

2. STUDI LITERATUR

Game 3D

Gunter dan Kenny (seperti dikutip dalam Tanjung, 2013) menyatakan bahwa *game* 3D merupakan *game* atau permainan yang menggunakan grafik tiga dimensi (3D). Dimensi berupa angka-angka yang didasarkan pada topologi atau sifat metrik sebuah objek. Dimensi pada grafik *game* 3D yaitu gambaran data geometris yang tiga dimensional, seperti X, Y, dan Z. Data-data ini akan tersimpan pada komputer, kemudian digunakan untuk menghitung dan melakukan *render* pada sebuah sketsa atau ilustrasi dua dimensi (2D). Objek yang dipakai sebagai *asset game* 3D dapat dilihat dari segala sudut pandang (360°), berbeda dengan *asset-asset game* 2D yang hanya dapat dilihat dari arah-arah tertentu (depan, samping, belakang). Pada *game* 3D ini, *asset-asset* dapat dibuat melalui tahap *3D modelling*, *3D rendering*, serta *3D computer graphics software* (hlm. 2).

Digital Sculpting

Wirawan (seperti dikutip dalam Sonjaya & Zahra, 2017) mengungkapkan teknik *3D modelling* baru yaitu *digital sculpting*. *Digital sculpting* adalah teknik *3D modelling* yang dilakukan dengan memahat model 3D. Hal ini menyebabkan pahatan objek 3D memiliki resolusi poligon dan tingkat kehalusan yang sangat tinggi. Dengan demikian, teknik ini sering digunakan untuk memperlihatkan detail-detail pada objek 3D. *Digital Sculpting* dikerjakan di *software* 3D biasa (*Maya*, *3DS Max*, *Blender*, dan lain-lain) serta *software* 3D khusus *sculpting* seperti *ZBrush* (hlm. 38). Beane (seperti dikutip dalam Maharani, 2017) menambahkan bahwa permukaan model 3D dapat dibentuk dan diatur dengan dorongan dan tarikan seperti bahan tanah liat (hlm. 16).

Mesh

Mesh merupakan bentuk utuh dari sebuah objek 3D yang mempunyai volume, bentuk, dan ruang. Tiga elemen yang membentuk *mesh* ini yaitu *vertex*, *edge*, dan *face*. *Vertex* adalah titik-titik yang berada di koordinat X, Y, dan Z. Dua *vertex* yang berdekatan akan dihubungkan dengan sebuah garis lurus bernama

edge. *Edge* dapat dilihat sebagai suatu garis yang dimiliki oleh sebuah objek 3D. *Face* terbentuk dari tiga *vertex* dan *edge* yang saling terhubung sehingga *face* merupakan bidang permukaan dengan kurva yang tertutup. Kelompok-kelompok *vertex*, *edge*, serta *face* yang disatukan akan menciptakan *mesh* objek 3D (Fadya & Sari, 2018, hlm. 44).

Poligon

Poligon (*polygon*) atau *face* merupakan salah satu elemen utama yang membentuk sebuah objek 3D. Dalam suatu model 3D, poligon tampak sebagai suatu bidang dengan jumlah sisi tertentu yang tercipta dari gabungan kumpulan *vertex* dan *edge*. Poligon model 3D memiliki tiga tipe yaitu *quad*, *tris*, dan *n-gon*. Jenis poligon yang paling umum digunakan adalah *quad* yang mempunyai empat sisi. Tiga garis (*edges*) yang bersambungan akan membentuk jenis poligon *tris*. *Tris* merupakan tipe poligon yang digunakan pada sebuah *game engine*. Sementara itu, poligon dengan lebih dari empat sisi akan disebut dengan *n-gon* (Martyastiadi dkk., 2015, hlm. 51-52).

Efisiensi Jumlah Poligon terhadap Mesh

Efisiensi jumlah poligon (*poly-count*) merupakan aspek yang sangat dibutuhkan dalam pembentukan model 3D pada suatu *game*. Hal ini dikarenakan *game engine* tidak kuat untuk melakukan *real-time rendering* pada model yang terlalu *high-poly*. Padahal, jumlah poligon akan terus mengalami peningkatan dalam suatu proses *3D modelling*. Oleh sebab itu, model dengan *poly-count* rendah (*low-poly*) diperlukan untuk meringankan kerja *video card* dan *rendering* (Martyastiadi dkk., 2015, hlm. 52). Terävä (2017) menjelaskan bahwa *poly-count* dapat digunakan untuk menentukan suatu model *low-poly*. Namun, bentuk poligon *tris* pada *game engine* menyebabkan penggandaan perhitungan poligon *quad* karena *quad* terdiri dari dua segitiga yang bersatu. Definisi model *low-poly* bisa menjadi subjektif berdasarkan keperluan model 3D dan kemampuan kerja komputer pada perangkat yang dituju. Komputer modern dan *game consoles* bisa menampilkan ratusan ribu hingga jutaan *tris* secara *real-time*. Sebaliknya,

perangkat mobile dan handheld game consoles memiliki keterbatasan kemampuan perangkat (hlm. 17).

Workflow* Perubahan Model 3D *High-Poly* Menjadi *Low-Poly* di *ZBrush* dan *Maya

Workflow perubahan model 3D *high-poly* menjadi *low-poly* dipraktikkan agar model versi *low-poly* akan tetap memiliki detail-detail visual yang hampir sama dengan model versi *high-poly*. Detail-detail model *low-poly* direpresentasikan dengan manipulasi tangkapan cahaya di atas permukaan model. Hal ini dilakukan karena jutaan poligon pada *sculpting* model *high-poly* di *ZBrush* akan memperlambat kerja *real-time rendering*. Terdapat empat langkah dalam memperkecil jumlah poligon pada model *sculpting high-poly* hingga menjadi model yang lebih *low-poly*. Keempat langkah tersebut yaitu pembuatan model *high-poly*, duplikasi dan pengurangan model *high-poly*, pembuatan model *low-poly*, serta metode alternatif untuk penciptaan model *low-poly*.

Pembuatan model *high-poly* dapat dimulai dengan penciptaan *base mesh* di *Maya*, kemudian *detailing* model dapat dibentuk ketika dimasukkan ke *ZBrush*. Setelah dilakukan *subdivide* beberapa kali, jumlah poligon akan mencapai jutaan poligon karena resolusi tinggi diperlukan untuk mengukir detail-detail halus. Setelah itu, duplikasi model *high-poly* harus diterapkan sebelum pengurangan jumlah poligon agar hasil duplikasi model *sculpting* tidak ikut mengalami modifikasi. Untuk mengurangi jumlah poligon dengan mempertahankan detail-detail halus, *Decimation Master* pada *palette ZPlugin* dapat dipakai. Model *high-poly* yang telah dikurangi tersebut akan diekspor dalam bentuk *file OBJ*.

Pada tahap pembuatan model *low-poly*, beberapa tingkatan *Subdivision* pada *palette Geometry* dapat digunakan untuk mengekspor versi *low-poly* dari *sculpting* model *high-poly*. Semakin tinggi tingkatan *Subdivision*, semakin mirip bentuk dan ukuran model dengan model *high-poly*. Model versi *low-poly* ini kemudian diekspor dalam bentuk *file OBJ*. Metode kedua untuk menciptakan model *low-poly* yaitu pengaplikasian *ZRemesher* pada hasil duplikasi model *high-*

poly. *ZRemesher* ditemukan pada *subpalette Geometry* agar model dapat di-*retopology* secara otomatis di *ZBrush*. Metode lain yang dapat diterapkan adalah penggunaan *tool Quad Draw* di *Maya* untuk menggambar *retopology* di atas permukaan model *high-poly*. (Webster, 2017, hlm. 41-42)

Setelah hasil *retopology* model *low-poly* didapatkan, *mesh* baru tersebut akan mengalami kehilangan detail-detail halus. Karena itu, *ZBrush* menyediakan sebuah *tool* yang dapat memproyeksikan detail-detail halus pada model yang lebih *low-poly*. *Tool* tersebut adalah *ProjectAll* yang dapat ditemukan di *palette Tool / SubTool*. Proyeksi baru akan membuat *mesh* memiliki jumlah poligon rendah dengan tingkat detail yang sama. Hal ini disebabkan oleh penempatan poligon yang menjadi lebih optimal pada *mesh retopology* (Scherer, 2011).

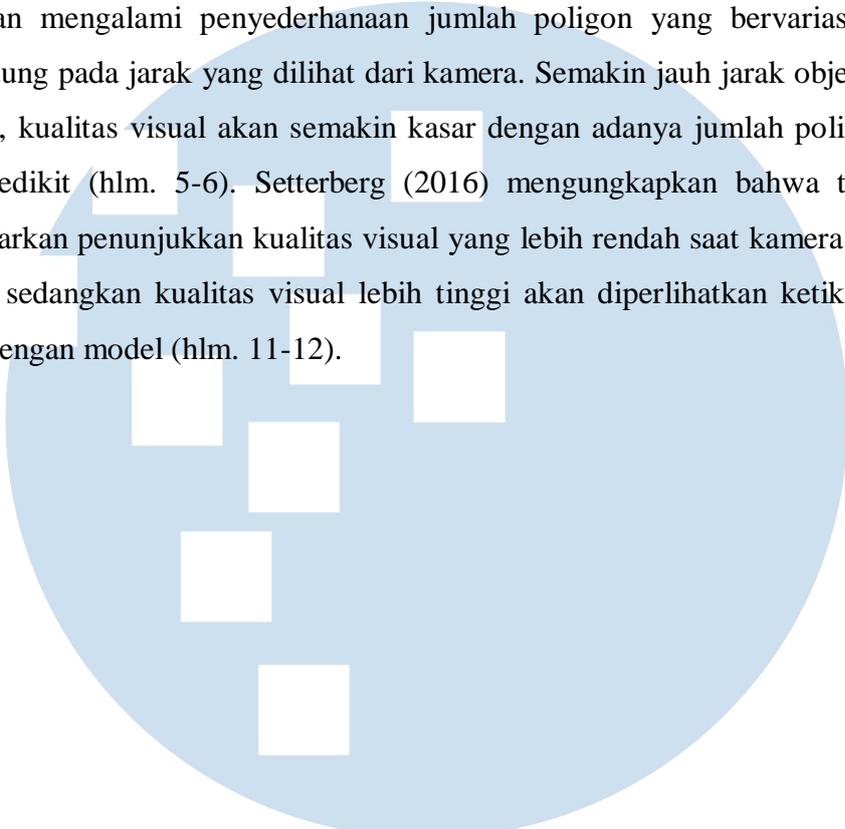
Retopology

Menurut Setterberg (2016), *retopology* adalah teknik pengurangan jumlah titik-titik *vertex* pada model, tetapi keseluruhan gambaran visual masih tetap dijaga (hlm. 22). *Retopology* merupakan penciptaan topologi baru untuk model 3D yang biasanya dilakukan dengan meletakkan *mesh low-poly* tepat di permukaan *mesh high-poly*. *Mesh* baru diciptakan dengan mengikuti bentuk visual dari model 3D. Hasil *mesh retopology* umumnya didasarkan pada tipe poligon *quads* daripada *tris* sehingga tahap-tahap selanjutnya dapat berjalan lebih mudah. Tahap-tahap yang dimaksudkan seperti tahap *rigging* atau tahap perubahan model 3D yang diam menjadi model yang mampu bergerak, berjalan, dan berlari (Barsanti & Guidi, 2017, hlm. 332).

Level of Detail (LoD)

Menurut Mahdi dkk. (2013), teknik *Level of Detail (LoD)* dapat dipakai untuk mengatur kerumitan sebuah model 3D berdasarkan jarak objek tersebut dilihat atau ketentuan lainnya. Teknik ini pertama kali diperkenalkan oleh James Clark pada tahun 1976 demi meringankan kerja proses grafis, namun kualitas visual yang sesuai tetap dapat dipertahankan. Dia mengungkapkan bahwa kualitas proses dapat menjadi lebih optimal dengan penataan detail-detail model yang

sesuai dengan jarak objek tersebut dilihat. Konsep dasar *LoD* adalah suatu model 3D akan mengalami penyederhanaan jumlah poligon yang bervariasi dengan bergantung pada jarak yang dilihat dari kamera. Semakin jauh jarak objek dengan kamera, kualitas visual akan semakin kasar dengan adanya jumlah poligon yang lebih sedikit (hlm. 5-6). Setterberg (2016) mengungkapkan bahwa teknik ini menawarkan penunjukkan kualitas visual yang lebih rendah saat kamera jauh dari model, sedangkan kualitas visual lebih tinggi akan diperlihatkan ketika kamera dekat dengan model (hlm. 11-12).

A large, light blue circular watermark logo is centered on the page. It features a stylized building facade with several rectangular windows of varying sizes and positions.

UMMN

UNIVERSITAS
MULTIMEDIA
NUSANTARA