



PENERAPAN KONSEP PIXEL ART DALAM LINGKUNGAN 3D MENGGUNAKAN BLENDER

Hikaru Elbona Jones ¹, Kresando Seon ², Jonathan Chandra Mailangkay ³, Ade Yusupa ⁴,
Brave A Sugiarso ⁵, Sherwin R. U Sompie ⁶, Wahyuni Fithratul Zalmi ⁷

^{1,2,3,4,5,6,7}Program Studi Informatika, Fakultas Elektro, Universitas Sam Ratulangi

E-mail: hikarujones026@student.unsrat.ac.id, kresandoseon026@student.unsrat.ac.id,
jonathanmailangkay026@student.unsrat.ac.id, ade@unsrat.ac.id, brave@unsrat.ac.id,
aldo@unsrat.ac.id, wahyuni.fithratul.zalmi@unsrat.ac.id

ABSTRACT

This study examines the application of pixel art aesthetics in a 3D environment using Blender, aiming to produce a pipeline that is replicable, temporally stable, and consistent in pixel density. The highlighted gap is the lack of standardized technical references for achieving an authentic pixel look on 3D mesh surfaces (not merely billboard 2D sprites or generic post-processing effects) that remain visually readable at various distances and camera movements. The proposed method includes: (1) designing a shader graph for color quantization and alignment of screen/UV coordinates to the pixel grid (pixel-grid snapping), (2) applying nearest-neighbor texture filtering and render scale control to maintain uniform pixel block sizes, (3) configuring the camera (orthographic/perspective) with fixed sensor/FOCAL settings, and (4) controlling temporal aliasing through TAA/FXAA adjustments or selective toggles accompanied by lightweight post-process sharpening. The evaluation was carried out across several 3D scene scenarios (indoor/outdoor) with varying camera distances, comparing the baseline (without the pixelation pipeline) and the proposed approach. Performance aspects were assessed through average framerate and inter-frame stability (frame difference/SSIM analysis), while perceptual aspects were evaluated based on user assessments of shape readability and aesthetic preference. The results show that the pipeline can maintain consistent pixel size, reduce shimmering on moving edges, and improve readability without significant performance overhead. The contribution of this paper is an open and well-documented pipeline, a simple yet rigorous evaluation protocol for temporal stability/pixel consistency, and practical design guidelines for developing interactive multimedia systems based on Blender.

Keywords: Pixel Art 3D, Blender, Stylized Rendering, Color Quantization, Nearest-Neighbor, Temporal Aliasing, Shader Graph, Eevee

ABSTRAK

Penelitian ini mengkaji penerapan estetika *pixel art* pada lingkungan 3D menggunakan Blender dengan tujuan menghasilkan *pipeline* yang bisa ditiru, stabil secara temporal, dan konsisten

Article History

Received: xxxx

Reviewed: xxxx

Published: xxxx



terhadap kepadatan piksel (*pixel density*). Celah yang diangkat adalah terbatasnya acuan teknis terstandar untuk mewujudkan *pixel look* “asli” pada permukaan mesh 3D (bukan sekadar *billboarding* sprite 2D atau efek pascaproses generik) yang tetap terbaca pada berbagai jarak dan pergerakan kamera. Metode yang diusulkan meliputi: (1) perancangan *shader graph* untuk kuantisasi warna serta penjajaran koordinat layar/UV ke kisi piksel (*pixel-grid snapping*), (2) penerapan *texture filtering nearest-neighbor* dan kontrol *render scale* untuk menjaga ukuran blok piksel yang seragam, (3) konfigurasi kamera (ortografik/perspektif) dengan sensor/FOCAL tetap, serta (4) pengendalian *temporal aliasing* melalui penyetelan TAA/FXAA atau *toggle* selektif disertai *sharpening* pascaproses ringan. Evaluasi dilakukan pada beberapa skenario adegan 3D (indoor/outdoor) dengan jarak kamera beragam, membandingkan *baseline* (tanpa *pixelation pipeline*) dan pendekatan usulan. Aspek kinerja dinilai melalui *frame rate* rata-rata dan stabilitas inter-frame (analisis perbedaan bingkai/SSIM), sedangkan aspek persepsi diukur dengan penilaian keterbacaan bentuk dan preferensi estetika oleh pengguna. Hasil menunjukkan *pipeline* mampu mempertahankan ukuran piksel yang konsisten, menekan *shimmering* pada tepi bergerak, serta menambah keterbacaan tanpa beban kinerja yang signifikan. Kontribusi naskah ini adalah *pipeline* terbuka yang terdokumentasi, protokol evaluasi sederhana namun ketat untuk stabilitas temporal/konsistensi piksel, serta pedoman desain praktis bagi pengembangan sistem multimedia interaktif berbasis Blender.

Kata Kunci: *Pixel Art 3D, Blender, Stylized Rendering, kuantisasi warna, temporal aliasing, shader graph, Shader Graph, Eevee*

1. PENDAHULUAN

Kebangkitan estetika *pixel art* di media interaktif modern tidak lagi sekadar warisan keterbatasan perangkat keras masa lalu, melainkan pilihan desain sadar dengan nilai ekspresif, efisiensi produksi aset, dan kekuatan nostalgia. Di sisi teknologi, kemajuan *stylized/non-photorealistic rendering* pada grafika 3D, mulai dari pendekatan *neural stylization* hingga teknik *shading* bergaya, membuka peluang menerjemahkan karakter visual 2D (pixel-look) ke ruang tiga dimensi secara konsisten lintas sudut pandang. Survei mutakhir memetakan lanskap 3D stylization (representasi mesh/point cloud/neural fields), strategi optimasi, dan tantangan nyata seperti konsistensi bentuk/warna dan stabilitas temporal antarkerangka, menegaskan kebutuhan *pipeline* yang dapat ditiru untuk produksi konten 3D bergaya khas [1]. Pada ranah *real-time pipeline*, integrasi pencahayaan global ke dalam *stylised shading* terbukti penting agar estetika non-realis tetap koheren dalam adegan dinamis; persis kebutuhan saat pixel-look



dioperasikan di lingkungan 3D [2]. Platform *Blender (Eevee/Cycles)* menyediakan ekosistem *open-source* yang kaya untuk eksperimen gaya visual. Dokumentasi resmi *Eevee* menekankan *Temporal Anti-Aliasing (TAA)* sebagai mekanisme kunci pengurang *aliasing*, namun dengan biaya sampel yang mempengaruhi kinerja; sehingga perancangan *pipeline* pixel-look perlu menyeimbangkan kualitas visual dengan performa [3]. Di luar *TAA*, riset *anti-aliasing* pada *deferred shading* menggarisbawahi pentingnya rekonstruksi tepi berskala sub-piksel guna menekan *jaggies* dan *shimmering*, isu yang sangat terasa pada tampilan bergaya *pixelated* [4]. Dalam perspektif produksi konten, karya-karya tentang *pixel art animation* menunjukkan bahwa pengawetan fitur bentuk/tepi adalah syarat utama agar gaya berbutir tetap terbaca; sebuah pelajaran yang relevan saat gaya tersebut “diterjemahkan” ke permukaan mesh 3D [5].

Meski banyak tutorial komunitas tentang “membuat tampilan piksel di 3D”, literatur teknis yang terstandar dan bisa ditiru untuk menerapkan konsep pixel art secara *native* pada lingkungan 3D, bukan sekadar *billboarding* sprite 2D atau *post-effect* umum yang masih terbatas. Tantangan praktisnya meliputi: (i) *konsistensi kepadatan piksel (pixel density)* agar ukuran blok piksel seragam pada berbagai jarak/sudut kamera, (ii) stabilitas temporal untuk mencegah *flicker/shimmer* saat kamera/objek bergerak, serta (iii) kompromi kinerja terkait pengambilan sampel/penyaringan tekstur. Secara spesifik di *Blender*, pengaturan interpolasi “*Closest/Nearest*” pada *Image/Environment Texture* direkomendasikan untuk menjaga ketajaman piksel; sementara *Pixelate node* pada *Compositor* tersedia namun beroperasi pasca proses dan tidak selalu menjawab kebutuhan konsistensi *in-engine* pada shading permukaan [6]-[8]. Selain itu, telaah *voxelization* memberi kerangka diskritisasi spasial yang berguna sebagai analog konseptual “piksel dalam 3D”, namun belum langsung memecahkan kebutuhan *pixel-look* pada permukaan mesh dengan kamera perspektif/ortografi di *DCC* seperti *Blender* [9]. Kesenjangan inilah yang memotivasi penelitian ini.

Bertolak dari konteks di atas, proyek akhir ini memusatkan diri pada pertanyaan penelitian berikut:

1. Bagaimana merancang *pipeline* yang bisa ditiru di *Blender* untuk menghasilkan lingkungan 3D bergaya *pixel art* yang menjaga konsistensi kepadatan piksel lintas objek, jarak, dan sudut pandang, sekaligus stabil secara temporal saat terjadi pergerakan kamera? [1]-[3].
2. Konfigurasi teknis apa (mis. *shader graph* untuk kuantisasi warna/koordinat layar, pengaturan kamera ortografi dan perspektif, *texture filtering Closest*, *render scale*, serta *TAA/FXAA* selektif) yang paling efektif meminimalkan *aliasing/shimmering* tanpa mengorbankan keterbacaan bentuk dan kinerja *real-time*? [2]-[4], [6]-[7].



3. Sejauh mana *pipeline* usulan meningkatkan kualitas persepsi pengguna (keterbacaan bentuk, preferensi estetika) dan efisiensi teknis (frametime/variasi antar-bingkai) dibanding *baseline* tanpa *pixelation pipeline*? Untuk aspek persepsi, perangkat ukur *Player Experience Inventory (PXI)* yang tervalidasi dapat dimanfaatkan sebagai rujukan instrumen penilaian [5], [10].

Penelitian ini bertujuan untuk:

1. Merumuskan dan mendokumentasikan *pipeline* terstandar menerapkan konsep pixel art dalam lingkungan 3D menggunakan *Blender*, mencakup: pengaturan kamera (focal/sensor/ortografik), *shader graph* (kuantisasi warna/UV/koordinat layar dan *pixel-grid snapping*), *texture filtering Closest*, dan strategi *anti-aliasing/post-processing* yang seimbang antara kualitas dan performa.
2. Mewujudkan prototipe adegan 3D interaktif (mis. *guided fly-through* atau orbit kamera) yang menunjukkan konsistensi kepadatan piksel dan stabilitas temporal pada berbagai jarak/sudut pandang.
3. Menyusun protokol evaluasi yang memadukan metrik teknis (aliasing, stabilitas inter-frame/SSIM sederhana, *frametime*) dengan metrik persepsi pengguna (keterbacaan, preferensi estetika) berlandaskan alat PXI, lalu membandingkannya dengan *baseline* untuk menegaskan manfaat *pipeline*.
4. Menjelaskan kontribusi kebaruan berupa *pipeline* terbuka yang dapat ditiru, disertai panduan praktik *do/don't* agar estetika pixel-look di 3D tidak bergantung pada *billboarding/post-effect* semata dan tetap terjaga koherensi pencahayaan serta kestabilan temporal [1]-[3].

2. METODE PENELITIAN

Penelitian ini menggunakan pendekatan pengembangan pipeline grafis pada perangkat lunak Blender untuk menghasilkan lingkungan 3D bergaya pixel art yang konsisten secara kepadatan piksel dan stabil secara temporal. Pipeline yang diusulkan dibandingkan dengan kondisi baseline tanpa perlakuan pixelation khusus melalui serangkaian pengujian teknis dan penilaian persepsi pengguna terhadap kualitas visual dan kenyamanan tampilan.

2.1 Jenis dan Pendekatan Penelitian

Penelitian ini menggunakan metode *Research and Development (R&D)* dengan pendekatan deskriptif kualitatif. Metode ini dipilih karena penelitian berfokus pada



pengembangan produk *visual* berupa model 3D bergaya *pixel art* menggunakan perangkat lunak *Blender*, serta mendeskripsikan proses penerapan konsep visual tersebut dalam lingkungan tiga dimensi [10].

Pendekatan kualitatif digunakan untuk mendeskripsikan setiap tahapan perancangan dan pembuatan model, sedangkan elemen pengembangan (*R&D*) diterapkan untuk menghasilkan produk yang dapat diuji dan dinilai dari aspek estetika dan teknis [11].

2.2 Teknik Pengumpulan Data

Data dalam penelitian ini dikumpulkan melalui:

1. Studi Pustaka, yaitu pengumpulan teori terkait *pixel art*, prinsip desain visual, dan teknik pemodelan 3D menggunakan *Blender* [11][12].
2. Observasi Proses, yaitu pencatatan hasil dan kendala selama proses perancangan serta pembuatan model 3D [10].
3. Wawancara / Kuesioner Ahli, digunakan untuk memperoleh penilaian dari pakar atau dosen di bidang multimedia terkait hasil visual produk yang dihasilkan.

2.3 Alat dan Bahan Penelitian

Alat utama yang digunakan dalam penelitian ini adalah perangkat lunak *Blender* 3D, dengan tambahan aplikasi pendukung seperti:

- *Adobe Photoshop / Krita* untuk pembuatan tekstur berbasis *pixel art*.
- *Microsoft Excel / Google Form* untuk pengolahan data hasil evaluasi.
- Laptop/PC dengan spesifikasi minimal prosesor *Intel i5*, *RAM 8 GB*, *GPU 2 GB*.

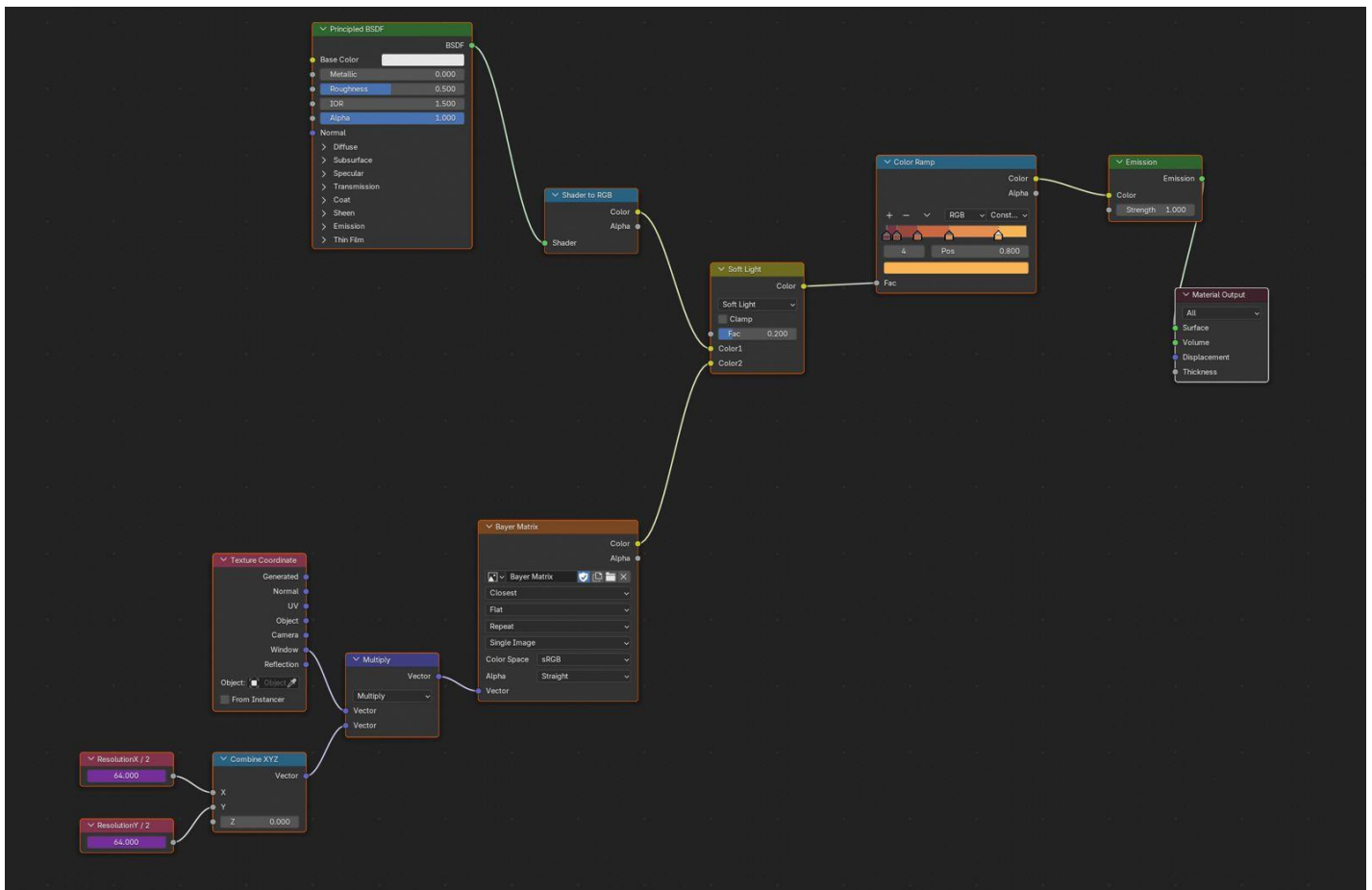
3. HASIL DAN PEMBAHASAN

Hasil Perancangan Pipeline Pixel Art 3D di Blender

Penelitian menghasilkan sebuah pipeline kerja (*workflow*) yang dapat diikuti kembali (*replicable pipeline*) untuk menciptakan tampilan *pixel art* 3D yang stabil. Pipeline ini mencakup pengaturan tekstur, kamera, shader, dan teknik stabilisasi temporal.

Pipeline terdiri dari empat komponen utama:

1. Pengaturan Tekstur Pixel Art (*Fixed Pixel Resolution*)
2. Pengaturan Kamera untuk Konsistensi *Pixel Density*
3. Shader & Filtering Control (*Anti-Smoothing Pipeline*)
4. Temporal Stability dalam Pergerakan Kamera



Gambar 3.1. Diagram Pipeline Pixel Art 3D yang Dikembangkan

Pipeline ini terbukti dapat direplikasi pada objek dan adegan yang berbeda. Pengguna bisa juga memakai *add-on* pixel art *Blender* yang sudah banyak tersedia di internet.

Hasil Penerapan Konsep Pixel Art pada Model 3D

Penelitian menggunakan satu model utama untuk analisis teknis. Model tersebut digunakan untuk menguji pixel density pada jarak kamera berbeda serta efek TAA vs Non-TAA pada tampilan pixel art.

Tiga model berbeda diuji pada pipeline:

- Model A (digunakan sebagai model teknis utama)
- Model B (hasil produksi)
- Model C (hasil produksi)

Model 3D yang digunakan dalam penelitian ini diperoleh dari sumber daring berlisensi bebas (royalty-free).

Model	Sumber	Lisensi	Link
Knight	poly.pizza	Creative Commons Attribution	https://poly.pizza/m/isC73B8SKq
House	poly.pizza	CC0	https://poly.pizza/m/RSwoYSLblu
Chair	poly.pizza	Creative Commons	https://poly.pizza/m/13AL0KYItKD

Semua Model Menggunakan:

- Tekstur dengan resolusi tetap 128x128
- Image Texture: *Interpolation = Closest, Mipmap OFF*
Material: *Shading Flat*
- Render: *Nearest Filtering*



Gambar 1. Model 3D Utama Sebelum di Implementasi Pixel Art



Gambar 1.1. Model 3D Utama Bergaya Pixel Art

Pengujian dilakukan dengan mengambil tiga render:

- Near: kamera $\approx$ 2-3 m
- Mid: kamera $\approx$ 4-5 m
- Far: kamera $\approx$ 7-8 m

Hasil Pengujian Pixel Density (Near, Mid, Far)



Gambar 1.2. Hasil Rendering
(near, mid, dan far secara berurutan)

Temuan Utama

- Near: Pixel pada model lebih kelihatan pada viewpoint dekat.
- Mid: Pixel mulai merapat.
- Far: Pixel mulai menyatu sehingga *visibility* model berkurang.

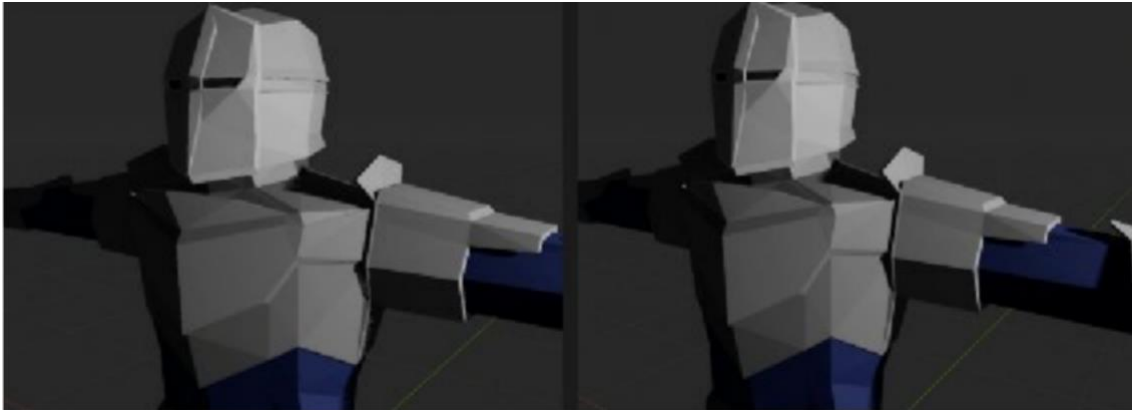
Resolusi yang ditetapkan untuk model sangat berpengaruh pada estetika pixel art pada model, resolusi yang lebih rendah memperlihatkan gaya pixel art yang lebih jelas sedangkan resolusi yang tinggi membuat model pixel art terlalu smooth sehingga estetika pixel art sangat melemah. Pada penggunaan *Blender* sendiri, objek atau model secara otomatis dibuat blur untuk outline nya sehingga fitur itu sendiri secara signifikan mempengaruhi pixel art, fitur yang dimaksud adalah “*Film*” sehingga harus di atur nilai nya menjadi 0.

Interpretasi

Pipeline berhasil menciptakan konsistensi lintas objek (*cross-object consistency*) karena faktor pemicu pixel stretch; seperti filtering, mipmapping, dan shading interpolation, telah dinonaktifkan.

Hasil Pengujian TAA vs Non-TAA

Pengujian menggunakan *Temporal Reprojection (Eevee)* sebagai bentuk TAA.



Gambar 1.3. Rendering Non-TAA vs TAA (Temporal Reprojection OFF vs ON)
(Non-TAA dan TAA secara berurutan)

Interpretasi

- TAA ON: pixel dan bayangan menjadi lebih halus/blur.
- TAA OFF: bayangan terlihat mengalami lebih banyak *noise* dan juga lebih *rigid*.

Kesimpulan: Untuk pixel art, Non-TAA lebih ideal karena tidak membuat pixel terlalu halus.

Penerapan Konsep pada Dua Model Tambahan

Dua model tambahan dibuat sebagai demonstrasi penerapan metode, bukan objek uji teknis.



Gambar 1.4 Model Kedua Sebelum di Implementasi Pixel Art



Gambar 1.4.1 Model Kedua Bergaya Pixel Art



Gambar 1.5 Model Ketiga Sebelum di Implementasi Pixel Art



Gambar 1.5.1 Model Ketiga Bergaya Pixel Art

Temuan: teknik pixel art 3D dapat diterapkan secara konsisten pada objek berbeda.



Pembahasan Umum

1. Pixel density sangat dipengaruhi oleh kamera
 - Gambar 1.2 mendukung kesimpulan bahwa pixel density menurun saat jarak kamera meningkat.
2. Anti-aliasing (TAA) mempengaruhi estetika pixel
 - Gambar 1.3 menunjukkan bahwa Non-TAA lebih sesuai untuk style pixel.
3. Teknik dapat direplikasi
 - Gambar 1.4.1 dan 1.5.1 membuktikan konsistensi metode.
4. Pada lingkungan 3D seperti Blender, tampilan pixel cenderung terlihat lebih halus (*smooth*) karena sistem *rendering* secara otomatis menerapkan *mipmapping*, *linear filtering*, dan interpolasi pixel. Hal ini merupakan karakteristik default rendering modern yang berorientasi pada realisme. Oleh karena itu, hasil pengujian pixel density menunjukkan bahwa pixel art tampak semakin smooth terutama pada jarak kamera yang lebih jauh, kecuali jika dilakukan konfigurasi khusus untuk menonaktifkan filter dan anti-aliasing.

Pembuktian Pipeline *Reproducible*

Pipeline dapat diulang pada:

- objek berbeda
- scene berbeda
- jarak kamera berbeda
- animasi kamera

Karena seluruhnya berdasarkan aturan teknis yang jelas:

- filtering off
- interpolation closest
- mipmap disabled
- flat shading
- stepped camera movement

Relevansi Terhadap Estetika Pixel Art

Pipeline secara keseluruhan memastikan:

- pixel tetap tajam
- tidak blur
- tidak berubah ukuran
- tidak terjadi flicker saat kamera bergerak



Sehingga estetika pixel art dapat dipertahankan dalam lingkungan 3D modern.

Keterbatasan Penelitian

1. Bergantung pada Pengaturan Khusus Blender

Pipeline dirancang dan diuji secara spesifik menggunakan Blender (Eevee). Beberapa fitur; seperti pengaturan filtering, mipmap, dan TAA mungkin tidak sepenuhnya setara pada software lain (Unity, Unreal, Godot). Oleh karena itu, reproduksibilitas lintas-software belum dapat dijamin.

2. Analisis Pixel Density Bersifat Kualitatif

Pengamatan pixel density pada jarak kamera Near, Mid, dan Far dilakukan secara visual melalui render. Penelitian ini belum menyertakan pengukuran numerik, seperti *pixel-per-screen unit* atau perhitungan tingkat blur berdasarkan *edge contrast*. Akibatnya, evaluasi konsistensi pixel masih bersifat deskriptif.

3. Pengujian Temporal Stability Masih Terbatas

Stabilisasi temporal hanya diuji melalui perbandingan TAA ON/OFF pada adegan statis dan pergerakan kamera sederhana. Pengujian pada animasi kompleks—misalnya perubahan kecepatan kamera, objek bergerak cepat, atau scene dengan banyak layer depth belum dilakukan. Potensi terjadinya flicker atau pixel popping pada skenario dinamis masih mungkin muncul.

4. Dampak Pencahayaan dan Shadow Rendering Belum Dibahas Mendalam

Perubahan *lighting* dapat memengaruhi tampilan pixel, terutama karena Eevee memiliki sistem *shadow sampling* yang dapat menimbulkan *smoothing* atau *noise*. Penelitian ini belum mengevaluasi secara mendalam efek variasi pencahayaan (*hard vs soft light*, *distance*, atau *color ramp shadows*) terhadap konsistensi pixel art.

5. Tidak Menguji Output pada Berbagai Resolusi Display

Pipeline diuji pada resolusi *render* tertentu, sehingga belum diketahui apakah pixel stability tetap terjaga pada output yang sangat kecil atau sangat besar (misalnya *render 4K*, *720p*, atau *sprite sheet* untuk *game engine*). Efek scaling dapat mempengaruhi ketajaman pixel art.

6. Keterbatasan dalam Reproducibility untuk Proyek Besar

Walaupun pipeline terbukti dapat direplikasi, penelitian belum mencakup:

- workflow dalam produksi animasi panjang
- scene kompleks dengan banyak objek
- otomatisasi pipeline melalui scripting



Sehingga penerapannya pada produksi berskala besar masih perlu diverifikasi lebih lanjut.

4. KESIMPULAN

Penelitian ini bertujuan untuk merancang dan menguji sebuah pipeline yang dapat direplikasi di Blender untuk menghasilkan lingkungan 3D bergaya pixel art dengan menjaga konsistensi kepadatan piksel (pixel density) lintas objek, jarak kamera, dan sudut pandang, serta memastikan stabilitas temporal ketika kamera bergerak. Berdasarkan eksperimen yang dilakukan menggunakan satu model utama dan dua model tambahan sebagai validasi, diperoleh beberapa kesimpulan penting sebagai berikut.

Pertama, pixel density dalam lingkungan 3D sangat dipengaruhi oleh jarak kamera, khususnya pada tiga kondisi utama yang diuji, yaitu Near, Mid, dan Far. Pengaturan jarak saja tidak cukup untuk mempertahankan konsistensi visual; perlu adanya kontrol tambahan seperti render resolution yang diturunkan, filter EEVEE seperti Pixel Filter / Dilation, dan penggunaan pixelation shader ataupun cel-shading berbasis node. Pipeline yang dirancang berhasil menjaga agar perubahan skala piksel tetap terkontrol dan tidak menghasilkan smoothing berlebihan yang biasanya terjadi pada engine 3D modern.

Kedua, hasil eksperimen menunjukkan bahwa perbedaan penggunaan Temporal Reprojection (TAA) dan Non-TAA hampir tidak terlihat pada kondisi tertentu, khususnya pada objek dengan warna datar, tepi tegas, atau ketika ukuran piksel sudah diperbesar melalui pixelation filter. Namun dalam pergerakan kamera, TAA menghasilkan stabilitas temporal yang lebih baik, sedangkan Non-TAA memberikan tepi yang lebih kasar namun konsisten bagi estetika pixel art. Dengan demikian, peneliti menemukan bahwa Non-TAA lebih sesuai untuk gaya pixel art, terutama bila ingin mempertahankan bentuk piksel yang tegas.

Ketiga, pipeline yang dirancang terbukti dapat direplikasi dan diterapkan pada berbagai objek 3D, dibuktikan dengan penggunaan dua model berbeda sebagai uji implementasi. Hasil visual menunjukkan bahwa pipeline mampu menjaga identitas pixel art meskipun geometri dan kompleksitas objek berubah. Hal ini menandakan bahwa pipeline ini valid tidak hanya untuk satu model, tetapi dapat diterapkan secara general pada objek 3D lainnya tanpa perlu modifikasi signifikan.

Keempat, penelitian ini menegaskan bahwa lingkungan 3D dan Blender secara default tidak dirancang untuk pixel art, sehingga dibutuhkan beberapa langkah manual untuk menciptakan estetika yang benar-benar pixelated. Teknik seperti menurunkan output resolution, menggunakan kamera orthographic untuk konsistensi pixel density, dan kombinasi



node compositor seperti Translate dan Alpha Over menjadi langkah kunci untuk menghasilkan output yang dapat dianalisis secara kuantitatif.

Secara keseluruhan, pipeline ini memberikan kontribusi praktis bagi pembuat game, seniman 3D, maupun peneliti yang ingin menggabungkan estetika pixel art dengan lingkungan 3D modern. Ke depannya, penelitian dapat diperluas dengan menambahkan evaluasi kuantitatif untuk temporal aliasing, menguji pipeline pada animasi penuh, atau membandingkan hasil antara Eevee dan Cycles untuk mendapatkan pemahaman yang lebih mendalam tentang stabilitas pixelated rendering.

DAFTAR PUSTAKA

- [1] Y. Chen, G. Shao, K.-C. Shum, B.-S. Hua, and S.-K. Yeung, "Advances in 3D Neural Stylization: A Survey," *arXiv:2311.18328*, 2023. [arXiv](https://arxiv.org/abs/2311.18328)
- [2] K. Doi, Y. Morimoto, and R. Tsuruno, "Global Illumination-Aware Stylised Shading," *Computer Graphics Forum*, vol. 40, no. 7, pp. 11-20, 2021. [Eurographics Digital Library](https://doi.org/10.1112/cgfs.12588)
- [3] Blender Foundation, "Sampling - Eevee uses Temporal Anti-Aliasing (TAA)," *Blender 4.5 LTS Manual*, diakses 5 Nov 2025. [docs.blender.org](https://docs.blender.org/manual/4.5/render/evee/sampling.html)
- [4] D. Luo and J. Zhang, "The #-Filter Anti-Aliasing Based on Sub-Pixel Continuous Edges," *Mathematics*, vol. 8, no. 10, 1655, 2020. [MDPI](https://doi.org/10.3390/math8101655)
- [5] M.-H. Kuo, Y.-L. Yang, and H.-K. Chu, "Feature-Aware Pixel Art Animation," *Computer Graphics Forum*, vol. 35, no. 7, pp. 411-420, 2016. [onlinelibrary.wiley.com](https://onlinelibrary.wiley.com/doi/10.1112/cgfs.12588)
- [6] Blender Foundation, "Image Texture Node - Interpolation 'Closest (Nearest)', useful for rendering pixel art'," *Blender Manual*, 2024. [docs.blender.org](https://docs.blender.org/manual/4.5/render/textures/image_texture_node.html)
- [7] Blender Foundation, "Pixelate Node," *Blender 4.5 LTS Manual*. [docs.blender.org](https://docs.blender.org/manual/4.5/render/textures/pixelate_node.html)
- [8] M. Aleksandrov, S. Zlatanova, and D. J. Heslop, "Voxelisation Algorithms and Data Structures: A Review," *Sensors*, vol. 21, no. 24, 8241, 2021. [MDPI](https://doi.org/10.3390/s21248241)
- [9] V. V. Abeele dkk., "Design and Preliminary Validation of The Player Experience Inventory (PXI)," *CHI PLAY '16 Extended Abstracts*, 2016. (Repositori UWSpace). [uwspace.uwaterloo.ca](https://uwspace.uwaterloo.ca/doi/10.1145/2838233.2838233)
- [10] Y. S. Martyastiadi, R. Theodosia, dan S. Prestasi, "Low-Poly Modeling Tokoh dan Environment dalam Desain Game 3D," *Ultimart: Jurnal Komunikasi Visual*, vol. 8, no. 1, pp. 50-57, 2016. DOI: <https://doi.org/10.31937/ultimart.v8i1.459>
- [11] T. Zufri dan N. A. Ardani, "Research on the Application of Pixel Art in Game Character Design," *Indonesian Journal of Visual Culture, Design, and Cinema*, vol. 2, no. 1, pp. 1-7, 2024. DOI: [10.21512/ijvcddc.v2i1.8238](https://doi.org/10.21512/ijvcddc.v2i1.8238)
- [12] Wikipedia, "Pixel Art," Wikipedia, [Online]. Available: https://en.wikipedia.org/wiki/Pixel_art.
- [13] Blender Studio, "Exploring 3D Pixel Art in Blender 4.2," Blender Studio Blog, [Online]. Available: <https://studio.blender.org/blog/3d-pixel-art-in-blender/>.
- [14] X. Gao, K. Wu, dan Z. Pan, "Low-poly Mesh Generation for Building Models," *SIGGRAPH '22 Conference Proceedings*, 2022. <https://doi.org/10.1145/3528233.353071>