Episode 2: The Artist and the Automaton

How are procedural generation, automation, and AI algorithms reshaping the role of art and games? How is it impacting artists, particularly in the games indusrty? In this episode, Dr. Sara M. Grimes (Director of the KMDI) chats with Dr. Aleena Chia, lecturer in the Media, Communications and Cultural Studies Department at Goldsmiths, University of London and expert on the cultures of game development, about her research on the ongoing transformation of creative work in the digital games industry.

Dr. Sara Grimes 0:01

Lately, my social media feeds have been flooded with images that my friends and colleagues have made using DALL·E 2. Made by Open AI, DALL·E 2 is an AI system that can make art or photorealistic images based on descriptions provided by users. The results are sometimes absurd or silly, sometimes beautiful or provocative, and often all of the above. But the thing that seems to impress everyone the most is the fact that all you have to do to use this incredibly sophisticated AI system is type in a description using everyday language. With DALL·E 2, you can make pretty much any idea, even a vague one or a complex one, come to life, immediately, and in seemingly infinite iterations.

DALL·E and other popular AI systems have rekindled ongoing debates about the future of creative work in the digital age. What impact will machine learning have on the humans whose livelihoods are based on creating images for art, for advertising, and knowledge media? Will their jobs disappear? More recently, the conversation has broadened to include a closer examination of how these AI systems learned to make art in the first place. DALL·E 2 is trained on existing artwork made by humans and shared

online, from which the AI system extracts formal elements and generates new combinations. It also replicates individual artist styles without permission or attribution. An example widely covered in the news media is how Polish artist Greg Rutkowski has become one of the most commonly used prompts on at least two different AI image generators. Is this copyright infringement? Is it forgery? Is it unfair competition?

The fact is digital tools and algorithms have been changing and challenging creative work for years now. AI generated art is just a recent development in a long-standing struggle to determine how different forms of creative work are organized and valued in our digital society. Nowhere is this more apparent than in the games industry, where creativity in artistic merit are highly celebrated, but it's also an industry where creative work increasingly involves procedural generation, automation, and AI algorithms. How is this reshaping the role of art and games? How is it impacting artists?

Dr. Sara Grimes 02:40

Dr. Aleena Chia is tackling these questions head on, through her groundbreaking research on the ongoing transformation of creative work in the digital games industry. Dr. Chia is a lecturer in the Media, Communications and Cultural Studies Department at Goldsmiths, University of London. She's the co-editor of *Reckoning with Social Media; Disconnection in the Age of the Tech Lash*, published by Roman and Littlefield in 2022, and the co-author of *Technopharmacology* published by the University of Minnesota Press also in 2022. Dr. Chia is a world-leading expert on the cultures of game development, conducting ethnographic work that explores the lived experiences of the people who make games. She examines how ideas about creativity connect with ideals of personhood among game workers and how these connections are stratified across race and class. Her key arguments and theories advance a deeper understanding of how these connections in turn shape the way that game makers use digital technologies in their work and in their play.

Dr. Chia's findings on the changing conditions of creative work in the games industry have been published in numerous articles and conference proceedings, including two fantastic articles published in 2022; "The Artist in the Automaton in Digital Game Production," which appeared in the *Journal of Convergence and The Metaverse*, "But Not The Way You Think; Game Engines and Automation Beyond Game Development,"

published in the *Journal Critical Studies in Media Communication*. These articles both explore major shifts that have recently taken place in terms of the tools that are commonly used by developers to make games and in how creative work is divided and defined across an increasingly global digital games industry.

Through this work, Dr. Chia provides a nuanced and timely cultural and political economic context for understanding how the arrival of new AI systems amplify trends that were already underway, and for thinking more critically about AI's hidden reliance on human in the loop labor.

I'm Sara Grimes, Director of the Knowledge Media Design Institute at the University of Toronto and host of the Critical Technology Podcast. Today, I'll be speaking with Dr. Aleena Chia about her theories and research on creative work, art, and automation in the digital games industry.

Let's just jump right in. What is procedural content generation, or PCG?

Dr. Aleena Chia 05:18

Okay so PCG refers to the creation of game content. Sometimes it's automatic, sometimes it's semi-automatic and it's through algorithmic means. What this means is that you are creating game content, and this can be anything from 3D models, but also terrain maps, even music, creating all these things through limited or indirect user input. What you're doing is that you're defining the parameters and the operations and inputting that in an indirect or limited way.

Just to give you an example, let's say you want to populate a virtual environment. Let's say it looks like semi natural looking. You've got grass, you've got river banks, and then you want to have bridges, because human characters need to cross bridges. This is a vast, vast landscape and you need to place these bridges in a way in which is semi-automatic. You can start to, instead of hand placing these bridges, finding out literally I want to put the bridge here or there... instead, what you can do is that you can define the parameters of what does a riverbank look like, which riverbanks can be bridged, which river banks should be bridged and what are the operations of bridging. What kind of bridge goes where. All these kinds of attributes and conditions ... it's kind of like

writing a formula and then executing that formula across what would be seen as a kind of data set.

Here all the arts assets and the map, all these form a database in which to apply this algorithm or a series of algorithms to in that way. It can also be applied to different things such as, for example, narratives, even animation and other kinds of use cases. You could have something quite simple. You could have a few lines of code. Let's say you want to use procedural methods to create a texture. You use a few lines of code. This is a very established computer graphics method, but it could also be a lot more complex. You could have custom tools to generate detailed architecture of let's say an entire city block and beyond.

Dr. Aleena Chia 07:39

A lot of these tools, some of them are custom made by game developers within the studios, but some of them are also built into commercial software tools. For example, Houdini is one of them. There are also a lot of PCG aspects to, for example, game engines like Unity or Unreal. An algorithm, basically what it is, it's a series of steps. Like I said, it's applied to a set of data and it's designed to reach an outcome. At a more basic level, an algorithm is a computational procedure. What it does is that it automates a process. Oftentimes, this is a process that was performed by a person. Whether it's generating game content or calculating probabilities or even recognizing faces, what it does is that it transforms the task and it redefines the value of different kinds of tasks in creative industries. This has implications for creative work.

People are interested in PCG for all sorts of reasons. Sometimes it's for artistic experimentation. Sometimes indies do this, but also researchers. They use procedural methods as a form of artistic experimentation. They have been used in what are called road light games for many decades. The use cases that I'm interested in are mostly procedural generation in the AAA or blockbuster games context. Here the goal is usually to make labor more efficient, to hire less people, basically. It means that you can do with a few programmers who build this procedural generation system, who input those parameters, you can do with fewer of those programmers instead of a much larger group of asset artists. This is usually the calculation that's being made. This is the

justification that's being used. The idea here is to create a game, and those art assets, cheaper and faster with the same quality.

As we know, games are getting larger and larger. There's this continuous push for vaster game worlds. Games as a service as well is a trend that demands more content on schedule. That's the business model. There are several reasons why PCG is used. Not all of them are compatible, but they all operate on the same basic mechanisms.

Dr. Sara Grimes 10:25

In the article, "The Artist and The Automaton in Digital Game Production," you describe how the industry's adoption of PCG is leading to a bifurcation of creative work into affective and mechanical forms of cultural production. I'd like to ask you about both forms, starting with the affective, which you describe as the tuning or conditioning performed by humans as they "tweak generated results based on what feels right." Can you please explain what this means? What do algorithms have to do with feelings and affective of labor?

Dr. Aleena Chia 11:03

Procedurally generated content is never going to be perfect. You have a system, you have these parameters, and you have output that oftentimes going to be rough around the edges. Artists have to come in and they have to hand author these models. They have to kind of look through all the maps that have been generated and say, "Oh. These ones are not going to work because it leads to too many dead ends here." They have to be tweaked a little bit. They have to be audited after they've been generated. These inevitable glitches that happen along the way, they create temporary work for people. This is called articulation work. There's articulation work that happens, not just with PCG, but across all kinds of computational forms of work, or work that is done with computers and digital systems.

Folks who study automation, they call this the paradox of automation. It's that this desire to eliminate human labor, so for example through PCG, it always generates new tasks for these humans. Often time it changes the quality, it changes the value that is placed on this work. It's like a maintenance work. Instead of design, it becomes a form of

maintenance work. Algorithms, they don't train or tune or augment themselves on their own. This is the paradox at the heart of automation. Algorithmic systems like PCG, they are not simply designed, they also need to be conditioned by humans who clean up and who clean up after these automated systems who seed content, they bridge processes, and they also tune these results that are too difficult or too expensive for these computational systems to undertake. It's not that the computational system can't do it, it's that it's not cost effective to do it that way. It's cheaper to get a human to do it.

This is actually why it's called the paradox of automation's last mile. In game development, articulation work takes on an affective dimension. To give you an example, there are a lot of dynamic and destructible environments in game worlds, things that you can sort of blow up with a sledgehammer or with different kinds of guns. The animation of those destructible environments they have to be specific to the different kinds of materials that they're created from, as well as the force that was created by the weapon that was destroying it. All that needs to feel right. When you use a sledgehammer on a desk, a wooden desk, and specifically a 1950s mid-century modern desk, it's going to splinter in a specific way depending on the veneer. You can have a grasp of how complex these things are. It's hard to program all the parameters of all these different materials.

Instead, this is done procedurally. There is a system in which all these different materials and then all their interactions, they will animate. The destruction will animate in a specific way, but a human being needs to come in at that point to sort of figure out does this feel right? Physically, this may be correct, but it may not feel good. It has to feel good for that player to destroy part of that environment. This is where the human becomes an important aspect of these kinds of automated systems. That's why this focus on the affective dimension is an important part of the puzzle of thinking about game development within the game industry more broadly.

Dr. Sara Grimes 15:04

The second category is mechanical, which you argue includes both automated and outsourced labor, understood by those in the games industry as replicable, deterministic, artistic, but not truly creative. I'd love it if you could tell us more about these mechanical forms of creative work.

Dr. Aleena Chia 15:23

Within the games industry, there is a separation of two forms of art creation. The first of them is called ideation or is referred to as ideation. This relates to giving detailed documentation of what certain kinds of environments and objects should look like. It's kind of like a blueprint. Then there is another form of art creation, which is referred to as production. This means taking that blueprint, taking that detailed documentation, and then executing that blueprint into, for example, a 3D model, or texture, or a terrain. The splitting of ideation from production is a huge part of how AAA games get made, because these games are so huge that you need sprawling teams that span all the way from Montreal all the way to Vietnam and China. These are huge projects that require, oftentimes, a lot of outsourcing or what is known as external development.

In-house artists, artists that are based within the studio itself, what they're told is that they should focus on higher level conceptualization. They should only work on the blueprint. Whereas, outsourced artists, they can handle the modeling and texturing. The discourse here, or the way that this gets talked about is that in-house artists can be freed from the drudgery of modeling and texturing. Already you can see, just based on the categorization of this work, the modeling and texturing is seen as in a way less valued, in a way, less skillful. This was really interesting to me, because the outsourcing and automation were used almost interchangeably. Developers would talk about how just outsource it, just automate it away.

These two kinds of art production would be referred to as scalable art solutions. What happens here is that in both cases, in the case of automation, but also in the case of outsourcing, there is a devaluation of work that can be automated. This kind of way of talking about automation and work that's boring and repetitive, it's something that happens not just in the games industry, but it also happens in creative industries more broadly, that are undergoing automation. There is this popular way of talking in which automation is a good thing, because it helps artists to be more creative. It helps reassure human workers that they're not going to be made redundant. There are a lot of workers in the games industry that feels that their jobs will be threatened, and this is a way to reassure them by telling them that. They are after all artists; they can focus on ideation instead of production or execution.

Dr. Aleena Chia 18:37

What I also found interesting within this context is that there was a manualization of digital design. If you think about it, modeling and texturing, it's done on a computer. It's not really manual in that way. Yet it was talked about as if it was manual labor. Using the hand. A lot of terms came up, like hand placed, hand modeled, hand built, hand edited, and even hand designed. This is part of the argument that I'm trying to make in that when a process gets automated, then the task itself, it becomes manualized in that way.

Just to give you an example, outsourcing vendors would talk about their artists in countries like Vietnam, countries like China as having skill when it comes to the technical. They would say explicitly that you can't expect them to do your art direction, that you're not paying them to be creative. Depending on the blueprint that you give them or depending on the detailed documentation that you provide, you will get that back exactly. If you send them ugly, you will get ugly back, except that it'll be very well executed. They're not paid to be creative, they're not paid to have the right feel for something. In fact, they're not even expected to feel. They're not paid for the expertise as a whole person with memories, with biographies, with cultural capital. Instead, they're seen almost as a mechanical component.

My concern with this is that the right feeling is a kind of cultural capital that is based on gender, that is based on race, as well as class. Having the right memory of a brand when you're growing up, having a feel for a franchise, all of these things, they can't be trained. They're part of our biographies as human beings. Growing up in the right place, having the right friends, playing the right games, and even wearing the right clothes, all these are part of the cultural capital that we inhabit and that some of these game workers are using, as a whole person, in order to decide what feels good and what doesn't feel good.

Dr. Sara Grimes 21:08

One of your core arguments is that these processes and their resulting stratification of creative work are deeply informed by racial capitalism. Drawing on the work of Lisa Lowe, you describe on page 396 that "radical capitalism foregrounds how global

capitalist expansion assigns differential value onto labor, resources, and markets across regions and populations according to colonial divisions". Can you please say a little more about this?

Dr. Aleena Chia 21:40

Racial capitalism is a useful concept here because what it does is that it looks at racialized labor as fundamental to the emergence of modern capitalism. Colonialism, and imperial expansion, like all these things, they came into play to make capitalism what it is. What we understand as capitalism is really racial capitalism. It's really shot through with all the dynamics of race and the way in which it differentiates between people and places. This differential value that is placed on different human lives and different human labor, this is really part of the story of capitalism. Just to quote from Lisa Lowe, racial capitalism, it captures the sense that actually existing capitalism exploits through culturally and socially constructed differences such as race, gender, region, nationality, and so on. Capitalism really relies upon the elaboration and reproduction and indeed the exploitation of racial differences. More fundamentally, race making is really intrinsic to the process of capital accumulation, because what racism does is that it supplies the precarious and exploitable lives that capitalism needs to extract that land and that labor.

Why this is a useful framework to work with is because it helps to scaffold a way of thinking about these artists. Why is it that some of these artists, they are valued as a whole person for their feelings and their biographies, whereas others are only valued for what their hands can do within a specified task? One scholar that I want to highlight that works within this framework of racial capitalism is Lilly Irani. In an article she describes how creativity in, what we understand as design thinking, within engineering schools, is assumed to emerge from a personal biography. Think about Steve Jobs. Steve Jobs is a brand in and of himself that is based on his entire biography, not just the single output. In this sense, there is a kind of racialization of what creativity can be. It's linked to automation in a way in which some people are seen as more creative, whereas other people are seen as more mechanical.

Dr. Aleena Chia 24:30

If you think about the way that Asians, East Asians as well as South Asians are portrayed in the media in which they are seen as more robotic and less creative. This is a way in which racial capitalism operates to use creativity as a differentiator of different kinds of value that are placed on tasks, that are placed on creative outputs, but more importantly placed on human workers. Automation anxiety, it goes back much further than PCG. It can be traced back to assumptions of who is a fully human subject. In many ways, all this work that was repetitive and dull and boring, what these developers called shoveling assets through pipelines, the enchantment of automation in a way is about removing this kind of work. What it forgets or what it leaves out is that this is the work that has always been done by people who are racialized, minoritized, and feminized in many cases. Their labor has always been seen as boring, derivative, expendable, mechanical. What's important for me here, interchangeable, with automatic processes.

Dr. Sara Grimes 25:59

I'd like to circle back to the broader industry trends that these shifts emerge out of. You describe many of these processes as resulting or linked to the adoption of procedural systems in game development, something you refer to in the article and in some of your sponsors here today as the PCG pipeline. What is the PCG pipeline? How is it different from previous forms of cultural production in the development of games?

Dr. Aleena Chia 26:27

Within the games industry studios that make blockbuster or AAA games, they oftentimes have to work in very large and geographically distributed teams. Within this industry, there are many kinds of disciplines. Artists have a certain kind of expertise, engineers have different kind of expertise, musicians, and so on and so forth. They work on different parts of the game. Game development is modular, especially at this scale. If you're making a large game, it has to be done in a modular fashion. It gets assembled by project managers at different stages of the game development process. When you think about modularity in this way, PCG, it kind of changes the stakes a little bit, because it's not just modular in which different things are coming together at different times, there are a lot of interdependencies, very complex interdependencies. The workflow becomes exceedingly important.

To give you an example of why this modularity is different in the PCG context, once you have a system in place that generates, let's say, shrubs across an entire terrain. During the articulation process when you clean them up a bit, artists have a little bit of flexibility to change and tweak, but they don't have the flexibility to actually change anything fundamental about that design. Let's say you need to change the structure of that shrub. You can't do it by hand, you need to go into the PCG system. You need an engineer to then change the parameters of how that shrub is generated across all the terrains. You can't just change it in one place, you have to change it everywhere. That's why the PCG pipeline has a kind of different structure in which artists are in fact disempowered. What this means is that there is a kind of assimilative force that is at play within PCG systems.

This complicates how we usually think about human control because here the engineers or the technical artists, they have to design what is tweakable. They design the control points for artists to then come in and do that articulation. In this sense, engineers or technical artists, they design the tools. By doing so, they define what tasks should be automated and what tasks require articulation or handcrafting. In this sense, it becomes a very political role that engineers are taking up, because they then define what is valued within the art production pipeline. They also decide, not just behind the scenes, but also what games look like, what aspects of a virtual world ends up being procedurally generated, what aspects of that virtual world gets a lot more care and attention. In this way, toolmakers, they get to define the value of different kinds of work. What is deemed as shoveling assets through pipelines and what is reserved as creative work that artists find meaningful.

Dr. Sara Grimes 30:02

Your article entitled "The Metaverse, But Not the Way You Think" focuses on a different but closely related industry trend that also emerged over the past several years, the rise and spread of game engines such as Unity and Unreal. Game engines are often seen as making game development more accessible, to a wider diversity of game creators, but this article shows that they're also really important and asymmetrical power relations at work. How do game engines fed into these broader transformations unfolding in the realms of game development and creative work?

Dr. Aleena Chia 30:39

Game engines seem like they're helping designers out. On one hand, it seems like they're giving you more tools, they're giving you more options to take your vision and bring that to life. Game engines automate the creation of art assets and the writing of code. The labor of programmers becomes economized in a way. If you want to make a game and you're a small team, you don't need to write code for core mechanics. While this sounds like a positive move, and it is a positive move, it does democratize game development. Game engines are a kind of double edged sword. At the same time, engines also have a kind of constraining way of working in that they enforce design standards. They, in fact, streamline different kinds of game making ideas and techniques based on the way in which their tools work. Similar to PCG, there is a way in which these tools, they define what can be done with them. They also define the kind of value that is placed on different forms of work.

Within the pipeline engines like Unity, they integrate with different kinds of 3D graphics applications. Maya is one of them, Blender is another one of them. By doing this, even before any kind of artistic work is done, they are already locking developers into certain kinds of workflows and certain kinds of formats. This has to do with strategic forms of interoperability between the game engine and different software packages for digital creation. Unreal has its own ecosystem. Unity has its own ecosystem. That's one way in which this lock in happens. This also happens with distribution. To give you an example, game engines like Unity, they lock developers into certain kinds of distribution outlets. There is a store and a marketplace on which you can sell your game. In this sense, there is a kind of vertical integration. That's why folks like Maxwell Foxman and David Nieborg have talked about game engines as platform tools because they in fact consolidate power and they consolidate equity in these really monopolistic ways.

Dr. Sara Grimes 33:14

My last question is the one I'm asking all of my guests this season. What should we all be thinking about when considering the political dimensions of digital technology?

Dr. Aleena Chia 33:25

Digital technology and automation, they're not synonymous, but they're increasingly intertwined with each other. Within this context, we have to really wonder what is this human in the loop that's become kind of a lexicon of popular AI? How does the human figure within these kinds of increasingly automated systems? How does the human figure in terms of tasks that are increasingly automatable? Predictive studies of the US and the UK, they suggest that creative occupations are relatively future-proof as compared to other kinds of jobs. As compared to, for example, jobs that involve manual work or work that requires less flexibility. Even though creative work is more future-proofed, this very much depends on the job category and the sector of employment. Hybrid roles, these are the ones that are safer, because they combine creative as well as technical skills. Technical artists with programming skills to build and maintain these tools, they have less risk of losing their jobs.

The way the games industry is stratified engineering roles, versus art roles, versus support roles, these are all exceedingly racialized, stratified by gender class and so on. When we think about which jobs within this broad category of art roles are future approved, we have to really think about the specificity of that role, what kind of hybrid expertise it brings together, and its sector of deployment. Affective conditioners of PCG, they will play an essential role, but this role will be downstream from automated processes. What's important to really consider when we think about this human in the loop, is who will have access to the right feelings? Artists who lack the right feelings, who don't have the kind of techno masculinity that is valued within the games industry, but also within games as a practice more broadly, they will be limited to mechanical conditioning. Instead of operating as controllers or even complementors to these kinds of systems, they're going to be increasingly extensions of these automated systems. That's something that we need to pay attention to.

Dr. Sara Grimes 36:15

A big thanks to Professor Chia for joining us today.

The Critical Technology Podcast is produced by me, Sara Grimes, with support from the KMDI. Audio mix and sound design by Mika Sustar and Mehrdad Ranjbar. Music by Micki-Lee Smith. Theme song by Taekun Park. Our logo was designed by JP King. The

artwork for today's episode was created by Kenji Toyooka. Please subscribe to stay up to date on new episodes and posts as they become available. Thank you for listening.