

LOW-POLY MODELING TOKOH DAN ENVIRONMENT DALAM DESAIN GAME 3D

Yusup S. Martyastiadi¹
Raissa Theodosia²
Sera Prestasi³

Abstrak: *Game* 3D yang baik dan menarik mempunyai tingkat waktu *render* nyata (*real-time rendering*) yang tinggi. Di sisi lain, tampilan visual dengan resolusi tinggi akan menambah nilai *immersion* bagi pemain. Bagi pengembang *game* 3D, dua hal tersebut menjadi tantangan dalam mendesain *game*. Secara umum, aplikasi *game engine* membutuhkan aset *game* 3D dengan jumlah *polygon* yang rendah (*low-poly*) untuk mendukung kecepatan *real-time rendering*. Hal ini dilakukan untuk menghindari tampilan visual dan animasi yang terputus-putus di saat memainkan *game* 3D. Dalam makalah ini akan dipaparkan penerapan *low-poly modeling* yang didukung dengan metode *normal map*, *projection map* dan *texturing* agar tampilan realistiknya tetap tinggi.

Keywords: desain *game* 3D, *low-poly modeling*, *normal map*, *projection*, *real-time rendering*

Pendahuluan

Satu dekade ini, perkembangan *game* di Indonesia menunjukkan peningkatan. Munculnya beberapa perusahaan *game* asing di Indonesia merupakan indikator baik akan perkembangan industri kreatif, khususnya *game* [1]. Perkembangan *game* ini diiringi kemajuan teknologi dalam proses pengembangan dan distribusi *game* di pasaran. Saat ini pengguna *game* tidak hanya menggunakan komputer dalam memainkan *game*, tetapi mulai digantikan

pula oleh *gadget*, baik berupa *smartphone* maupun tablet.

Teknologi memang bukan satu-satunya faktor dalam keberhasilan pemasaran *game*, namun tentu saja dipengaruhi juga oleh sisi artistik visual dan cerita dalam *game* itu sendiri. Gerakan animasi yang cenderung realistik akan mendukung *immersion* bagi pemain *game*. Efek tampilan dengan resolusi tinggi memberikan kesan nyata bagi pemain *game*, seakan-akan mereka berada di dalam dunia *game* tersebut.

¹Yusup Martyastiadi adalah Staf Pengajar pada Fakultas Seni dan Desain, Universitas Multimedia Nusantara (UMN) Tangerang.

e-mail : yusup.martyastiadi@umn.

²Raissa Theodosia & ³Sera Prestasi adalah Alumnus Fakultas Seni dan Desain, Universitas Multimedia Nusantara (UMN) Tangerang.

e-mail Raissa: theodosia_raissa@yahoo.com

e-mail Sera: whitezeraz@hotmail.com

Game 3 dimensi (3D) memberikan peluang *immersion* yang kuat bagi pengguna *game*. Empati pemain *game* bisa terpengaruhi oleh tampilan yang bagus, tidak adanya kesalahan *typo* dalam layout, kecerdasan buatan yang logis, gerakan animasi yang tidak terputus-putus, dan cerita dalam *gameplay* yang menarik. Beberapa hal tersebut yang membuat pemain “terce-lup” dalam dunia *game*.

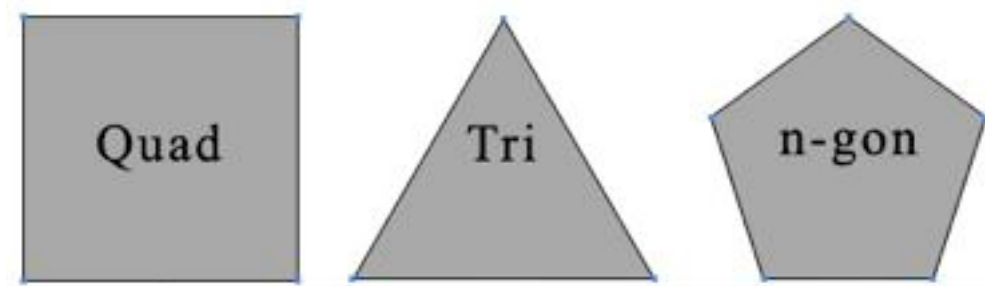
Namun perlu disadari, *game* 3D dengan jumlah *polygon* dan kualitas gambar yang tinggi akan menjadikan petaka dalam proses *rendering*. Hal ini akan menyebabkan tampilan *game* yang terputus-putus dan waktu loading yang lama. De-sainer *game* harus menyadari hal tersebut, sehingga perlu mendesain seluruh aset *game* 3D menjadi hemat waktu render. Setiap *polygon* dari sebuah obyek 3D akan dibaca dalam beberapa tris sebelum dilakukan *rendering* oleh *game engine*.

Makalah ini akan membahas metode penerapan *low-poly modeling* dalam desain *game* 3D. Metode ini sangat membantu

untuk memberikan kecepatan yang tinggi pada saat *game engine* melakukan *real-time rendering*. Sebagai pendukung *low-poly modeling*, perlu dilakukan proses *projection*, pemberian normal map pada aset 3D, dan *texturing* resolusi tinggi. Pembahasan *low-poly modeling* dibatasi hanya untuk model 3D bangunan dan karakter.

Telaah Literatur

Modeling adalah proses menciptakan geometri kompleks. Pemodelan *polygon* dapat kita lakukan untuk membuat obyek seperti apapun, bebas memanipulasi strukturnya. Hal ini bisa dilakukan dengan cara menggabungkan bagian-bagiannya, memotong, menyatukan dengan bangun lain tanpa mengganggu bentuk keseluruhan, tentu saja bila dikerjakan dengan baik [2]. Sebuah model 3D terdiri dari *polygon*, *edge*, dan *vertex*. *Polygon* adalah sebuah bentuk bersisi-*n* yang dibentuk oleh susunan *vertex* dan *edge* yang merupakan *vertex* yang berpasangan. *Polygon* biasanya memiliki empat tepi, sering juga disebut



Gambar 1. Tipe-tipe *Polygon*

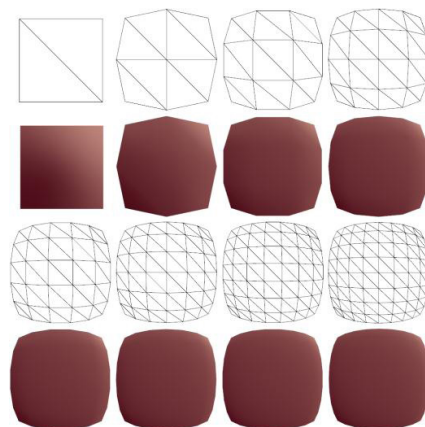
(<http://kennethvassbakk.com/3d-design/polygonal-modeling>)

sebagai *quad*. *Polygon* yang memiliki tiga simpul saling berhubungan disebut sebagai tris. Sedangkan *polygon* yang terdiri lebih dari empat edge disebut *n-gon*.

Seperti yang sudah diketahui, *game engine* mempunyai keterbatasan dalam melakukan real-time *rendering*. Oleh sebab itu, terdapat dua elemen penting dalam melakukan *modeling* 3D untuk *game* 3D, yaitu *polycount* dan topologi model 3D [3]. Dalam proses pembuatan obyek 3D, secara otomatis jumlah *polygon* akan terus bertambah. Pengurangan jumlah *polygon* akan membuat video card bekerja lebih

segitiga (tris). Pada dasarnya dalam *game engine* akan terjadi proses *Tessellation*. *Tessellation* adalah proses konversi *polygon-quads* menjadi tris dengan menyambungkan 2 vertex berseberangan pada tiap *polygon*. Proses ini dilakukan agar *game engine* dapat melihat geometri suatu model 3D dan video card dapat menampilkannya sesuai perspektifnya.

Dalam melakukan pengurangan *polygon* dan tris pada *modeling* tidak dapat dilakukan secara sembarangan karena dapat merusak struktur topologi aslinya. Topologi dalam *modeling* 3D



Gambar 2. Proses *tessellation* yang terjadi pada *game engine*
(http://www.blitzcode.net/images/projects/project_105_big.png)

ringan dan mampu melakukan proses render dengan cepat tiap detiknya. Semua aset 3D dalam *game* yang terdiri dari kumpulan *polygon* perlu dihitung oleh *game engine*, sehingga diperlukannya metode pemodelan *low-polygon* pada saat membuat aset *game* 3D. *Game engine* akan mengukur kepadatan model bukan dalam bentuk *polygon* tetapi dalam bentuk

menjadi bagian penting karena merupakan struktur permukaan suatu *polygon* dan juga sangat berpengaruh khususnya pada tekstur dengan *UVmaps*.

Sekitar tahun 2006, satu karakter 3D untuk *game* rata-rata terdiri dari 5000-7000 *polygon* [4]. Seiring berkembangnya teknologi komputer, angka tersebut meningkat. Namun perkembangan komputer

selalu memunculkan hal baru, sebagai contoh muncul visual efek berupa partikel. Sehingga ada baiknya, desainer *game* tetap memperhatikan jumlah *polygon*.

Low-poly modeling adalah proses pembuatan model 3D dengan jumlah *polygon* lebih sedikit dari *modeling* obyek yang sebenarnya. Prosesnya dimulai dengan melakukan *modeling* dengan menitikberatkan sisi artistik visual, dimana kecenderungannya akan menghasilkan jumlah *polygon* yang cukup banyak (*high-poly*). Model 3D *high-poly* tersebut akan menjadi referensi untuk melakukan rekonstruksi model 3D *high-poly* menjadi model 3D *low-poly*. Prosesnya dimulai dengan pengurangan *polygon*, kemudian *projection* untuk mendapatkan normal map, yang berfungsi untuk menimbulkan efek kedalaman pada suatu obyek yang rata. Efek ini memberikan ilusi 3D.

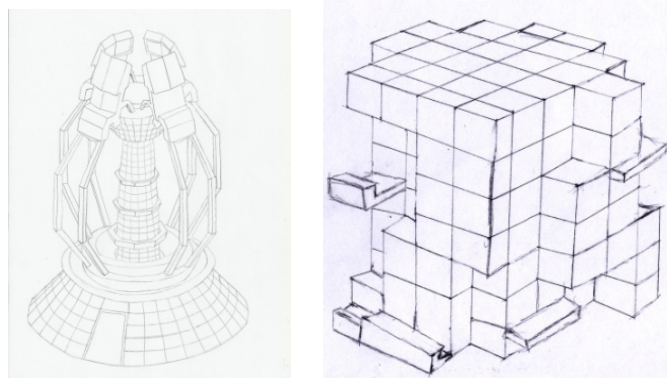
Metode dan Hasil

Dalam mendesain sebuah *game* 3D, terdapat beberapa tahap, yaitu praproduksi, produksi, dan pascaproduksi. Makalah ini akan lebih banyak menjabarkan proses produksi. Ada beberapa hal dalam tahap praproduksi yang akan tetap disinggung sebagai jembatan dalam penjelasan proses produksi. Secara umum, tahap praproduksi selalu dimulai dengan penyusunan konsep, pembuatan sketsa, dan model sheet untuk proses *modeling* 3D. Setelah itu tahap produksi dilakukan, mulai dari *modeling*, *UV-mapping*, *projection*, dan *texturing*. Pada model karakter 3D, dilakukan juga *digital sculpting* untuk menambah artistik.

Pembahasan dalam makalah ini difokuskan pada model 3D berupa bangunan (studi kasus *game 3D Emendation*) dan karakter (studi kasus *game 3D Indictus*) sebagai sampling.

A. Bangunan

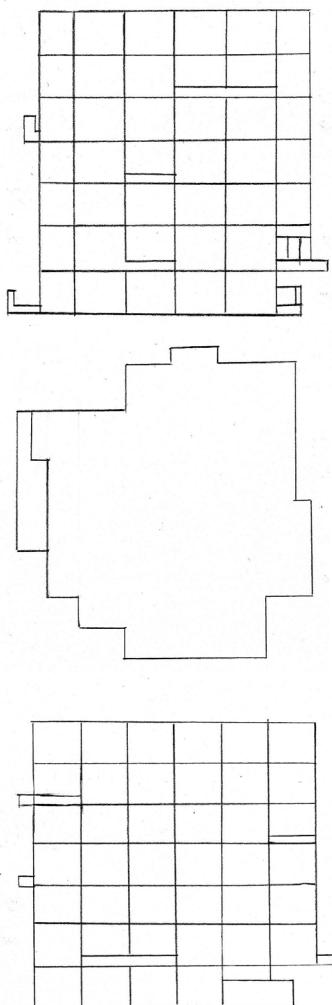
Dunia *game* yang dibangun dalam *game*



Gambar 3. Sketsa kasar bangunan dalam *game 3D Emendation*

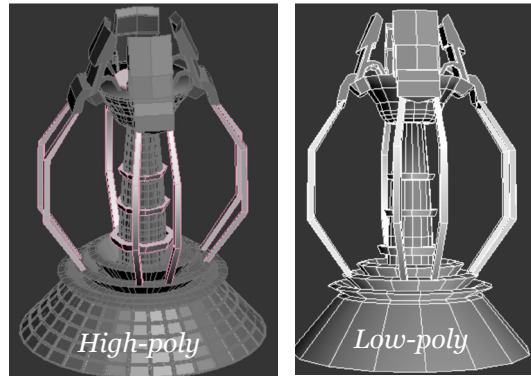
ini mempunyai konsep futuristik dengan desain bangunan yang unik dan modern. Setelah konsep bangunan disusun, maka divisualisasikan melalui sketsa.

Tal tan mo-

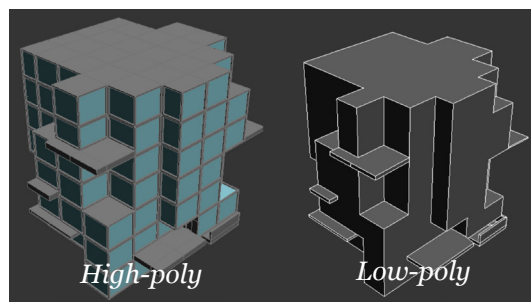


Gambar 4. Model sheet bangunan apartemen tampak kiri, atas, depan

del sheet untuk pemodelan 3D. Model sheet berfungsi untuk membantu mempermudah dalam pembuatan model obyek 3D yang lebih rumit.



Gambar 5. Bangunan Bos high-poly (14169 tris) dan low-poly (2000 tris)



Gambar 6. Bangunan apartemen high-poly (3464 tris) dan low-poly (377 tris)

Modeling dilakukan menggunakan software 3D Max 2011, dengan teknik modeling yang digunakan adalah box modeling. Pengerjaan tahap modeling dilakukan dua kali yaitu modeling high-poly dan low-poly.

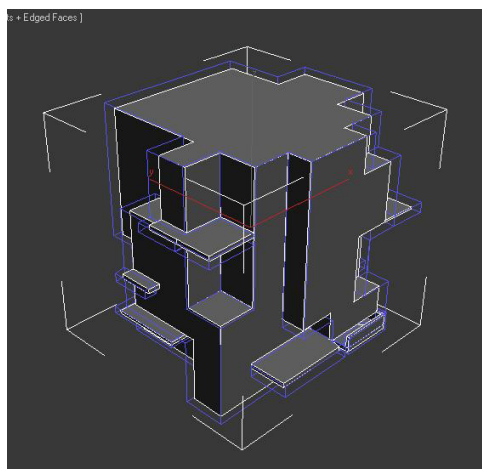
Perbandingan jumlah tris pada dua bangunan di atas cukup signifikan. Rata-rata, jumlah tris pada model 3D low-poly adalah 10% dari jumlah tris model 3D high-poly. Namun secara artistik, model 3D low-poly lebih terkesan sederhana dan tidak detil.

Dalam game pembuatan modeling high-poly berfungsi untuk menghasilkan normal map yang akan digunakan sebagai bump map. Bump map akan memberikan kesan

nyata terhadap model *low-poly*.

Setelah pemodelan 3D, dilanjutkan dengan proses *UV-Mapping*. *UV-Mapping* merupakan suatu proses penyusunan *polygon-polygon* model 3D menjadi beberapa bagian yang kemudian akan menghasilkan *UV Template* dua dimensi. Dalam proses pembuatan film animasi, tahap *UV Mapping* dilakukan pada model *high-poly*. Sedangkan dalam pembuatan *game 3D*, *UV-Mapping* dilakukan pada model *low-poly*.

Projection sangat penting dalam pengembangan *game 3D*. *Projection* mempunyai manfaat untuk menghasilkan normal



Gambar 7. Garis biru pada gambar menunjukkan area *projection* model 3D *low-poly*

map yang akan digunakan pada saat *texturing*. Sehingga model 3D *low-poly* akan terlihat seperti *high-poly*. *Projection* memiliki kontrol untuk memproyeksikan data dari obyek ke obyek yang berbeda dengan *modifier projection*.

B. Karakter

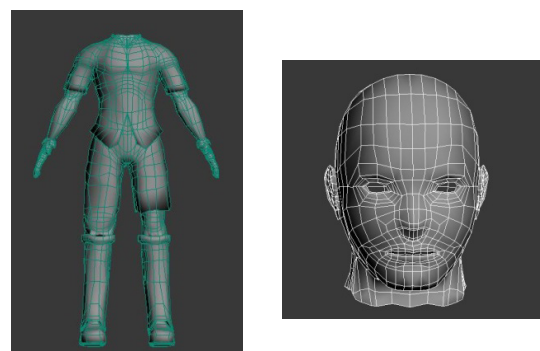
Desain karakter *game 3D Indictus* dibangun dengan referensi masa *Steampunk* dengan



Gambar 8. Sketsa tokoh utama dan Bos musuh [5].

pakaian kemeja lengan panjang, *vest* (rompi), celana panjang, *gaiters* (pelindung betis yang biasa terbuat dari kain, kulit, atau poliester). Tokoh utama dalam *game Indictus* mempunyai musuh utama yang biasa disebut Bos.

Setelah konsep dan sketsa diselesaikan



Gambar 9. Proses *modeling* menggunakan teknik *edge* dan *box modeling*

maka proses *modeling* dilakukan. Tentu saja *modeling* akan mudah dilakukan dengan bantuan model sheet. Karakter 3D disebut juga model 3D organik. Proses

Tabel 1. Ujicoba *real-time rendering*

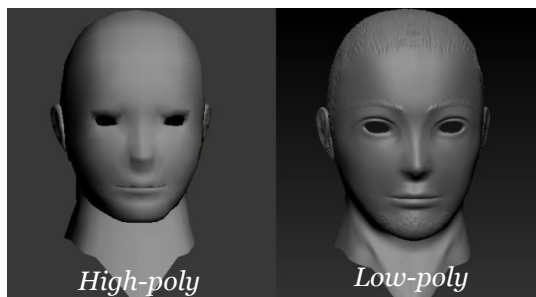
Aset 3D	High-poly			Low-poly		
	Tris (ribuan)	Rendering (fps)	Draw Call	Tris (ribuan)	Rendering (fps)	Draw Call
Apartemen	3,5	1098,3 (0,9 ms/f)	6	0,377	1196,0 (0,8 ms/f)	1
Gedung Bos	18	1039,2 (1,0 ms/f)	6	2,2	1230,9 (0,8 ms/f)	3
Tokoh utama	1600	109,6 (9,1 ms/f)	36	6,6	152,7 (6,5 ms/f)	8
Boss	1800	137,2 (7,1 ms/f)	63	6,1	391,6 (2,6 ms/f)	7
Monster kura-kura	1000	71,2 (14 ms/f)	45	1,8	129 (7,7 ms/f)	7

pemodelannya biasanya dilakukan dengan teknik *edge modeling* dan *box modeling*.

Seperti yang dinyatakan oleh Franson

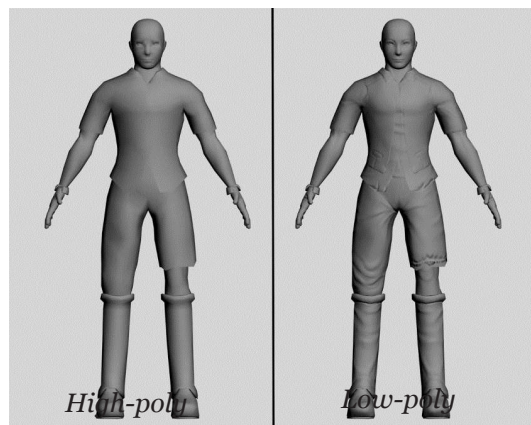
pada karakter.

Penambahan efek kedalaman pada model karakter 3D *low-poly* dapat dilakukan de-



Gambar 10. Perbandingan kepala *low-poly* dan *high-poly* (setelah melalui proses *digital sculpting*)

dan Thomas [4], bahwa untuk karakter *game* 3D, jumlah *polygon* dibawah 8000 sudah bisa dikatakan model 3D *low-poly*. *Digital sculpting* bisa dilakukan juga untuk memberikan kesan lebih nyata dan detail

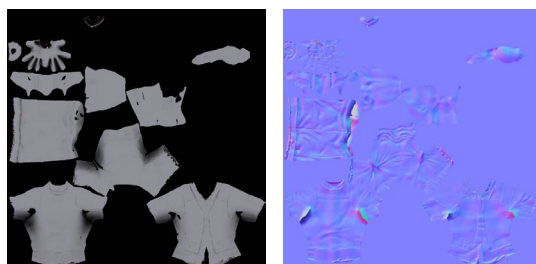


Gambar 12. Perbandingan model *low-poly* tanpa dan dengan *normal bump*

ngan proses “render to texture” pada model *high-poly*. Proses ini bisa menghasilkan *complete map* dan *normal map*.

Ilusi kedalaman 3D yang ditimbulkan cukup signifikan pada model karakter 3D yang telah melalui *baking render to texture*.

Tabel 1. menunjukkan perbandingan kecepatan render *high-poly* dan *low-poly* untuk masing-masing aset 3D. Eksperimen *real-time rendering* dilakukan di *game engine* Unity. Percobaan tersebut menggunakan kamera statis, sehingga



Gambar 11. *Complete map* dan *normal map* dari proses *baking render to texture*

pencuplikan hasil render diambil tanpa ada pergerakan dari kamera.

Tentu saja, hasil *rendering* dengan kamera yang bergerak akan berbeda ketika dilakukan di dunia *game* yang lebih lengkap, dengan seluruh aset *game* 3D sudah terpasang sesuai *game* play



Gambar 13. Tokoh utama dalam *game* 3D *Indictus* setelah diberikan tekstur

yang didesain. Maka hasil *rendering* akan lebih dinamis sesuai dengan jumlah obyek yang berada di depan kamera. Selain itu, visualisasi aset 3D (*draw call batching*) dalam *game* dipengaruhi juga oleh spesifikasi *video card* pada komputer yang digunakan.

Kesimpulan

Penerapan *low-poly modeling* dalam

pembangunan *game* 3D membantu mempercepat proses *rendering* dalam *game engine*. Semakin sedikit jumlah tris yang dibaca oleh *game engine*, maka proses real-time *rendering* yang terjadi semakin cepat, sehingga *game* dengan tampilan yang terkesan realistis akan tetap bisa dimainkan tanpa terputus-putus.

Referensi

Asih, R. (10 November 2012). *Invasi Industri Game Asing*. Tempo. Diakses dari [http:// www.tempo.co/read/news/2012/11/10/172440881/](http://www.tempo.co/read/news/2012/11/10/172440881/)

Russo, M. (2006). *Polygonal Modeling Basic and Advance Technique*. USA: Wordware Publishing.

Watkins, A. (2011). *Creating games with Unity and Maya*. Burlington: Elsevier.

Franson, D., & Thomas, E. (2007). *Game Character Design Complete*. Boston: Thomson Course Technology.

Theodosia, R. (2013). *Desain Karakter Dalam Game "Indictus"*. Laporan Tugas Akhir. Universitas Multimedia Nusantara. Tangerang.

Prestasi, S. (2013). *Pemodelan Environment 3d Dan Penerapannya Dalam Game "Emendation"*. Laporan Tugas Akhir. Universitas Multimedia Nusantara. Tangerang.