruang memiliki kerangka dasar yang dimiliki pada hewan karnivora umumnya yang diadaptasi untuk menopang tubuhnya yang berat. Hewan karnivora lainnya memiliki kerangka yang lebih ringan yang dibangun untuk kecepatan. Jenis karnivora yang dapat berjalan cepat dan gesit memiliki kaki yang panjang dengan tulang yang tipis, akan tetapi beruang memiliki kaki yang relatif pendek dari ukuran tubuh mereka. Tulang pada kaki beruang mampu menopang badan yang begitu besar karena tulang mereka yang sangat tebal. Kebanyakan dari hewan karnivora berjalan menggunakan ujung kaki mereka, sedangkan beruang berbeda, beruang menggunakan seluruh kakinya dari tumit hingga ujung kaki mereka yang menancap ke tanah. Berikut contoh gambar yang memperlihatkan struktur tulang pada beruang (Cavendish, 2010).



Gambar 2.3. Contoh Bentuk dan Penempatan Tulang pada Beruang (Mammal Anatomy: An Illustrated Guide, 2010)

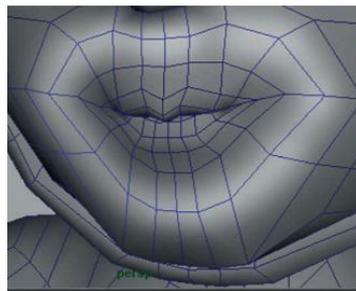
## 2.5. Topology

Sebelum memasuki *rigging*, sebuah tokoh harus memiliki *topology* yang baik terlebih dahulu. *Topology* adalah sebuah alur jalannya *polygon* dalam membentuk sebuah *mesh*. Sebuah bentuk tokoh yang akan di-*rigging* harus mudah diseleksi, dan diubah sesuai dengan yang diinginkan.



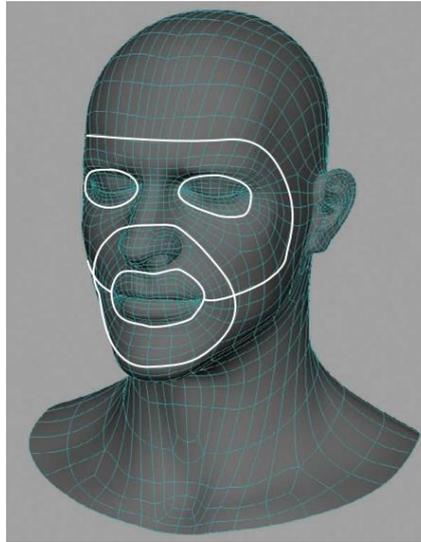
Gambar 2.4. Contoh Topologi Pada Tokoh (3D Animation Essentials, 2012)

Untuk objek yang kaku dan tidak membutuhkan deformasi, seperti senjata, mobil, atau rumah, hanya membutuhkan *topology* yang bersih untuk *texturing* dan *editing*. Akan tetapi, untuk objek yang organik maka membutuhkan *topology* yang baik agar dapat dimanipulasi atau melakukan deformasi sesuai dengan yang diinginkan. *Edgeloop* menjadi hal penting dalam pembuatan *topology* yang baik (Vaughan, 2012). *Edgeloop* adalah deretan garis yang terhubung dari ujung ke ujung mendefinisikan fitur tertentu. Ada contoh gambar *edgeloop* pada mulut (O’Hailey, 2013).



Gambar 2.5. Contoh Gambar *Edgeloop* Pada Mulut  
(Rig It Right! Maya Animation Rigging Concepts, 2013)

*Edgelooping* adalah membentuk permukaan *modeling* agar bergerak mengalir secara alami satu sama lain. Hal ini memastikan kulit berubah bentuk ke arah otot-otot secara alami. Jadi model yang bagus harus memiliki *edgeloop* yang mengitari sepanjang kontur permukaan otot yang sama (O’Hailey, 2013). *Edgeloop* pada wajah mengitari di sekitar mata dan mulut yang meniru otot wajah. Berikut ada contoh gambar *edgeloop* yang baik pada wajah manusia (Beane, 2012).



Gambar 2.6. Contoh *Topology* Pada Wajah  
(3D Animation Essentials, 2012)

## 2.6. *Rigging*

Dalam *3D Animation*, pembuat harus memikirkan apa saja yang dibutuhkan oleh model untuk membuat animasi, mungkin membutuhkan *twist*, *bend*, *stretch*, atau model yang *static*. Setelah model telah jadi dibentuk, model tersebut membutuhkan sebuah tulang (*skeleton*) yang dimasukkan ke dalamnya untuk dapat dianimasikan menjadi pose-pose yang menarik. Proses ini disebut dengan *rigging* (Wyatt, 2010). Membuat tokoh bergerak dengan benar membutuhkan kerangka yang baik. Membutuhkan pemasangan posisi *rig* dan jumlah sambungan antar *rig* yang tepat. *Rig* menyediakan berbagai fungsi, salah satu diantaranya adalah merubah bentuk geometri (Palamar, 2011). *Rigging* merupakan tahapan yang berdasarkan teknik dalam 3D animation, dan *rigging* juga sering dikatakan sebagai tahap persiapan sebelum memasuki tahap animasi dalam 3D (Assaf, 2016). *Rigging* dan animasi adalah dua tahap yang saling bergantung satu sama

lain, tanpa ada yang satunya, maka yang lain tidak akan dapat tampil dengan baik (Beane, 2012).

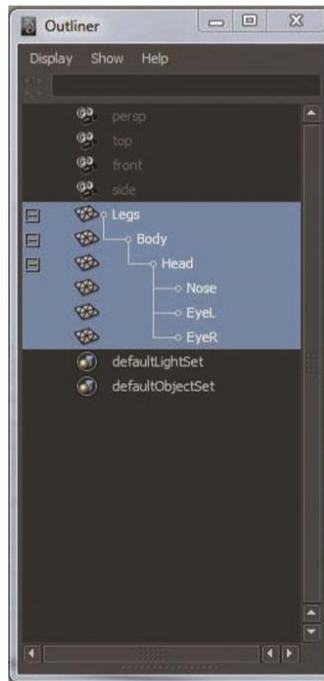
*Rig* sebuah tokoh dibuat dari beberapa tulang dan *controller* sehingga memungkinkan animator untuk memanipulasi dan memberikan pose pada tokoh. Tulang-tulang ini dibentuk dimulai dari tulang panggul dimana sebagian besar gerakan tubuh berasal dan panggul juga merupakan pusat gravitasi manusia. Mekanisme *rig* yang baik harus dapat memungkinkan sebuah tokoh untuk membengkokkan tubuhnya secara wajar. Mekanisme ini merupakan sebuah tugas kompleks yang meliputi gabungan dari beberapa tulang, sendi, deformasi, *surfaces*, *kinematics*, dan *blendshape*.

Jumlah tulang yang akan dimasukkan ke dalam tokoh bergantung pada seberapa detil pergerakan yang akan dibuat. Setelah semua tulang telah dibuat dan disambungkan, selanjutnya adalah membuat batasan terhadap beberapa pergerakan seperti pada tulang leher untuk membatasi agar leher tidak berputar 360 derajat (Wyatt, 2010).

### **2.6.1. Parenting**

Semua *rig* didasarkan pada sistem hierarki dan kontrol yang bekerja secara berurutan untuk membuat artikulasi objek anda. *Parenting* merupakan bentuk hierarki paling dasar yaitu hubungan antara orang tua dan anak, satu objek sebagai orang tua atau induk, dan yang lainnya sebagai anak. Objek yang sebagai anak dapat bergerak, berputar, *scale* secara bebas tanpa terikat dari objek induk, tetapi ketika objek induk digerakkan maka objek anak akan mengikutinya. Dalam

parenting ini dapat memiliki beberapa anak dibawah orang tua seperti yang disebut saudara kandung, dan bahkan dapat memiliki anak-anak dari anak-anak. Salah satu contohnya sebagai berikut.



Gambar 2.7. Contoh Hierarki Parenting

(3D Animation Essentials, 2012)

“*Legs*” merupakan orang tua dari “*Body*” yang memiliki anak “*Head*”, dan memiliki anak lagi “*Nose*”, “*EyeL*”, “*EyeR*”. Di sebagian besar aplikasi 3D, pembuat dapat mengubah hubungan antara orang tua dan anak ini kapan saja. Pembuat dapat mengelompokkan objek satu sama lain, atau dapat mengelompokkan beberapa objek untuk bergerak sebagai satu unit. Pengelompokkan ini biasa disebut sebagai *grouping* (Beane, 2012).

### 2.6.2. *Constraints*

*Constraints* adalah sebuah sistem yang memungkinkan untuk mengontrol sebuah objek dengan objek lain. Dengan *constraints*, pembuat dapat menghubungkan antara *translation* sebuah objek dengan objek lain atau *rotate*, atau *scale*, bahkan dari surface sebuah objek. *Constraints* memiliki persamaan dengan *parenting* tetapi terdapat beberapa perbedaan.

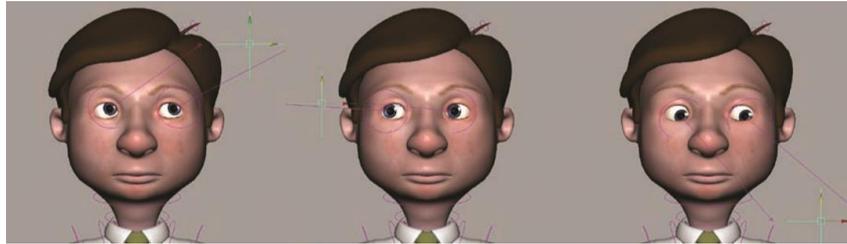
*Constraints* memberikan kontrol lebih kepada pembuat *rig* sehingga cara ini merupakan solusi yang tepat untuk membentuk sebuah koneksi (Beane, 2012). Banyak dari perangkat 3D yang menggunakan beberapa *constraints* dibawah ini, yaitu:

- **Point**

Merupakan sebuah contoh *constraints* yang dapat menghubungkan *translation* objek dengan objek yang lain dan *constraints* ini memungkinkan untuk menghubungkan lebih dari 2 objek (Beane, 2012).

- **Aim**

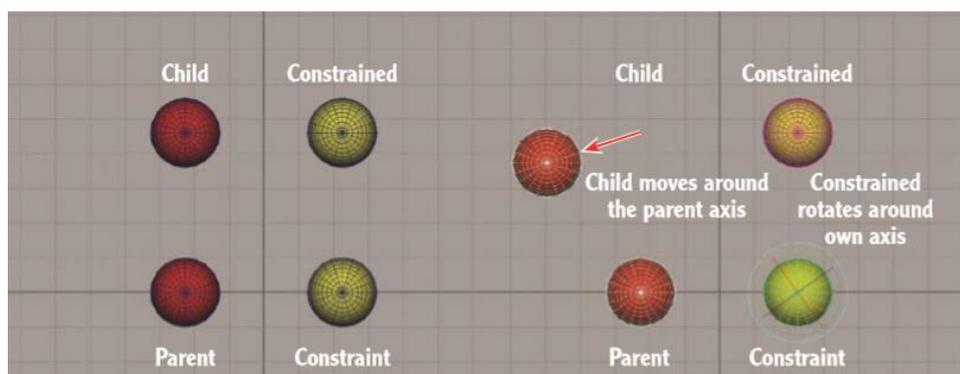
*Constraints* yang dapat menghubungkan *rotation* objek dengan objek lain. Salah satu contohnya adalah mata yang diam di satu posisi dan digerakkan untuk menghadap sebuah *locator* atau *controller*, seperti pada gambar dibawah ini (Beane, 2012).



Gambar 2.8. Contoh *Constraints Aim* pada Mata  
(3D Animation Essentials, 2012)

- **Orient**

Sedikit berbeda dengan *Aim*, *constraints* ini dapat menghubungkan *rotation* objek dengan *rotation* objek lain. *Constraints* ini juga memiliki kemiripan dengan sistem *parenting*, persamaannya adalah ketika memutar objek induk, maka objek anak akan mengikutinya. Perbedaannya dalam *parenting*, ketika memutar objek induk, objek anak akan ikut berputar mengikuti poros induknya, tetapi dalam *constraints* ini ketika memutar objek induk, objek anak akan ikut berputar juga tetapi tidak mengikuti poros induknya melainkan poros dirinya sendiri (Beane, 2012).



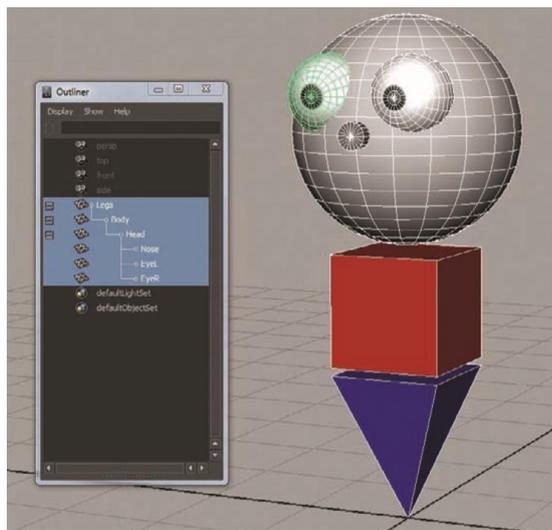
Gambar 2.9. Contoh Perbedaan *Parenting* dan *Constraint Orient*  
(3D Animation Essentials, 2012)

- **Scale**

*Constraints* ini memungkinkan pembuat untuk menghubungkan *scale* objek dengan *scale* objek lain. Dengan kata lain, *scale* sebuah objek mengatur *scale* dari objek lain atau beberapa objek yang lain (Beane, 2012).

### 2.6.3. *Pivot Points*

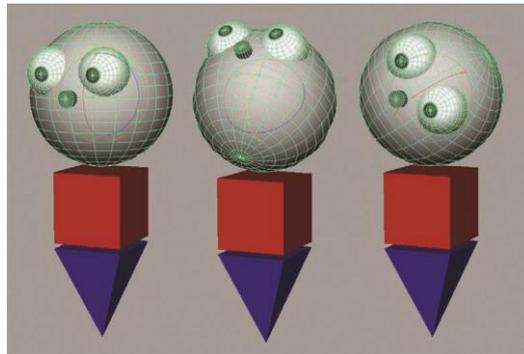
Posisi pivot atau titik pivot adalah lokasi dimana objek berputar. Dalam animasi 3D, titik pivot juga merupakan posisi dimana manipulator lain memindahkan atau *scale* sebuah objek. Mengatur posisi pivot yang benar untuk menciptakan titik manipulasi yang benar adalah hal yang termasuk penting dalam membuat sebuah *rig*. Berikut terdapat contoh hierarki *rig* dasar.



Gambar 2.10. Contoh Hierarki *Rig* Dasar  
(3D Animation Essentials, 2012)

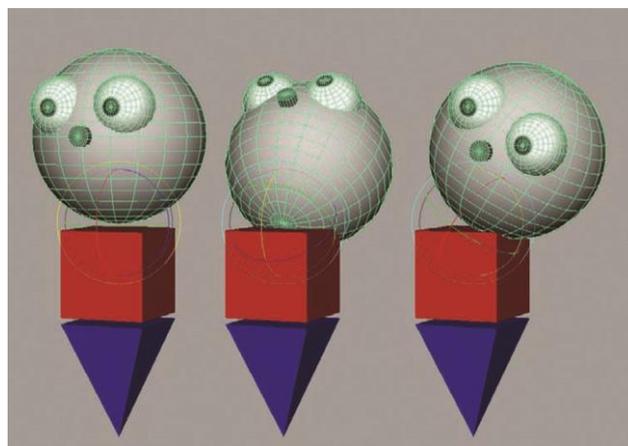
*Rig* di atas hanyalah *rig* dasar hubungan antara orang tua dengan anak dan semua bagiannya yang terhubung satu sama lain. Untuk menggerakkan kepala dengan cara yang tepat agar sesuai dengan rotasi kepala pada umumnya, titik pivot harus berada mendekati titik yang berhubungan dengan tubuh, biasanya

adalah bagian leher. Namun, titik pivot *default* bola ada pada pusatnya, jadi tidak akan dapat memutar kepala seperti pada umumnya, melainkan berputar seperti pada gambar di bawah ini.



Gambar 2.11. Contoh *Rotate* pada Titik Pivot *Default*  
(3D Animation Essentials, 2012)

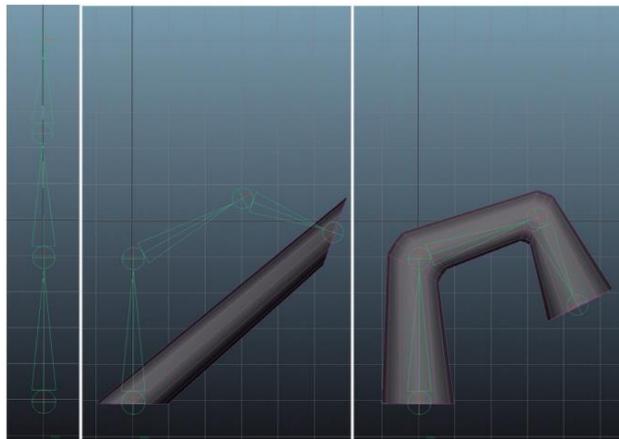
Untuk mengatasi kejadian di atas, pembuat dapat memindahkan titik pivot tersebut ke bawah (sekitar bagian leher) dengan cara menekan tombol “d” atau “insert” sambil memindahkan pivot tersebut. Dengan memindahkan titik pivot ke bawah, maka pembuat dapat menggerakkan kepala dengan tepat seperti pada umumnya (Beane, 2012).



Gambar 2.12. Contoh *Rotate* pada Titik Pivot yang Tepat  
(3D Animation Essentials, 2012)

#### 2.6.4. *Skeleton*

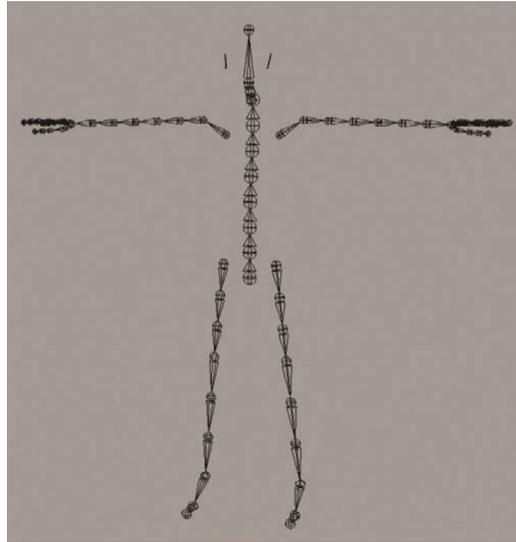
Di dalam rig terdapat *skeleton, joints*, yang terdiri dari satu set hierarki yang dapat dipilih dan digunakan untuk mendefinisikan deformasi tertentu (Beane, 2012). *Joint* berguna untuk memungkinkan deformasi terhadap suatu objek dan *joint* sendiri membutuhkan *edges* dan *vertex* pada objek untuk melakukan deformasi (Watkins, 2012).



Gambar 2.13. Contoh Gambar *Influence Joint*  
(Getting Started in 3D with Maya, 2012)

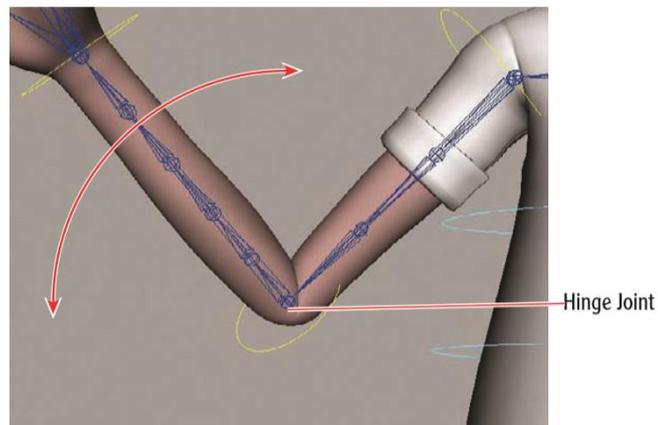
*Mesh* sendiri tidak bengkok, satu-satunya cara untuk membengkokkan *mesh* adalah titik *vertex* dan *edges* dalam *mesh* tersebut. Pada gambar tengah, tidak ada *edges* dan *vertex* pada badan *mesh* sehingga *mesh* tidak dapat ikut bengkok, dan pada gambar sebelah kanan, *mesh* yang terdapat *edges* dan *vertex* dapat mengikuti pergerakan *joint* (Watkins, 2012).

*Skeleton* ini adalah bentuk dasar untuk rig tokoh tingkat menengah dan hingga untuk yang sudah mahir. *Joints* bisa digunakan untuk model *non-organic* seperti mobil dan senjata yang berguna untuk memudahkan pemilihan dan melakukan deformasi (Beane, 2012).



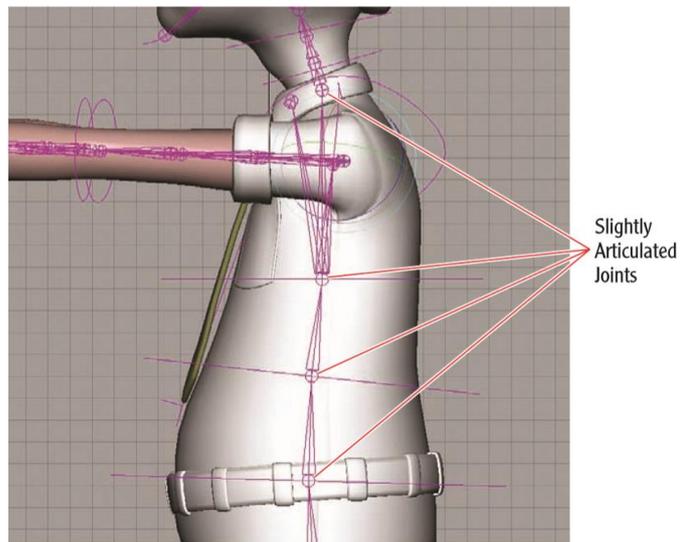
Gambar 2.14. Contoh Bentuk *Skeleton* Manusia  
(3D Animation Essentials, 2012)

Ada 4 contoh jenis gerakan yang dapat digerakkan melalui *joints*, yaitu sendi engsel (*Hinge joint*) yang berputar pada satu sumbu, seperti lutut atau siku.



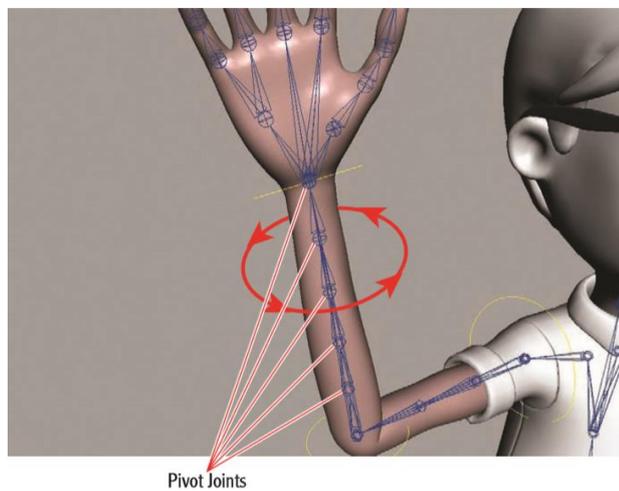
Gambar 2.15. Contoh *Hinge Joint*  
(3D Animation Essentials, 2012)

*An articulated joints* dapat memutar sedikit, tetapi banyak dari sambungan tulang-tulang ini yang berputar sekaligus untuk menciptakan berbagai macam gerakan, seperti pada tulang belakang.



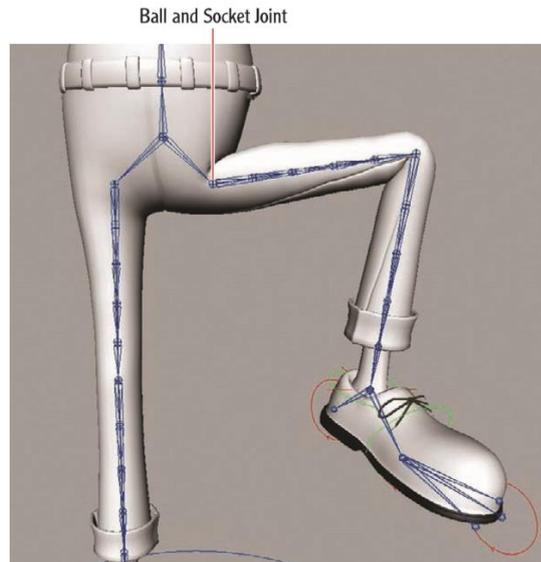
Gambar 2.16. Contoh *Articulated Joints*  
(3D Animation Essentials, 2012)

A *pivot joint* yang dapat memutar sekitar satu sumbu, contohnya seperti pergelangan tangan.



Gambar 2.17. Contoh *Pivot Joints*  
(3D Animation Essentials, 2012)

A *ball-and-socket joint* dapat berputar ke berbagai arah, seperti bahu atau panggul.



Gambar 2.18. Contoh *Ball and Socket Joint*  
(3D Animation Essentials, 2012)

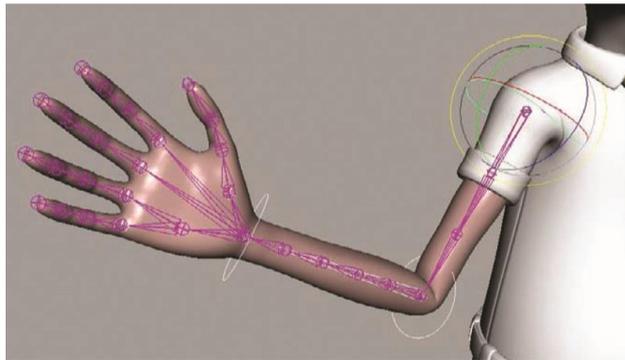
Semua jenis-jenis *joints* ini dibuat sesuai dengan kebutuhan animasi yang ingin dicapai (Beane, 2012).

- Forward dan Inverse Kinematics

Tulang-tulang yang dibuat akan dapat dimanipulasi dengan baik dan membentuk struktur yang baik. Pergerakan tulang digerakkan dengan *kinematics*. Terdapat 2 *kinematics* dalam pergerakan tulang yaitu, *Forward Kinematics* (FK), dan *Inverse Kinematics* (IK) (Wyatt, 2010). Kedua jenis pergerakan tulang ini memberikan cara yang berbeda kepada animator untuk mencapai sebuah animasi tertentu.

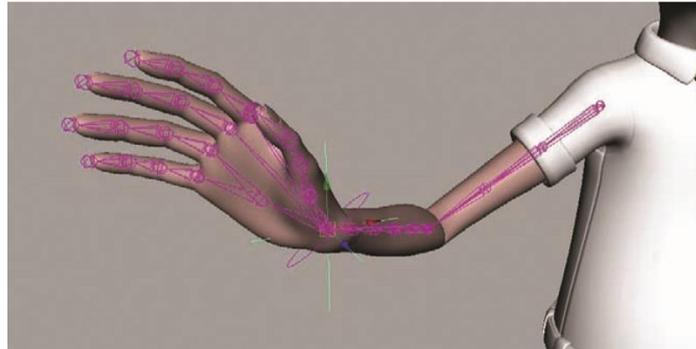
*Forward Kinematics* adalah gerakan yang paling natural untuk menggerakkan tulang tokoh, dan bekerja menurun sesuai hierarki. Jika tokoh ingin memegang sebuah benda, maka pembuat akan melakukan *rotate* pada bahu tokoh, lalu sikut tokoh, lalu pergelangan tangan (Wyatt, 2010). Hal itu semua terjadi karena *Forward Kinematics* didasarkan pada sambungan lengan yang

ditarik dan dihubungkan sesuai urutan bahu, sikut, dan pergelangan tangan. Berikut contoh gambar *Forward Kinematics* pada bagian tangan, animator akan memutar setiap lingkaran untuk mengontrol bahu, sikut, lalu pergelangan (Beane, 2012).



Gambar 2.19. Contoh *Forward Kinematics*  
(3D Animation Essentials, 2012)

Sedangkan *Inverse Kinematics*, bekerja menaiki hierarki berlawanan dengan *Forward Kinematics*, jika pergelangan tangan tokoh digerakkan ke posisi tertentu maka sikut, dan bahu tokoh akan ikut bergerak secara langsung oleh komputer. Namun gerakan yang dihasilkan oleh *Invers Kinematics* ini tidak dapat ditebak karena komputer akan membaca pergerakan yang langsung sehingga memungkinkan sikut tokoh bengkok ke arah yang salah (Wyatt, 2010). *Inverse Kinematics* membalikkan hierarki lengan yang aslinya sehingga memungkinkan animator untuk bekerja dengan cara yang berbeda. Ada contoh gambar untuk *Inverse Kinematics*, animator menggerakkan pergelangan tangan untuk menggerakkan keseluruhan tangan tokoh secara langsung (Beane, 2012).



Gambar 2.20. Contoh *Inverse Kinematics*  
(3D Animation Essentials, 2012)

Forward Kinematics dan Invers Kinematics dapat digunakan pada tangan, kaki, tubuh, ekor, atau sayap, akan tetapi beberapa animasi ada yang membutuhkan tipe yang spesifik antara FK atau IK. Contohnya, banyak dari kaki tokoh menggunakan IK untuk kaki tetap berada di bawah tanah saat tubuh bergerak. Dan juga ketika tokoh yang menaruh tangannya di atas meja, maka IK adalah pilihan yang tepat karena tangan tokoh dapat diam di atas meja selagi tokoh bergerak. Hal itu semua terjadi karena jika menggunakan FK, maka ketika tubuh digerakkan, tangan dan kaki akan ikut bergerak mengikuti tubuhnya. Tangan dan kaki tidak bisa diam di satu tempat. Untungnya kebanyakan dari *rig* tokoh dapat mengubah antara FK dan IK, jadi hal ini akan mempermudah animator untuk menggerakkan tokoh sesuai dengan yang diinginkan (Beane, 2012).

#### **2.6.5. Deformasi**

Setelah membuat sistem kerangka, perlu untuk menerapkan deformasi pada geometri untuk membuat geometri mengikuti kerangka yang telah dibuat. Deformasi kebanyakan memungkinkan pembuat untuk mengatur opsi pada bagian

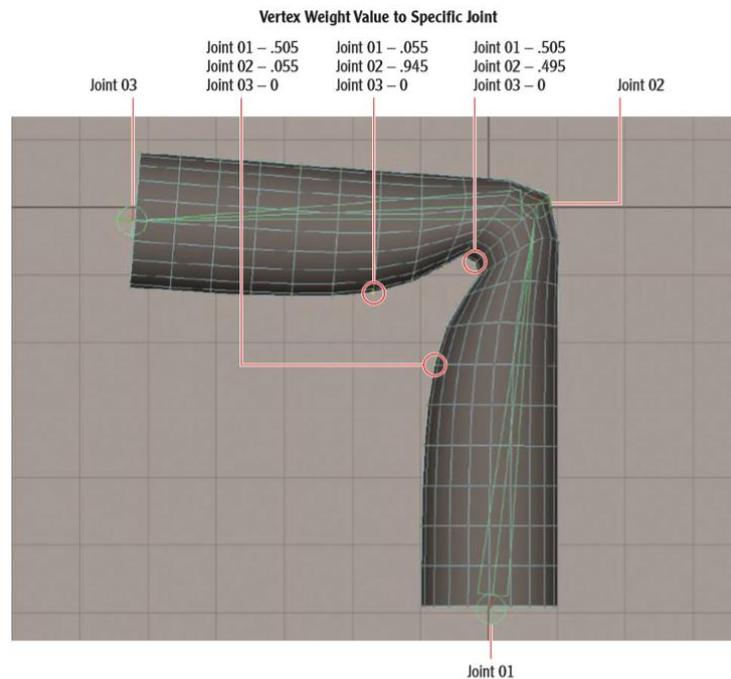
komponen geometri bukan satu keseluruhan bagian objek. Komponen geometri adalah bagian-bagian individu yang terdapat dalam sebuah objek, seperti *vertices*, *edges*, dan *faces*. Terdapat banyak jenis deformasi yang tersedia saat ini, dan tidak semua perangkat 3D memiliki tipe yang sama. Namun, ada beberapa tipe umum yang terdapat pada perangkat 3D biasanya adalah *skinning* atau *enveloping*, *lattice*, dan *blendshapes* (Beane, 2012).

- **Skinning atau Enveloping**

*Skinning* atau *enveloping* merupakan salah satu bentuk deformasi yang memungkinkan pembuat *rig* untuk memberikan isi pada setiap vertex objek geometri yang dipengaruhi oleh setiap sambungan hierarki *skeleton*. Ini memungkinkan *mesh* untuk bergerak di atas *skeleton*. *Mesh* ini biasa disebut juga sebagai *skin*. Banyak perangkat 3D yang memiliki berbagai jenis *skinning*, diantaranya *smooth skinning* and *rigid skinning* (Beane, 2012).

- **Smooth Skinning**

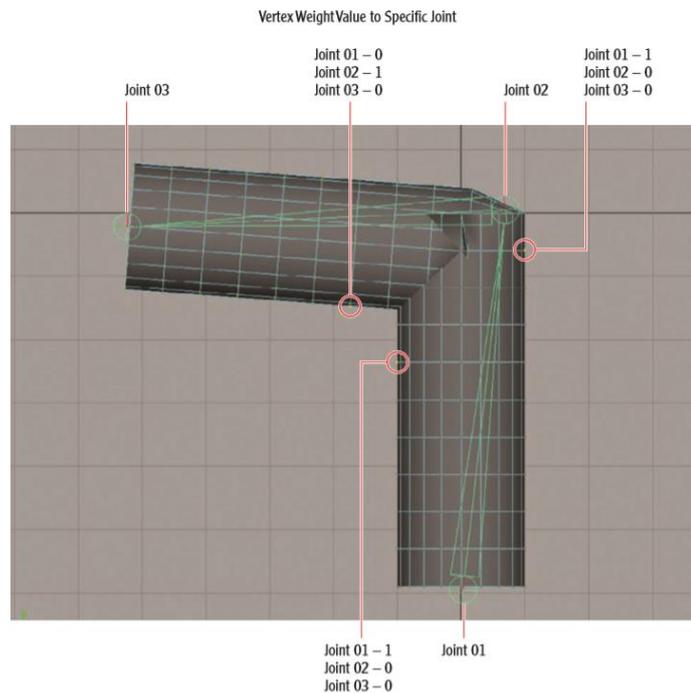
*Smooth skinning* memungkinkan pembuat untuk mengatur isi pada setiap *vertex* dengan beberapa nilai pada *joints* yang berbeda dan memungkinkan untuk melakukan deformasi yang halus. *Smooth skinning* adalah metode *skinning* yang paling banyak digunakan saat ini. *Skinning* ini digunakan untuk hampir semua jenis objek organik seperti pada tokoh, makhluk, tanaman, dan tabung atau kabel. Contoh sebagai berikut (Beane, 2012).



Gambar 2.21. Contoh *Smooth Skinning*  
(3D Animation Essentials, 2012)

- **Rigid Skinning**

Dalam *Rigid skinning*, hasil yang dicapai akan terlihat kaku. Hal tersebut disebabkan karena di dalam *skinning* ini hanya memberikan semua nilai atau tidak ada nilai sama skali kepada *vertex*. Artinya hanya ada 100 persen atau 0 persen yang dapat diberikan kepada *vertex*. *Rigid skinning* secara default akan menciptakan deformasi yang kaku seperti pada gambar di bawah ini.



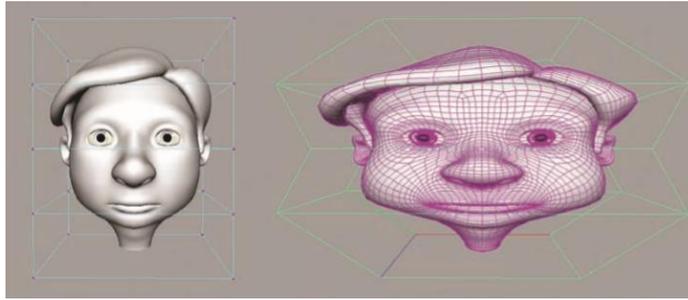
Gambar 2.22. Contoh *Rigid Skinning*

(3D Animation Essentials, 2012)

*Rigid skinning* dalam hal teknis memang bekerja lebih cepat dari pada *smooth skinning*, tetapi *rigid skinning* lebih sulit dalam pengerjaannya untuk mencapai deformasi yang halus. *Rigid skinning* ini dapat digunakan untuk jenis objek yang sama seperti *smooth skinning* (misalnya tokoh, makhluk, dan kehidupan tanaman), tetapi dalam *rigid skinning* diperlukan deformer lain untuk mencapai transisi geometri yang halus (Beane, 2012).

- **Lattice**

*Lattice* merupakan bentuk sangkar yang membungkus *mesh* geometri yang lebih padat dan mengubah bentuk *mesh* dengan halus seperti pada gambar di bawah ini.

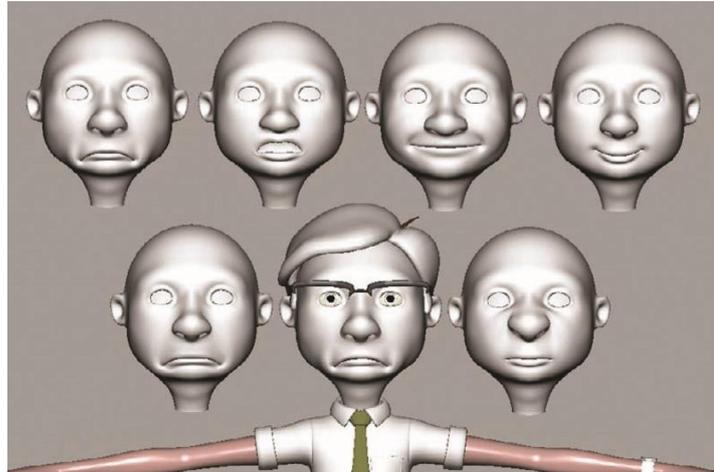


Gambar 2.23. Contoh *Lattice*  
(3D Animation Essentials, 2012)

Akan tetapi, *lattice* dapat merusak *mesh* atau beberapa *mesh* sekaligus. Deformasi ini merupakan *deformer* yang kuat dan berguna karena dapat digunakan dalam berbagai cara. Sebagai contoh, pembuat bisa membuat sebuah pagar lurus, akan tetapi jika pagar tersebut ingin dibuat memiliki tampilan seperti *cartoon straight-line*, maka bisa menggunakan *lattice* dan mendeformasi pagar tersebut (Beane, 2012).

- **Blendshape**

*Blendshape* adalah bentuk deformasi yang memungkinkan pembuat untuk membuat objek, contohnya kepala dan kemudian menduplikasi objek tersebut dan mengubahnya (misalkan menambahkan ekspresi senyum ke wajah) untuk membuat *target shape*. *Blendshape* menghubungkan kedua objek bersamaan dan memungkinkan bentuk asli berubah menjadi *target shape* tadi (ekspresi senyum di wajah). *Blendshape* juga dapat digunakan dengan lebih dari satu *target shape*. Berikut contohnya.



Gambar 2.24. Contoh *Blendshape*  
(3D Animation Essentials, 2012)

Blendshapes juga dapat digunakan untuk memperbaiki *skinning* di tempat-tempat seperti siku dan bahu, dan dapat diotomatiskan untuk memicu ketika tokoh berada pada gaya tertentu, untuk memperbaiki deformasi yang buruk (Beane, 2012).

## 2.7. *Facial Expression*

Wajah adalah bagian tubuh manusia yang dapat dibedakan. Wajah digunakan untuk mengenali orang-orang yang dikenal dan tidak kenal. Aktor yang dibuat harus mampu menunjukkan ekspresi yang natural agar tidak terlihat aneh karena penonton mudah mengenali keganjalan jika terjadi kesalahan (Palamar, 2011). Sebagai animator tentu ingin membuat tokoh yang dibuat memiliki sebuah ekspresi agar terlihat hidup. Ekspresi tokoh merupakan hal terpenting untuk setiap tokoh dalam setiap cerita. Emosi merupakan inti dari setiap keputusan, tindakan, dan kata setiap karakter, yang semuanya bertujuan untuk menggerakkan cerita. Tanpa emosi, perjalanan pribadi karakter tidak ada maknanya. Sebagai makhluk

emosional, perasaan mendorong kita. Mereka mengarahkan pilihan kita, menentukan dengan siapa kita akan menghabiskan waktu, dan menentukan nilai kita. Dalam animasi, penonton juga ingin menonton pertunjukkan yang bagus untuk dapat terhubung dengan tokoh yang telah menyediakan hiburan, dan permasalahan yang dialami oleh tokoh akan menambah makna bagi perjalanan hidup penonton (Ackerman & Puglisi, 2012).

*Facial Expression* adalah cara bagaimana kita berkomunikasi satu sama lain. Untuk membuat *Facial Expression* ke dalam tokoh, pembuat harus mengerti bagaimana wajah itu bekerja terlebih dahulu. Wajah dapat membuat banyaknya ekspresi yang berbeda-beda untuk menyampaikan perasaan manusia yang kompleks (Wyatt, 2010). Hal itu semua terjadi karena ada banyaknya otot pada wajah yang jumlah pastinya masih diperdebatkan.

Otot-otot wajah yang beraneka ragam bergabung dan dapat melakukan berbagai tindakan. Pada akhirnya jumlah pastinya tidak relevan, sehingga hanya dilihat dari gerakan ekspresi yang serupa. Otot terdiri dari tiga bagian. Pertama adalah *anchor point*, berfungsi untuk menarik otot kepada posisi yang tetap. Kedua adalah *muscle belly*, bagian otot yang melakukan pengerutan atau pengencangan. Ketiga adalah *insertion*, bagian otot yang terhubung ke jaringan lain (Palamar, 2011).

Agar tokoh yang dibuat terlihat hidup tentu harus mempunyai ekspresi yang sesuai dengan keadaan situasi atau aksi yang akan dilakukannya. Untuk membuat itu maka pembuat harus menyederhanakan ekspresi wajah yang

kompleks ini. Mempelajari ekspresi wajah dapat dengan menonton film yang diperankan oleh aktor yang hebat ataupun melihat kaca untuk melihat ekspresi kita sendiri (Wyatt, 2010).

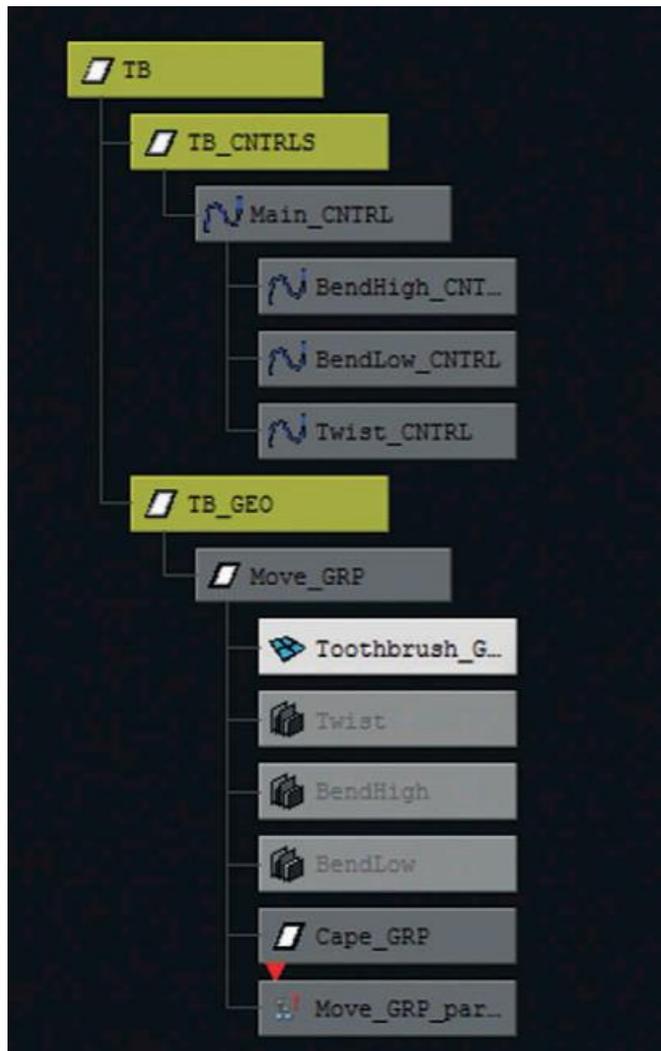
Sebagai animator, pembuat harus melakukan observasi akan *Facial Expression* dengan baik. Penonton telah memiliki harapan yang tinggi sebelum menonton. Penonton tidak ingin diberitahu bagaimana perasaan tokoh, melainkan penonton ingin mengalami emosi yang dialami tokoh untuk diri mereka sendiri. Untuk dapat mewujudkannya, pembuat harus memastikan bahwa tokoh mengekspresikan emosinya dengan cara yang mudah dikenali dan menarik untuk dijadikan tontonan. Animasi yang baik memiliki tokoh yang dapat memperlihatkan emosi atau membuat jalan cerita berjalan dengan hanya menampilkannya saja tanpa menjelaskan sesuatu. Menampilkan memang lebih sulit daripada memberi tahu secara langsung, namun itu semua sangat berarti karena dapat membuat penonton lebih dekat dengan tokoh dan bisa menimbulkan rasa empati. Ketika membuat sebuah emosi, pikirkan bagaimana jika keadaan atau situasi tersebut terjadi pada diri pembuat sendiri. Yang kedua adalah mengenali sifat dari tokoh yang akan dibuat. Manusia memiliki bermacam-macam kepribadian, dan di setiap kepribadian dalam merespon sebuah kejadian itu berbeda-beda. Setiap kepribadian itu unik karena sebagian besar dipengaruhi oleh peristiwa dari masa lalu. Hal ini merupakan salah satu cara untuk mendapatkan empati dari penonton, dengan menunjukkan bagaimana tokoh menjadi dirinya yang sekarang ini. Apapun yang akan tokoh alami, tunjukkan sebuah emosi yang

spesifik mungkin dan buat ekspresi yang mudah untuk dikenali oleh penonton (Ackerman & Puglisi, 2012).

Titik fokus pada ekspresi wajah adalah ekspresi mata. Ekspresi mata adalah yang pertama kali penonton akan lihat ketika dihadapi dengan *shot close-up*. Ketika melihat orang lain, terutama yang akan dilihat adalah mata untuk melakukan emosi. Alis dan mulut juga sangat penting pada ekspresi wajah dalam mengkomunikasikan emosi tertentu (Wyatt, 2010).

## **2.8. Finalisasi**

*Rigger* harus menyumbangkan waktu untuk memnbersihkan hasil dari *rigging* yang telah dibuat. Kegunaan untuk melakukan ini adalah untuk tidak membuat animator melihat hal-hal yang tidak dimaksud untuk digerakkan secara langsung dan ada beberapa sistem yang dapat merusak *rig* jika digerakkan. Oleh karena itu, lebih baik semua *attributes* yang tidak digunakan saat animasi disembunyikan dan di-*lock* agar tidak bergerak. Kita tidak menginginkan animator menyentuh sistem *rigging* dan merusaknya seperti *IK handles*, *skeletons*, dan lain-lain (O’Hailey, 2013).



Gambar 2.25. Contoh Merapihkan Outliner  
(Rig It Right! Maya Animation Rigging Concepts, 2013)