Epistemic Prestige in Unreal's Physically Based Rendering

4

In the fall of 2014, NVIDIA proved the moon landing.

Coupling their company's graphics processing unit (GPU) hardware and global illumination (GI) software with Unreal Engine 4, NVIDIA researchers and artists developed a simulation of the July 1969 US moon landing. The team digitally rebuilt the Mare Tranquillitatis—the "Sea of Tranquility" that served as the landing site for the three NASA astronauts on the Apollo 11 mission—as well as the lunar excursion module (LEM)—the bi-sectioned, spiderlike vehicle that would ferry the Apollo team to and from the surface (see figure 4.1). These reconstructions required geometry—the topological digital information that makes up what we could colloquially call the "actors" and "set" of the scene—as well as environmental information. The lighting conditions, visual atmospheric qualities, and camera effects made the scene feel physically grounded and visually real. The graphical rendering of the scene is "photorealistic" in multiple senses of the word. It is photoreal in the colloquial way the term is employed as a stand-in for visual mimesis and also, in more literal terms, as a reproduction of the quirks of a particular photographic apparatus: the combination of the unique lighting conditions on the Moon with Buzz Aldrin's Hasselblad 500 EL camera.

NVIDIA's target was the moon landing conspiracy theory, a long-standing fable that the moon landing was faked by the US government. The landing, conspiracists argue, served as propaganda, highlighting US scientific dominance over the Soviet Union and diverting attention from the Vietnam War. The theory, like many successful conspiracy campaigns,[1] is a

4.1 Screenshot from NVIDIA's moon landing demo (https://www.youtube.com/watch?v=QIap1jL14WU).

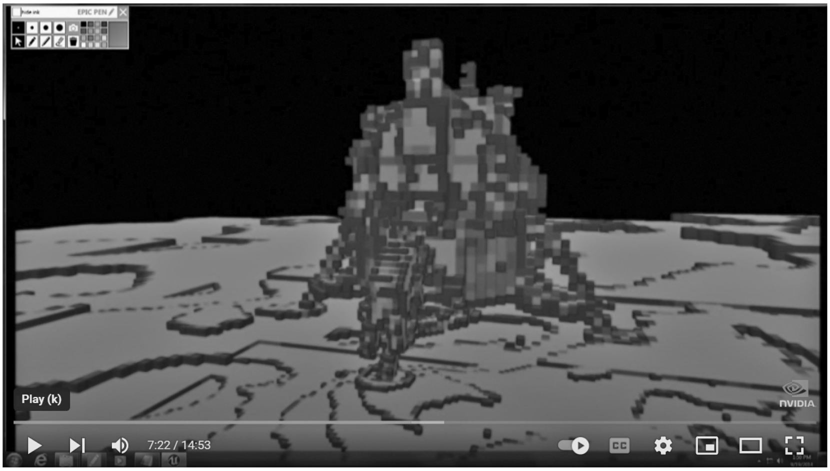
multifaceted, interlocking, and sometimes contradictory series of stories and narrators, filled with half-truths, uncharitable interpretations, dog whistles, and outright lies, circulated on message boards and social media and mainstreamed by right-wing outlets like Fox News.[2] Some of these narratives focus on technical and scientific themes—doubting the capacity to produce the amount of energy needed to get to the moon, claiming the Apollo team would have succumbed to radiation poisoning from the Van Allen belt, theorizing that lunar daytime heat would have melted camera equipment. However, the most well-trafficked claims are visual. They suggest that the United States government collaborated with Hollywood (particularly Stanley Kubrick) to fake the moon landing film and television broadcast.

The ocular-centric evidence levied against Kubrick and the moon landing are wide and varied, and include a fluttering flag in a windless atmosphere and the lack of star visibility in a black lunar sky. NVIDIA's demo, however, was focused on the theory's concern with light—its sources, reflections, ambient quality, and direction. No other single element of the moon landing has been more treated by conspiracists as a stand-in for truth than light, which serves as a useful vehicle to clad antisemitic narratives of joint Hollywood-government conspiracies in the cloth of scientific objectivity. Some of these stories question the naturalness of the light on the lunar surface—how specular glints bounce off of lunar dust, and the length and sharpness of the shadows cast by the LEM and by the Apollo team. Other stories highlight the entanglement of light and technology. They suggest that hot spots of bright areas on the lunar surface are studio lighting equipment, and question the availability of camera technology that allowed a rapid fifty frames per second of film to be captured in an extraterrestrial environment.

NVIDIA's moon landing demo is a fascinating case of transmedia[3] storytelling. Viewers experience NVIDIA's re-creation though a multitude of media formats including press releases, conference keynotes, and video documentaries, alongside simulation footage interspersed with developer interviews. In 2019, a downloadable version of the interactive visualization tool kit was released to the public, which presented users with a real-time rendering of Armstrong, Aldrin, the LEM, and the lunar surface. When the program is first launched, a slow and sweeping camera pans the viewer's eye artfully over the landing site, as lo-fi audio recordings of conversations between the Apollo team and mission control play in the background. The pans are Kubrick-like in their speed and motion, perhaps in a wink to the theory's attacks on the famous director. As the camera moves, the figures of Aldrin and Armstrong are frozen in time, lending an ethereal, diorama-like quality to the scene.

Simultaneously, users are presented with gamelike tools through which they are able to visually manipulate aspects of the scene, such as the camera angle and the lunar time of day. Some of these manipulations directly counter conspiracy claims; they allow users to change camera angles, experiment with the position of the sun in the sky above the landing site to see its effect on shadows, and adjust camera exposure in order to see stars (while also blowing out and making unseeable the lunar surface). Player-viewers can toggle Armstrong's presence on the lunar surface on and off. His bright white spacesuit provides bounced light that brightens Aldrin's descent from the LEM and creates an illusion of a studio fill light on the moon. These interactive systems are packaged with a text-based README file describing how each visualization option directly refutes a conspiracy narrative. Put together, they provide a compelling illustration of how the apparatus of Moon-camera-astronaut produced the unique visual effects seen in the actual photography of the moon landing.

Other tools, however, are designed to highlight the computational work NVIDIA's software is doing behind the scenes. Players are able to view wireframes—skeletal frameworks of the polygonal meshes that make up 3D models—of the scene and turn on and off the material displacement mapping that adds the illusion of depth to flat surfaces. Players can also view the scene in its boxy "voxel mode," which renders the scene as abstracted cubes of color that serve as the backbone for NVIDIA's global



4.2 Screenshot of the moon landing demo's "voxel" mode, which shows the different boxes of light that serve as base calculation units for rendering the scene. (https://www.youtube.com/watch?v=QIap1jL14WU)

illumination lighting and shading algorithms (see figure 4.2). This may initially feel odd, as these options essentially show how even "photorealistic" computer graphics are elaborate smokescreens. The real-time computationally rendered image is a collage of shortcuts, hacks, and visual tricks, all creatively duct-taped together to allow for an image to be redrawn on the screen thirty to sixty times per second. The bounces, color shifts, and refractions of light are particularly complicated to render. The most achievable goal of real-time rendering is not to accurately model the world, but to reproduce in its viewer a feeling of being in a lit world. Exposing the illusion to the user, as NVIDIA does, would seem to undermine a demo ostensibly designed to prove the scientific validity of the moon landing.

It's easy to be cynical about these graphical options. This demo is more about selling NVIDIA graphics cards and Unreal software than about contributing to public scientific discourse. The moon landing demo and the press releases surrounding it highlighted NVIDIA's Maxwell graphics card architecture, as well as Unreal Engine 4's newest implementation of NVIDIA raytracing models. Press releases aside, NVIDIA did not actually "prove" the moon landing, beyond the ways the conspiracy theory has been debunked many times through a variety of media, technologies, and documentation over the past decades. Nor, as this chapter highlights, is Unreal's rendering system a completely physically accurate reproduction of light. But to focus solely on NVIDIA and Epic's marketing goals would gloss over a core aspect of the demo: the visualization tools act as an epistemic and affective tool within the NVIDIA demo, allowing the viewer to peel back layers of light and shape. As the user strips and relayers these visualization tools, bumps in the LEM emerge and subside. Light morphs from rays into blocks and back again, stars brighten and recede in the black sky. The moon landing is presented to the user not only as a visualization of data, but also as a triumph of technological progress over the ignorance of the conspiracy theory. The demo traffics in the affective as much as it does in truth.

NVIDIA's Unreal moon landing demo thus allows for users to play with multiple "ways of seeing,"[4] alternatively seeing as though they were Aldrin's camera; as an imagined highly flexible contemporary camera capturing the 1969 scene; as a cinematic and aesthetic moment replete with camera pans and audio swells; and as Unreal itself, understanding the scene as voxels of simulated light and color. These ways of seeing are each entanglements of truth claims and affect—the wrapping together of scientific validity and technical progress, of the accuracy of the lighting model with American twentieth-century nostalgia. Dominic Kao and D. Fox Harrell have argued that these multiple ways of seeing have always been present and coproductive in games practice.[5] They work together to make each other—in game design parlance—"juicy":[6] the combination of rhetorical and ludic elements used as a rough rule of thumb in game design for whether or not a particular gameplay or visual feature looks and feels "good." They also work together to provide aesthetic and interpretive heuristics: though Epic and NIVIDA ask us to understand the moon landing demo through the lens of truth and visual mimesis, Kao and Harrell note that the kind of scientific viewpoint encouraged here is only one form of visual heuristic that we might think the scene through:

> Many visual methodologies exist, and any one of them is valid: compositional interpretation, cultural analysis, discourse analysis, semiology, etc. How harmonious are the colors? What is the spatial organization? Where is the viewer's eye drawn to? How will interpretation differ across people? How is power being constructed and reproduced? [7]

These visual methodologies are themselves entangled. Critical media theorist André Brock identifies the blending of fact and affect as a core component of all speech, including scientific speech: "logic (logos) depends on a particular style of presentation (objectivism), a particular set of values and beliefs (rationality and positivism), and specific techniques of argumentation (e.g., the scientific method and syllogism) in order to be effective, rendering 'science' as a set of emotional appeals to a specific audience."[8] Light, truth, science, affect, and technology are entangled in the NVIDIA demo.

This entanglement is what makes the demo persuasive. Its attendant README files and video interviews with NVIDIA developers on the promotional website for the demo provide an affective sense of "doing science."[9] Mathematical models and simulation graphics grace computer screens in the background of promotional videos. A host of talking heads of white men present their painstaking processes of discovery and "eureka" moments as they rebuild the landing site. The demo's slow camera pans, highly detailed models, rendering processes, interactive visualization tools, and ambient radio dialogue all contribute to feelings of excitement, wonder, and isolation. Through the demo, the Apollo team, NVIDIA team, and demo audience become linked by a shared immersion in the technical sublime.[10] The technological sublime—the ability for the technological to transcend human embodied limitations and express something greater and more real—is an essential quality of digital media, information technologies, and scientific discourse.

The acknowledgment of affective and discursive systems within technical discourse demands a closer examination of the historical and political legacies of those systems. As Brock argues through Joel Dinerstein, that same affective quality is historically linked with discourses of whiteness-as-technical, whiteness-as-rational, whiteness-as-universal:

> But if one accepts Dinerstein's figuration of whiteness as seminal to the American technocultural mythos, then the characteristics of whiteness—organization, embodiment, disembodiment, and enterprise—can be understood as jouissance, or desires, of new media and information technologies as well. Dinerstein also references "religion"—in this case, Carey's technological sublime—to highlight how relating information technologies to the domain of "the spirit" locates new media and information desire in transcendence. That is, removing the limitations of embodiment from travelling through space and time—or even he identification of a disembodied, ephemeral textual practice—defaults to whiteness.[11]

Race, in other words, is always a part of scientific practice and discourse. Whiteness becomes ontologically and epistemologically enacted with-and-through digital technologies in the NVIDIA demo.

This chapter and the next are concerned with what I label "white photorealism," the coproduction of graphical realism, scientific authority, labor, race, and bodies in computer graphics. Ontologically, how the world is produced in digital spaces is wrapped up in white logics of sublime transcendence. NVIDIA's "god trick" of the positionless, generic viewer unbound by human time and space is intended to make the demo feel *more* representative of the truth of the moon landing, not less. Epistemologically, the organizational structures of knowledge at play—including the mathematical rhetorical style employed by descriptions of global illumination and physically based rendering, as well as the interactive visualization tool kits that layer light and geometric detail on the scene—imply that the "realist" style in the NVIDIA demo is not a style or a rhetoric at all, but rather a clear-eyed reading of universal truth. That the demo is entangled with a pivotal moment in white American history only adds to its perceived epistemic value. Through intersectional feminist critiques of science, I argue that Unreal too participates in the enactment of the broader visual and epistemic construction of race, particularly that of pure objectivity as a component of whiteness, and that of the Black body as a derivation from the white body.

This argument is made in two steps. This chapter takes the first: I examine the history and practices of the pursuit of "good graphics" in Unreal, generally defined as photorealism, and as enacted through Unreal's turn toward physically based rendering (PBR) models and its "materials editor," through which artists access those models. PBR, I argue, succeeds because its aesthetic claims are grounded in cinematic and scientific logics of whiteness—an odd mix of the presumed objective (white) eye of scientific inquiry over that of human experience with the stylistics of (white) cinema. Chapter 5 takes the second step: that the cinematic apparatus, from which Unreal's PBR derives, inherits cinema's material deprivileging of Black persons and introduces new forms of capturing and warping Black bodies against standards of whiteness. PBR is successful in part because of its alliance with whiteness—it clads itself in (white men) physics scholars' epistemic authority, as mediated through Unreal and the broader games industry, and through that "*effective, productive, profitable, exploitable*"[12] authority contributes to the mechanization of racialized labor and the devaluing of Black perspectives and bodies.

Physically based rendering is an older technique in computer graphics that attempts to photo-accurately replicate the laws of optical physics in order to produce visual representations of the world. While initially promising, the technique was rarely used in the 1980s and 1990s, as its reliance on light-bounce calculations, raytracing, and physical simulation of multiple properties of a model's surface made the practice too computationally intensive for real-time rendering. Recent developments in

PBR, however, leverage new software models that allow currently available graphical processing units (GPUs) to render scenes more efficiently using parallel processing, increasing render power without requiring more hardware power. Coupled with more efficient raytracing methods, high-quality PBR render times can be as low as 1/80 of a second, making them ideal for use as live-action, in-camera effects for television and film, as seen in Disney's use of Unreal for set design in *The Mandalorian*. Beyond PBR's technical efficiency, however, the technique is celebrated for its supposed connection to physics—that it represents a move away from subjective judgements of light and color and toward a standardized adherence to "the real." Evangelists for the technique will often frame PBR less as a rendering practice and more like a scientific breakthrough. PBR's narrative of simulating reality, however, is a fantasy. It emerges from what physicist Chanda Prescod-Weinstein calls "white empiricism," or the presumption that the white man researcher has an epistemic privilege on objective, universal reality, whereas other identities—particularly those of Black women—are "produced as an ontological other."[13] Echoing Brock, the draw and relative legitimacy of these physical models are as much affective—and raced—as they are mathematical.

Standardizing Unreal Materials

Of all the work a game engine does, its capacity for rendering graphical output on the screen is by far the most evident to players. Despite the arguments by scholars and designers for more deeply interrogating multiple sensory experiences at play when interacting with a digital game,[14] game culture and game development software remain stubbornly ocularcentric,[15] with new game and game engine releases often accompanied by PR campaigns highlighting new visual technologies at play.

There is a material and cultural tension at play when examining game engine graphics. On the one hand, game engines certainly do have different material properties and practices that produce their visual outputs. These properties are recognized by game developers and players, to the point where game engines become characterized as having particular visual quirks. Casey O'Donnell, for example, has noted the broad perception of the Unreal Engine as having a "shiny" look to its games, due to the particular way Unreal handles lighting and specularity.[16] On the other hand, game style as produced with various engines comes in part from the practices and techniques entangled with those software packages. For example, in an interview with *BioShock Infinite* creative director Ken Levine, both he and *Unreal Tournament* and *Gears of War* developer Cliff

Bleszinski express frustration with the general perception that a game engine determines a game's aesthetics:

Bleszinski: When people [say] "Oh I don't like the way games in the Unreal Engine look . . ." and you're like, what happened was that *Gears* largely defined a visual style for games in this generation because, to be fair it had really good art direction, right? It kind of had an intentionally desaturated look to it, and a grainy, you know concretes and metal showing up.

Levine: And the engine does not.

Bleszinski: It's just pixels!

Levine: It's just pixels. The engine has almost nothing to say about its art direction.

Bleszinski: But what happens is that in certain studios people license the [Unreal] Engine and they'd see the way we build the assets, and they'd use similar specular values or use similar [polygon] counts instead of making their own path for it and so you wind up with stuff you can somehow spot and tell [is made in the same Engine].[17]

While we should be a little cautious with this quote—Bleszinski and Levine after all have a vested interest in making known that their aesthetic and design decisions are not totally driven by the technologies they are using—it does provide a valuable counterpoint to the popular conflation of engine and design. As Bleszinski notes, technique and labor, as much as engine power, are central to the look and feel of a game. Generational aesthetics of games are just as determined by the shared resources and practices of game developers as they are by the software being used.

The discussion of agential tension between engine properties and artist technique doesn't necessarily sell engine licenses, however. Epic Games has certainly worked hard to have the public perceive the Unreal Engine itself as powering the pinnacle of graphical fidelity in games. In 1997, a year before *Unreal*'s release, gaming magazine *NEXT Generation* published a cover story featuring a blocky, polygonal, sword-wielding alien from *Unreal*, provocatively titled "UNREAL! (Yes, this is an actual PC game screenshot)." Though rudimentary by today's standards, *Unreal*'s graphical leap forward from contemporary competitors like *Quake* not only helped sell the game. Such visuals in popular magazines also convinced a generation of first-person shooter developers to license the at-the-time-unnamed Unreal Engine as the graphical backbone of their own games.[18] *Unreal* marketing director Mark Rein made sure in press interviews to hype the toolset as much as he hyped the game: "All you have to do is look at [*Unreal*], how much better the textures look in 16- and 24-bit color and the way it blends, look at the water, and how much better the transparency is."[19] The gaming press was all too happy to oblige Epic's salesmanship. *NEXT* described *Unreal* as "built around one of the fastest, most flexible and sophisticated 3D engines ever designed, running at high resolution 16-bit color. It boasts real-time, multi-colored, and extremely dynamic light sourcing and sports a huge number of the most highly detailed texture maps *NEXT Generation* has yet seen in a game."[20]

Twenty-five years later, Unreal Engine's lighting and graphical rendering packages are themselves promoted as stand-alone products with distinct, marketable names and recognizable promotional campaigns. Much of the press coverage of Unreal Engine 5 (UE5), for example, has focused on the geometry rendering package Nanite and lighting package Lumen. Further demonstrating Epic's "build and they will come" strategy that unifies co-marketing and development, stripped-down beta releases of UE5 were made available prior to its full release that allowed developers to play with—and subsequently upload to YouTube—the Lumen and Nanite packages. The returns were a pseudo-guerilla marketing campaign in which Unreal fans generated photorealistic forests, landscapes, and golden retrievers. Their videos were often accompanied by gushing reviews of the engine, or fans marveling at the impressive technical tricks used by Epic's programmers to keep the engine running and performing well.

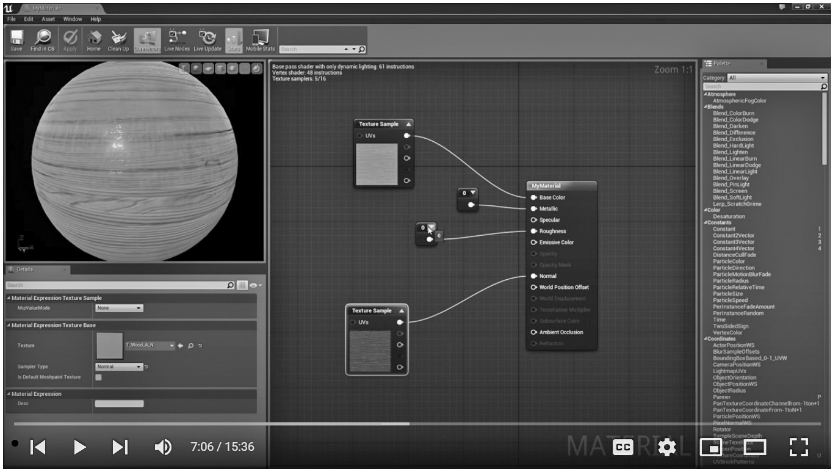
Photorealism is the coin of the realm in graphics engine hype. Stephanie Boluk, Patrick Lemieux, and Eric Freedman have argued that the focus on realistic graphics in games engines has changed how players and designers approach their work.[21] Traditional games and computational media workflows give center-stage objects a proportionally large share of the polygon count in a scene. By contrast, hyperrealist game expectations have led to a "visual economy" in which all assets are treated equally, and even the most banal subjects require "graphical overkill."[22] While that claim may sound hyperbolic, UE5 press releases celebrating the geometry and shadows of pebbles next to a character's feet demonstrate Epic's continued investment in highlighting Unreal's graphical capacities down to the smallest polygon. As evidenced by Bleszinski and Levine's conversation above, this dedication can at times lead to consumer misunderstanding of the Unreal Engine as *only* capable of producing graphically complex games, when in fact the Engine is licensed by companies with a wide array of visual styles.

Games' photorealistic rendering is often situated in opposition to "expressive" or "stylized" rendering; this distinction obscures important historical and technical markers. To begin, we should understand photorealism itself as a style, a mix of affective and aesthetic choices to create visual and narrative mood. Game's photorealistic style is in part inherited from cinema. As cinema studies scholar Julie Turnock demonstrates, realism and photorealism have become naturalized over time, made to be understood as the pure pursuit to visually reflect nature; however, "realism" has had multiple definitions within cinematic history, "emphasiz[ing] different aspects [of reality] at different times."[23] Cinematic realism, Turnock notes, has included aesthetic and political elements varying from appropriate location scouting, actor choice, editing selections, narrative and script selections, and stylistic documentarian methods such as those found in cinéma verité.[24] Only in the late twentieth century, beginning with the visual and commercial success of *2001: A Space Odyssey* and as refined through the major studio effects house Industrial Light and Magic (ILM), did "realism" come to be largely associated with computer graphics and special effects. This new association also popularly redefined the qualities of realism away from aesthetic decisions of filmmakers and instead toward pseudo-scientific narratives of a "natural or inevitable"[25] march toward "better" graphics.

This naturalized definition of computer graphics would also result in the making-invisible of photorealism's aesthetic and embodied qualities. Turnock argues, for example, that photorealism today is best understood as ILM photorealism: that of Industrial Light and Magic's house style. This style is doubly marked, visually by *Star Wars* director George Lucas's desire to synthesize the "credible and totally fantastic at the same time,"[26] and practically in Oscar-winning visual effects producer Dennis Muren's "eyeball test," or "proving the effect's realism because it looks right."[27] ILM graphics became the standard by which all photorealistic styles and practices were judged. While Lucas's combination of the "credible and the fantastic" recall the combination of truth claims and affective style embodied by the moon landing demo, Muren's, perhaps unintentionally, brings positionality into the photorealistic picture as well. What "looks right" to the eye, what is understood to be "credible," depends on the person watching. That both Lucas and Muren were well-connected white men is not lost on computer graphics researcher Ted Kim, who demonstrates how Lucas, Muren, and ILM effectively leveraged white narratives of "the scrappy startup" now common in Silicon Valley to write a history wherein their raced and classed positions had no impact on their cinematic and monetary success.[28]

ILM has an enduring impact on photorealist graphics from a material standpoint, as well. Turnock notes that, post 1980s, computer graphics tools have been shaped to "bend to the ILM aesthetic rather than the other way around."[29] In addition to demonstrating ILM's continued stylistic dominance, ILM's material gravity further blurs the artificial boundaries between photoreal and "expressive" rendering: in Unreal Engine, the processes for deploying the styles of photorealism and expressivism, as well as the shading algorithms that drive their calculations of light and color, heavily overlap. From a game developer's point of view, the vast majority of 3D rendering processes result from an entanglement of four elements in a given scene: the geometry, the position and direction of light, the shaders assigned to each object, and the position and properties of the virtual camera, which acts as a stand-in for the position of the player's body. Shaders are algorithmic processes instanced to each geometric element in the scene. They can be thought of as a layer of magical paint that coats a 3D model. I say "magical" because, in addition to determining the hue and reflectance of an object, shaders are used to create transparency, animations, subsurface lighting effects, glowing effects, and a host of other visual phenomena. Further, through the use of displacement algorithms and height maps, shaders can radically alter the shape of the underlying geometry of the model they are applied to. This shading technique is regularly leveraged in the games graphics pipeline to add geometric detail to simplified 3D models, which allows developers to keep visual detail high while keeping polygon counts (and therefore, computational memory usage) low. Each object in a 3D scene, then, is not just computational clay with a slick coat of digital paint. Each is an assemblage that emerges from the intra-actions of polygonal data and shading processes.

Since Unreal Engine 3 (UE3), game developers have worked on photoreal and stylized shader processes through Unreal's Materials Editor,[30] a node-based graph editor that allows developers to visually build shaders by connecting various datasets and algorithmic processes together (see figure 4.3). When in the editor, mathematical equations, texture maps, time, and space are treated equally, a logic of ontological-flattening-for-interoperability that is present throughout many of Unreal's packages and operational logics.[31] Hybrid interoperability is one of the first major epistemic hurdles new digital artists have to overcome when learning shading in Unreal. Artists must quickly become acclimated to workflows that involve subtracting images from one another, multiplying color by time, and using two-dimensional grayscale maps to generate three-dimensional geometry. There is a palpable feeling of wizardry that comes from successfully creating new visual phenomena that feel irreducible to the textures and equations that constitute them; there is even an entire genre of Twitter clout-play by skilled shader artists posting the cool visual effects they are able to produce through



Base pass shader with only dynamic lighting: 61 instructions
Vertex shader: 48 instructions
Texture samplers: 5/16

4.3 Screenshot showing how to use UE4's Materials Editor from an official Unreal tutorial (https://www.youtube.com/watch?v=sIMmDVLqh1s&list=PLZlv_N0_O1gbQjgY0nDwZNYe_N8IcYWS-&index=3).

unconventionally combining a hybridity of node types. Game shaders, quite literally, are more than the sum of their parts.

Each element in the Material Editor is presented to the user as a material expression: color-coded blocks have input channels and output channels, which are connected with digital thread in an ever-expanding tapestry. The term "expression" doubles as a mathematical and aesthetic term; each node simultaneously represents a single statement of at least two values connected by a mathematical operator and also acts as a step that contributes to the visual and aesthetic values expressed to the user. Unlike Deleuzian imaginations of the ever-emergent, rhizomatic structure of networks,[32] however, Unreal's material expression networks are strictly directional: no matter how complicated the network gets, all connections are eventually plugged into a single unique final expression node. Its output is visibly expressed on simple geometry—usually a sphere or cube—in a separate panel in the editor. Much of the work of the final stages of setting up a material network is figuring out the appropriate expression nodes needed to successfully reduce material expressions into the limited input slots of the master output node, with even many of the most complicated expression networks resulting in only four to five "final" output channels.

The "reduce-down-the-pipeline" model in the expression network workflow reinforces game development's operational logic of interoperability while also providing a new illusion of a standardized shading and rendering process. From the early 1990s through the mid-2000s, the shading process in graphics software (including UE and UE2) made explicit the need to select between different shading models for each piece of geometry in the scene. These different shading models used different algorithmic processes—including different inputs and outputs—to perform their visual work. Often, diverse models were used as rules of thumb to reproduce different physical phenomena and to create distinct visual effects. "Lambert" models—the brightly colored highlight on the surface of an object that gives a glossy or shiny look—for example, have no specular component to them. This makes them ideal for visualizing dull surfaces or for creating flat, cartoon-like coloring. Other times, multiple shader models are more visually consistent but have background processes that impact their viability for a given scene. "Blinn" and "Phong" models each contain specular highlights and are used to re-create metallic or plastic-like surfaces. Phong models, however, have a more complex specular calculation, which gives artists more control over the look of their objects while also requiring more computing power. A developer will make a decision whether to use Blinn or Phong for an individual object based on weighing multiple aesthetic, material, and relational judgements. They consider the visual importance of the object in the scene, the acceptable range of hardware required to render the game, the position of the camera and viewer's angle in relation to the object, and the presence (and computational expense) of other objects in the scene.

In Unreal, game artists are no longer presented with an array of shading models, but instead with the totalizing visual logic of the Materials Editor. Notably, artists *do* choose from multiple shaders, but in a subtler way. Nested across several editor submenus are options like "blend mode," "material domain," and "shading model"; these are different shaders cast as customizable options for a material. Depending on options selected in these submenus, input channels in the final material output node flicker on and off, and the visual qualities of the resulting material radically shift. Though visually understated, the continued presence of these models is important; there are many visual phenomena in the world that can be captured by standard shading models, and then there are "weird" phenomena that complicate the process. These weird phenomena are enacted due to both aesthetic and technical concerns. Some phenomena, like cartoon shading, can't be described using typical photographic mathematical expressions, and thus require custom shading models. Again, these phenomena tend to be clumped together as "expressive" visual rendering. Others, like how light is refracted and warped in a solid translucent object, *can* be described using basic physics, but require so much calculative work that re-creating the phenomena isn't yet feasible in real-time graphics, thus requiring a custom shader to "fake" the effect.

The layout of the Materials Editor does important epistemic work. It encourages the majority of game developers and artists to imagine materials creation as a standardized process with occasional offshoots. This imagination dovetails with increasing pushes toward overall pipeline standardization in game development.[33] This is not an accident: standardization and graphical fidelity go hand-in-hand: as industry-hyped consumer expectations for graphics increase, game developers face pressure to ensure their products consistently look "good" on a range of hardware. Players on desktop computers, multiple generations of home consoles, VR devices, and mobile platforms have been primed to expect hyperrealistic graphics, and as the market for multi-device play continues to expand, inter-device translational demands are skyrocketing. Game engines are beginning to reflect this increased need through standardization. Developers now expect engines to handle device translatability mostly in the background; any advantage an engine can give that eliminates the need to prepare multiple versions of assets or spend time mucking around in the code to boost framerates for individual devices is a massive financial incentive for AAA studios. The financial demands of AAA developers are further recast, often speciously, as being in the best interests of the independent dev community as well. Unreal and its chief competitor, Unity, have each characterized the hardware interoperability their software provides as a "democratizing" force for game development, allowing small and independent studios access to the same cross-development tools as AAA studios.[34] The current state of photorealistic game rendering pipelines, then, is an entanglement of the push for graphical fidelity, hardware variability, the shifting economic realities of the games market, and top-down, industry driven rhetoric of democratizing technical progress, all of which are contributing to technical and conceptual graphics standardization practices.

Physically Based Rendering

Standardization practices always assume some form of universality, either in input (i.e., that the world is made knowable through some repeatable process) or output (i.e., that the standardization of process produces a predictable or stable outcome), or both. From a user interface orientation, Unreal seems to privilege output; the main goal of the engine is preserving a universal play experience no matter the player's hardware paradigm.[35] Probing further into Unreal's rendering mechanics highlights how ontological universality—an assumed universalness to how the world works—also undergirds Unreal's graphical project. Unreal's physically based rendering processes explode the tension between the ontological and the phenomenological, placing the game developer at the center of a material and political fold.

While the Materials Editor is the primary interface for editing shaders in Unreal, much of the look of the shaders themselves is determined by algorithmic processes outside of the artist's control. Because of Unreal's shared source license, many AAA developers take advantage of the technical ability to alter and replace fundamental light and physics calculations at play in the Materials Editor. But for most users, Unreal's physically based materials system exerts tremendous influence over how a given scene will look, and how the artist needs to conceptualize the graphics pipeline.

Unreal's physically based materials are part of a larger push toward physically based rendering (PBR) in the games and animation industries—PBR models are quickly becoming the standardized approach to rendering across the majority of software packages. Despite this push, it was difficult in interviews with computer graphics developers to pin down a single, concrete definition or approach to PBR. What is consistent is the same turn toward naturalization of photorealism evidenced in cinema through the mass adoption of ILM house style. Despite PBR's rendering algorithms historically and aesthetically influenced by ILM, developers described PBR to me as "thinking about the world scientifically, instead of artistically," "modeling how light actually works, instead of how light looks," "a focus on the real world over fake ones," and "a graphical movement towards realism over expressivity." Discussions and research documents belied a fervor for the technique just beneath the surface; not only was PBR considered "more real" and "better looking," but there was a palpable desire for developers to "evangelize PBR to game teams" and artists.[36] The vast majority of developers and research documents I encountered consistently framed PBR as a more empirical, universal, and truthful representation of the world than non-physically based (NPB) approaches. This rhetoric would also reappear in NVIDIA's moon landing demo, which leveraged Unreal's physically based materials to claim that the fidelity of the simulation was a ground truth.

In this, both ILM and PBR's production of photorealism as both scientific and affective reflect the broader aesthetic and epistemic traditions of whiteness as chronicled by Richard Dyer. Dyer, in tracing the etymological use of "white" as connotating both the presence of all colors (as in the optical form of white light) and also the pure and unmarked (as in the blankness of the white canvas), argues that whiteness's duality allows it to operate as both a universal element and something "above" the stains of the world. "The slippage between white as a color and white as colorlessness," Dyer argues, "forms part of a system of thought and affect whereby white people are both particular and nothing in particular, both something and non-existent."[37]

We can characterize physically based rendering as a broad set of techniques that both produce computer graphics and also reproduce whiteness. PBR is an approach to computer graphics that attempts to reverse-engineer the physical properties of light and color to develop a standardized, systematized approach to rendering. Much of this reverse-engineering focuses on simulated physical properties of objects as represented by shader algorithms, such as whether or not an object is metal, how rough its surface is, whether it absorbs or refracts light. PBR derives its name from this simulated physicality. This mathematical standardization is enacted alongside evangelism that preaches the scientific and artistic superiority of universalized models of color and light. PBR is both a hailed scientific achievement and also the mundane representation of everyday life.

While PBR is often hyped as the bleeding edge of computer graphics, the technique is fairly old in computer graphics research; foundational papers on the subject were published between 1980 and 1982.[38] The technique had originally relied on raytracing, a process that tracks individual beams of light in a scene, enabling the artist to calculate soft bounced light, render photorealistic reflections, and change the accumulated color of an individual ray as it bounces off different colored surfaces. Tracing rays is computationally intensive, though, which has traditionally made the process (and therefore PBR) useful only for pre-rendered footage and still imagery. Recently, however, advances in both computer hardware and software have made PBR possible for real-time graphics engines. GPU hardware packs more computing power and forms of data architecture that allow for greater parallel processing, increasing their efficiency. Software-side, some PBR techniques have replaced raytracing methods with the same probabilistic random sampling method now widely employed across economic and political sentiment analysis[39] to determine the direction and color of light. While new methods produce less accurate results, they require substantially less computing time and can be fine-tuned by the artist to look "real enough," making them an ideal calculation method for real-time PBR.[40]

Color in PBR is made up of a combination of the temperature and intensity of light sources in the scene and the shader, which is applied to the objects in the scene. While non-physically based (NPB) shaders like Phong, Lambert, and Blinn are designed to create particular visual effects, PBR materials mimic the way light physically interacts with the physical structure of a surface. The color of a given pixel on a viewer's screen is determined by the shader, as applied to the geometry cast to that pixel. A PBR shader determines color by leveraging the object's base color, assigned physical properties, cast and reflected light, and other nearby objects. Think of this process as taking different approaches to graphically representing an apple. A PBR approach would be to determine the waxiness of the skin, the amount of porousness of the apple, if there are any chemical traces on the fruit's surface, the color of nearby objects, and the surface roughness of the apple. Then it would model an equation that measures how light will refract off and move through the apple given those properties. Conversely, an NPB approach would be to simply pick a red paint that matches the apple's color and gloss.

Despite the push toward universalization, there is no standard PBR formula or implementation across software packages. Unreal's models are based on PBR models developed at Disney for the 2012 film *Wreck-It Ralph*, though decomplexified so as to be viable for real-time rendering.[41] However, most PBR models (including Unreal's) center two core physical models of how light and surface interact: microfacet theory and the conservation of energy.

Microfacet theory abstracts the interaction of a ray of light with the surface and subsurface of an object. Light is modeled as a carrier of information. Human vision interprets the color of light via the angle the light takes to the eye, as well as the qualities of the surface the light has bounced off of. In the classic example, white light—made up of an equal distribution of all human-visible wavelengths of the color spectrum—hits a surface, say, again, an apple. The properties of the surface of the apple absorb many of the white light's wavelengths but reflect some back—in our case, those in the red part of the spectrum. The reflected light travels to the viewer's eye, where the spectral information is interpreted as the color of the surface: the red skin of an apple.

The direction and uniformity of light plays a role in vision as well. Microfacet theory posits that when rays of light hit a surface, their resulting directions are determined (1) by how light penetrates the surface membrane, bounces around inside the object, and exits the object, a process alternately called "subsurface scattering" or "diffusion," and (2) by microfacets, microscopic ridges, deformations, and divots in the surface of an object that deflect light rays. Diffusion contributes to an object's perceived color, while microfacets in the surface alter an object's reflectivity; the "rougher" the surface of an object is, the more it disrupts the uniform informational patterns of light, and the less image information is observed. The smoother an object's surface, the more reflective the object.

Microfacet theory is represented in graphical interface through a new, PBR-specific material channel called "roughness." A grayscale image map becomes mathematically interpreted (a black pixel = 0, a white pixel = 1, a middle gray pixel = 0.5) to drive how broken or uniform a surface is, and thus how reflective it is. A pixel in a roughness map that is closer to black is closer to 0, and thus smoother and more reflective. These images themselves need not be uniform, a quality taken advantage of by artists to introduce "real world grit" into a material. For example, a dark gray roughness map with random streaks of light gray, if applied to a smooth chrome sphere, will make it appear as though the sphere has rough scratches across its surface.

A universal roughness channel replaces the need for different calculations of specular values that marked the NPB era of Blinn and Phong shaders. This is where the conservation of energy enters the PBR equation. Since light is either diffused by a surface or reflected by it, diffused color and reflectivity are mutually exclusive.[42] A reflective surface bounces light almost immediately, limiting the ability of light to scatter inside the object and provide a diffused color. Coupled with the conversation of energy, in which the amount of light that leaves an object can never exceed the amount that was cast on that object, reflective objects' colors appear more black than an artist might expect. Conversely, the more diffuse color an object represents, the less reflection is possible, and the more the color trends toward white.

Finally, PBR's conservation of energy model splits the material world into a binary of conductive versus insulating materials. Conductive materials, almost universally metal, have high reflective values and tend to not scatter light, thus offering no diffuse color. Insulators, like most non-metals, will generally scatter some light, contributing to a brighter diffuse color. Reflectivity also changes depending on conductivity, with metals sometimes tinting the color of their reflections. In PBR, this binary is translated into a "metalness" channel for shaders. The value that PBR artists assign—metal or non-metal—fundamentally changes the physics calculations in the material.[43]

PBR's universalized, physics-based approach to photorealistic rendering has pragmatic and epistemic implications. Pragmatically, the movement toward a standardized shader equation limits the artist's need to manage multiple shader types with varying inputs and outputs (such as the Blinn, Phong, and Lambert models described above). Standardization also makes it easier for developers to move between different game and graphics engines without needing to learn completely new rendering workflows. Further, a physically based, universalized lighting and shading model allows for shaders to appear logically consistent, no matter their lighting conditions. In NPB rendering workflows, it was commonly required for graphics artists to develop separate shaders for the same object for different scenes. For example, an NPB shader for a character's leather jacket might look correct under ambient blue light approximating daylight but appear too dark or too specular under angled orange light in a later sunset scene. This difference in appearance would require artists to swap between "daytime" and "dusk" shaders depending on scene context. The microfacet and energy conservation principles in physically based models, in theory, circumvent this problem, as the leather jacket should appear in any simulated lighting conditions similar to how it would to the human eye in the physical world.

While the PBR artist is responsible for creating visually pleasing images and setting a visual mood appropriate to the game narrative, their render techniques now center a model of a universalized material world. Technical artist Joe Wilson argues that even "fantastical" stories and visions should be understood through the lens of physical reality:

> If your goal is to create a fantastical, stylized world, having accurate material definition is still very important. Even if you're creating a unicorn that farts rainbows, you still generally want that unicorn to obey the physics of light and matter.[44]

In PBR, even fantasy must subordinate to the laws of light and computer graphics' "quest for realism."[45] While this pragmatist approach to rendering may save the artist time, its true strength is saving production and management time. PBR increases studios' profits by both making the individual artist more efficient and creating a systemic practice of production that makes all artist output interoperable.[46] Assets produced by any artist will look the same in any scene, and thus a game's art style depends more on the systemization and managerial directing of an artist workforce, rather than on techniques and tastes of any individual artist. We shouldn't overstate PBR's labor impact, as game production has been increasingly systematized and managed over the past twenty years, even in NBP workflows. Still, PBR provides managers another tool in their toolbox to Taylorize even the most qualitative forms of labor.

Epistemically, PBR further encourages a shift in digital graphics toward physical simulation and systematized production, and away from what Heinrich Wölfflin calls a "painterly" way of knowing—an affective, phenomenological attention to the shape and quality of individual human experiences of color.[47] NPB lighting workflows, for example, often selectively apply multiple competing models of color to achieve different visual and affective reactions in the viewer. An artist may choose to follow complementary color shading, in which the specular highlights and cast shadows from an object are tinted in complement to the object's diffuse color, thus making the object appear more vibrant. A red apple, for example, may have a slight green tint to its highlight and shadow. Complementary color shading creates hot spots of shape and movement that appear both natural and ethereal, drawing in and immersing the viewer into new ways of experiencing color.

While this technique has been used by painters and watercolorists for hundreds of years, and is *phenomenologically* correct—in that complementary shading impacts human perception of color—it is *physically* incorrect from a universalized, mathematics-based model of color. Simultaneously, individual objects obeying unique rules for specular and shadow color introduce variation and unpredictability into the production pipeline, increasing development time and cost. As such, painterly elements like shadow color and specular color are wrested from the artist's control in PBR and are handled by algorithmic processes modifiable only by developers who have source-level engine access.

Diffracting PBR

I have thus far described how photorealism is not grounded in an objective understanding of visual reality, but instead in the cinematic styles and aesthetics of the film industry. These styles have been made natural in part because of the epistemic authority established by white men at the heights of the film industry, as well as in service of the game industry's desire to automate labor. The resulting naturalized industry narrative of PBR is that, as computer hardware becomes more capable, physics simulations get faster and more accurate, leading to better looking—and more truthful—representations of the world. This narrative produces media like the moon landing demo, which leverages the story to enact real-time graphics-as-truth. In the demo, light and physics are universally consistent, as PBR accurately models them, and thus a real-time visualization of the lunar surface can stand in for empirical and historical reality. Realtime PBR is hyped as its own scientific breakthrough. Yet, its mass adoption by computer graphics artists, influenced by PBR evangelists and management-benefiting time and labor reduction, reinforces the game industry's own techno-libertarian leanings. PBR represents an exciting triumph of technology, whose physical accuracy and industry adoption are entangled. PBR proves that graphics technology marches on toward perfectly simulating reality. The moon landing demo's exposing of the toolset behind the simulation thus, for viewers, *reinforces* the truth of the lunar diorama, as achieved through the triumph of technology, rather than undermines it.

It matters, though, that so much of the hype around PBR is grounded in narratives about the technique's grounding in physics. Not only does the rhetorical synthesis of physics and aesthetics mirror the broader interconnectedness of truth claims and affect traced in this chapter, but also the white masculinity of the field of physics itself lends cultural and epistemic validity to the naturalized and neoliberal claims of PBR advocates.

In her foundational work *Meeting the Universe Halfway*, physicist and feminist theorist Karen Barad argues for the adoption of "diffractive" reading and thinking practices in the sciences and humanities. Diffraction is a model in classical physics that characterizes the ripple patterns formed when waves encounter objects, passages, or other waves. Experiments with light in the nineteenth century showed that light exhibited both diffractive properties of waves and ray-like properties of particles. In the twentieth century, diffraction became famously associated with quantum experimentation that demonstrated how matter too follows wave-like behavior. Such diffractive patterns tend to overlap onto themselves to create ever more intricate patterns. Like ripples in a pond, they bounce off each other, the edges of the pond, and objects in the water. The water, the boundaries of the pond, the objects within it, and the objects that pass through it all make up what Barad calls "the apparatus" of the pond. To remove any of them would fundamentally change the apparatus, and thereby change the pond. Barad takes up the term diffraction as a call for "reading patterns of differences that make a difference."[48] Barad continues:

> The shift towards diffraction, towards differences that matter, is really a matter of what physicists call physical optics as compared to geometrical optics. Geometrical optics does not pay any attention to the nature of light. Actually, it is an approximation that gets used to study the optics of different lenses, or mirrors. And you just treat light as if it were a ray (an abstract notion). In other words, it is completely agnostic about whether light is a particle or a wave or anything else. It is just an approximation scheme for studying various apparatuses. By

> contrast, diffraction allows you to study both the nature of the apparatus and also the object. That is, both the nature of light and also the nature of the apparatus itself.[49]

Barad's definition of diffraction is doubly useful for us. First, she suggests that scientific, technological, and visual phenomena are co-constitutive. The scientific apparatus does not unveil an external reality, but is instead an active participant in producing that reality. Studying light, in Barad's example, sometimes tells us more about the mechanisms used to study light than it does about the nature of light itself. This brings us to how Barad's diffractive analysis underscores the rhetorical work that the term "based" does in "physically based rendering." Raytracing literally traces rays of light, thereby, applying Barad, simulating an apparatus of lenses and mirrors more than it grapples with the nature of light and matter. Despite evangelists' claims to simulate physical reality, PBR is not based on a universal physical reality, but rather on a specific arrangement of photoreality. The apparatus of PBR is thus as entangled with histories and epistemologies of the camera as it is with physics.

I have in this chapter already touched on some of the constitutive parts of the PBR apparatus: market forces and managerial logics that reflect intuitional scientific reasoning, producing PBR as a symbol of scientific and technological progress. The PBR model itself is a historically, materially, and politically specific apparatus, rather than the universal translator of light that it is advertised as in the NVIDIA demo. "A model is a work of fiction," philosopher of science Nancy Cartwright argues.[50] "Some properties ascribed in the model will be genuine properties of the objects modeled, but others will be merely properties of convenience."[51] Cartwright notes that while some properties of convenience are idealizations or abstractions of phenomena that make calculations easier—such as approximating light as a ray—others will be "pure fictions." They will contain elements not based in physical reality that make the model function better in conjunction with other laws of physics and mathematics.[52]

PBR is replete with these kinds of properties of convenience. The Disney PBR shading model, on which a vast majority of PBR models—including Unreal's—are derived, is labeled and designed as a "principled BRDF" model. BRDF stands for the bidirectional reflectance distribution function, a function for measuring the interaction of light and an opaque surface that, "describes how a surface reflects light for any illumination direction, any viewing direction, and any wavelength."[53] The BRDF serves as the backbone for almost all models of three-dimensional simulated light and color, including PBR and NPB shading techniques. The principles in the "principled shader" refer not to physical principles and properties, but instead to use case. Its variables, parameters, and value ranges were designed to be familiar to artists, even if that meant sometimes violating or tweaking physical laws. According to Disney programmer Brent Burley, development of PBR shading models was always biased toward the following five principles:

1. Intuitive rather than physical.
2. As few parameters as possible.
3. Parameters are zero to one over their plausible range.
4. Parameters are allowed to be pushed beyond their plausible range where it makes sense.
5. All combinations of parameters should be as robust and plausible as possible.[54]

The apparatus of the PBR BRDF, then, is already a hybrid of the intuitions of physicists and programmers. Physicists have epistemically produced light as a phenomenon to be modeled, and programmers have approached artists (and their "intuitions") as a subject to be modeled *for*. Some of the artist conveniences in the Disney model include the parametrization of values from zero to one (zero being "off," one being "on"), as we have seen implemented in the unique PBR roughness and metalness values. Another convenience is the ability to exceed physical reality "where it makes sense," which should be read as "when the mood of the narrative or visual impact requires it." The PBR apparatus is also biased toward the technical demands of the system it is constitutive of. Part of this bias toward technical demands is certainly reasonable. The principle that parameter combinations must be "plausible and robust" translates to "no combination of variables should break the visual acuity of the shader, nor cause fatal mathematical errors in the software." It would certainly impact artists' quality of life if the shading model they centrally relied on was capable of crashing their software.

But the conveniences of PBR also serve as a lens to illustrate the foundation of white empiricism on which contemporary PBR practices are built. As the principled model aims to ask artists for as few parameters as possible, the limited parameters fed into a PBR model have a large impact on the final look of the shader, making getting those parameters "right" a key concern to game artists. As a result, an entire paratextual[55] industry around Unreal's PBR models has emerged. Artists, engineers, and PBR enthusiasts share parameter data for various real-world materials. Their shared texts range from message boards where artists give advice on one another's shaders, engine and game "postmortem" documentation that details the construction of major game shaders, and infographics that show photographic imagery of real-world materials and the numerical translations needed to create them in Unreal's PBR.

One of the most popular PBR infographics comes from self-described PBR "evangelist" Sébastien Lagarde and his colleagues Sophie Van de Velde and Laurent Harduin at DONTNOD Entertainment.[56] The chart walks new PBR artists through various combinations of diffuse color, roughness, and metalness, and illustrates how successful integrations of these values can produce radically different, yet still physically real, material qualities. Though the chart was first developed in 2012, various iterations and links to Lagarde's blog posts about it still circulate on Unreal development forums today.

In one of these forums, Unreal developer James Baxter asks the most important question for PBR evangelists: "That looks good, I'm wondering though, are those [color] values based off of real-world examples or just what the artist thought looked good?"[57] This question serves as another example of a consistent PBR social phenomena we explore in chapter 5: the epistemic value of an objective, separate "real world" over a subjective visual that "just" looks good to the human eye. In PBR communities and rhetoric, PBR represents a triumph of empirical, objective measurement and simulation. A "real-world example," in this case, is not an artist's interpretation of an image of a material, but rather a measurement of a material's BRDF, as generated by a gonioreflectometer, a complex arrangement of lights, cameras, rotating mechanical arms, and rapid data processing units. The gonioreflectometer is designed to produce Haraway's "god trick," the simultaneous "view from everywhere" and "view from nowhere." By rotating around an object and measuring how specular reflections and shadows move and shift, dependent on the position of the camera, the gonioreflectometer can materially engineer an empirically impossible calculation—how does light bounce off an object when there is no observer to see it?—through the probabilistic stitching together of myriad image datasets.

As Barad's illustration of diffractive light shows us, however, measurements of light are produced by the apparatuses designed to measure them. They reveal to us as much about the material and social arrangements of scientific practice as they do about universal properties of light. Gonioreflectometers can produce findings only for a limited subset of physical materials; if light reflects off an object's surface too uniformly (i.e., the object is too glossy or mirrorlike), it becomes difficult, if not impossible, for cameras to capture the high dynamic range of that object's brightness and darkness.[58] Thus the objects chosen for gonioreflectometry tend to be beneath a certain gloss range, therefore limiting what of the "real world" can be measured and simulated by PBR. Alternatively, they are slightly roughened or deglossed before image capture, producing an inaccurate, but "good enough," BRDF capture. You wouldn't know this from perusing various PBR forums, however; there, decontextualized graphs and tables from scientific papers, and charts measuring BRDFs and other indices of light, are presented as universal truths to be replicated. BRDF indices used in PBR, then, are enacted as representative of a universal material reality through the denial of the material conditions that allowed those indices to come into being.

The "god trick" is not just an ontological claim about the legitimacy of empirical data, or of the philosophical limits of posing an objective world beyond the self. It is also an epistemic and political claim about what kinds of knowledge are made knowable and who counts as a legitimate knower in social and technical regimes. The impacts of what is knowable and who is a knower shapes our social and technical worlds; what counts as a "real" or "true" representation of the real world in a given system depends in part on whose voice in knowledge regimes is recognized as being able to speak to the real, or whose voice is an "appropriate" presence in a discussion of the real. The consternation over whether a PBR guide is based on BRDF measurements or eye is just one example of the contestation of knowledge: in this instance, who is a more appropriate knower, a mechanical apparatus or an artist? While artist knowledge and practice are clearly valued in the PBR development community (hence the artist-first principled BRDF), they are also framed as not an appropriate voice for determining the "true" parameters of a physically based shader.

The construction of "truth" in PBR as based on disciplinary prestige, rather than on other embodied or epistemic positions, mirrors a similar phenomenon in the practice of physics itself. Drawing from Joseph Martin's concept of prestige asymmetry,[59] Prescod-Weinstein argues that physics subfields like high energy physics that have more white men are constructed as more intellectually expansive than other subfields, while also being held to lower standards of empirical proof. String theory, for example, one of the most influential and—thanks to public-facing scientists like Stephen Hawking—popular models of the universe, has no observational or experimental evidence supporting it. Its popularity is instead fueled by a combination of compelling mathematical models, charismatic promises to unify multiple models of quantum gravity and spacetime physics, and the celebrity status of its (white men) proponents. This does not necessarily mean that string theory is *wrong*, Prescod-Weinstein argues. Rather, what is recognized as "prestigious" physics has less to do with the empirical processes of the scientific method and more to do with who is understood as having the capacity to speak representatively about the order of the physical world. This enactment is culturally and affectively powerful. "If you ever want to see physicists get emotional," Prescod-Weinstein quips, "stick proponents of different quantum gravity models in a room and tell them to discuss the relative merits of their models."[60]

Helen Longino and Sandra Harding have, respectively, advocated for understanding scientific knowledge as a form of social knowledge and for a "strong" objectivity that acknowledges how the position of the researcher shapes the outcome of research.[61] Further, Prescod-Weinstein builds on critical social theorist Patricia Hill Collins, who argues that Black women's thought is epistemically suppressed by the actions and cultures of scientific practice. In scientific practice, white men are constructed as more important figures, leading to a citational divide between men and everyone else.[62] It is also useful to read the concept of physics' white empiricism through feminist philosopher Luce Irigaray, who argues that physics' men-centeredness has not only dominated its cultures and practices, but also fundamentally permeated the methods and laws of physics themselves. In her essay "Is the Subject of Science Sexed?" Irigaray argues that not only are the conductors and observers of scientific practice sexed—their claims to rationality and objectivity are enabled by their maleness—but that also the processes and outcomes of science are imbued with Western masculine ideology, which emerges in practice through what Irigaray labels the scientist's "intuition."[63] She provides examples of scientific intuition: "proving the model's *universality*" and "posing *one* world before oneself, constituting a world *in front of* oneself, and of proving that the discovery is *effective*, *productive*, *profitable*, *exploitable*. And this signifies *progress*.[64]"

For Irigaray, the subject of science is doubly sexed: both the practicing subjects and the subjects practiced are sexed as male. To draw from Prescod-Weinstein's lens of white empiricism, we can argue that the subjects of physics, and in PBR, the subjects of light, are also doubly raced: the practicing subjects and the subjects practiced enact white epistemological and ontological frames.

The cultures, epistemologies, and ontologies of physics trickle down into cultures of PBR. Much of PBR's cultural cache among computer graphics researchers descends from its supposed adherence to prestigious models of light and matter from physics. Whereas NPB models merely "look good," PBR is ostensibly "real" (while also, importantly, "looking good"). Because PBR is "universal," its destiny is to eventually be the standard shading model for all graphical scenes, even "fantastical" ones.

Conclusion

Physically based rendering traffics in multiple kinds of white vision and authority: that of Dennis Muren's "eyeball" positionality, George Lucas's blending of the mundane and the fantastic, and the discipline of physics' white empirical epistemic authority. Ironically, PBR's god trick and triumph of science over artist are, paradoxically, both a sign of success and a marked reversal from the pursuit of photorealistic style in cinema from which Unreal's renderer derives. On the one hand, Lucas and Muren's style and models of production were so culturally successful that they became ahistoricized and naturalized. Their photorealistic styles worked, the story now goes, because Lucas and Muren were able to capture something about reality through technology. Simultaneously, the prowess of their well-trained cinematic eyes put them in the best positions to make judgments about how well their films reflected reality, and to build cinematic-technical-computational apparatuses that would come to be at both the technical and economic forefront of visual effects.

However, this naturalized mythos, cinematic and ludic photorealism as natural and inevitable, also sets the stage for the de-skilling of the visual artists who would follow in Lucas and Muren's steps. If what ILM did was discover a hidden, objective truth about the optical world, why can't a formula or algorithm be developed to discover those truths automatically? And while the cinematic eye brought to computer graphics and photorealism through the eyeball test was useful at the time, wouldn't the godlike, computerized eye be even more accurate, particularly if it was trained to follow the objective laws of physics? Who needs artist eyes and labor when the most successful an artist can be is to accurately render the real world—something a proper automated graphics system could do more quickly and more cheaply?

In chapter 5, I explore the ripple effects of the naturalization of photorealism and PBR on raced labor and bodies. As Aleena Chia has argued, the standardization of the game graphics pipeline made possible by shifts brought about by PBR and its translation in 3D image capturing produces a racialized automation of game art assets.[65] The "non-hero" and other background assets and environments most commonly automated through PBR are assets that, over the past decade, have been increasingly produced by gendered, racialized, outsourced labor.[66] Simultaneously, within "hero" characters themselves, the rendering of human skin inherits aesthetic and technical decisions that mark the white body as preferable and the human standard, decisions which, like that of photorealism more broadly, have been rendered natural through technical and cultural processes.

As chapter 5 argues, PBR practices in Unreal reproduce longer histories of race across painting, photography, and cinema, where white skin and the ideal white body are centered as the ideal human form. However, PBR's inheritance of the epistemic privilege of the field of physics allows for a further deprivileging of nonwhite bodies. Unflattering skin tones and representations can be framed not as an aesthetic choice of the author, but rather as the "ground truth" of the physically real lighting scenario—any problems, in other words, become placed on the bodies represented, not in the act of representation. Woven throughout naturalized histories of PBR and their impacts on graphics and labor practices are the whispers of whiteness: who gets to claim best reference to the natural world, whose stances and positions become understood as most objective, whose methods and practices are most naturally aligned with "the real"; whose labor and bodies are necessary—and whose are not.

Enacting Platforms

Feminist Technoscience and the Unreal Engine

By: James Malazita

Citation:

Enacting Platforms: Feminist Technoscience and the Unreal Engine
By: James Malazita
DOI: 10.7551/mitpress/14978.001.0001
ISBN (electronic): 9780262379076
Publisher: The MIT Press
Published: 2024

The open access edition of this book was made possible by
generous funding and support from MIT Press Direct to Open

The MIT Press

The MIT Press would like to thank the anonymous peer reviewers who provided comments on drafts of this book. The generous work of academic experts is essential for establishing the authority and quality of our publications. We acknowledge with gratitude the contributions of these otherwise uncredited readers.

This book was set in Filosofia OT by Toppan Best-set Premedia Limited.

Library of Congress Cataloging-in-Publication Data

Names: Malazita, James, author.
Title: Enacting platforms : feminist technoscience and the Unreal Engine /
James Malazita.
Description: Cambridge, Massachusetts : The MIT Press, 2024. |
Series: Platform studies | Includes bibliographical references and index.
Identifiers: LCCN 2023034695 (print) | LCCN 2023034696 (ebook) |
ISBN 9780262548243 (paperback) | ISBN 9780262379069 (epub) |
ISBN 9780262379076 (pdf)
Subjects: LCSH: Video games—Programming. | Video games—Design. |
Three-dimensional display systems. | UnrealScript (Computer program
language)
Classification: LCC QA76.76.V54 M35 2024 (print) | LCC QA76.76.V54
(ebook) | DDC 794.8/1513—dc23/eng/20231012
LC record available at https://lccn.loc.gov/2023034695
LC ebook record available at https://lccn.loc.gov/2023034696