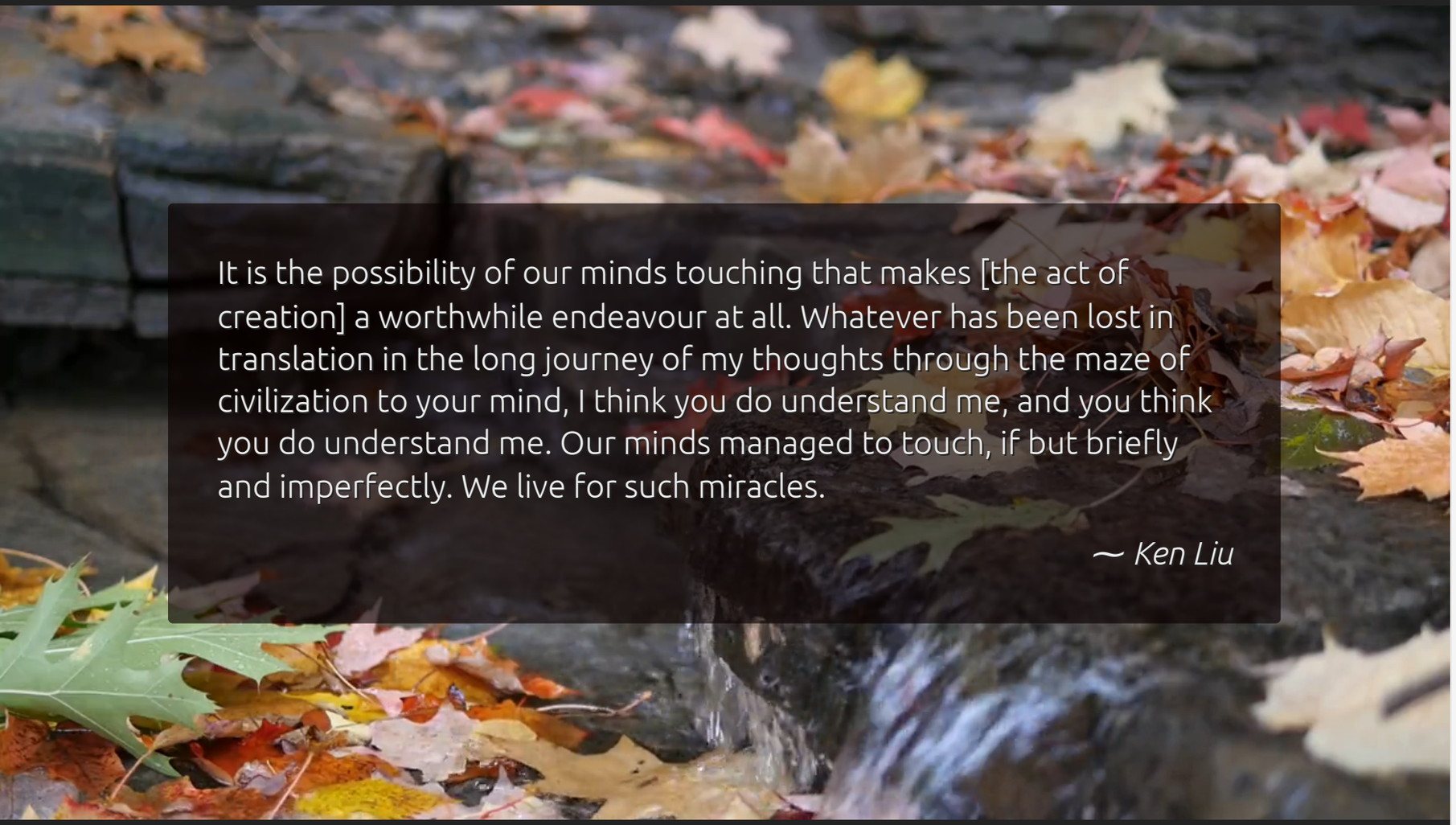
Welcome to Past, Present, Future, the second of five tutorials in the series.

It is entirely possible to make art with machine learning techniques without knowing much of the history or current context. I've done so myself. However, just as the nature of digital things has ethical implications (as we discovered in Part 1: Foundations), the history of digital things can change and deepen your relationship to your digital tools and create a connection, no matter how tenuous, with the complicated and amazing scientists and artists who came before you.

To paraphrase Ken Liu, from the preface to *The Paper Menagerie and Other Stories*:



"It is the possibility of our minds touching that makes [the act of creation] a worthwhile endeavour at all. Whatever has been lost in translation in the long journey of my thoughts through the maze of civilization to your mind, I think you do understand me, and you think you do understand me. Our minds managed to touch, if but briefly and imperfectly. We live for such miracles."

We'll begin with a look at what exponential development, in this case technological development, looks like because it is hard for humans to conceptualize. To really get a handle on how things change over time it is important to wrestle with this.

We'll go over the related histories of math, computers, neural networks and artificial intelligence. As with most scientific progress, the foundations were built relatively slowly but are critical to the incredible growth today. A better understanding of these fundamentals has changed how I look at all digital technologies.

We'll explore the waves of research; from successful prototypes, growing hope and promises, to subsequent disillusionment from failures suffered in all the different approaches to machine learning. We'll hear about the disagreements about the true path to machine thinking.

Then we'll survey a diverse set of art generated by machine to give some context to how artists were using software and ML research.

That will bring us to the present, where we'll critique the main drivers of current ML research and take a brief look at current tools available to artists.

Finally, I'll extrapolate these trends to the future, and some changes I would like to see.

This presentation is much more self-directed than Part 1: Foundations. There is much more to read, but those are details that are unnecessary for understanding the main concepts. Anything that I think you shouldn't skip I'll narrate. The rest is there to provide extra context when you want it. I encourage you to take breaks or come back to the unnarrated sections for reference at a later time.

Red text indicates hyperlinks, many to Wikipedia for more information, but the ones above will take you directly to the different sections and if you're short on time then I suggest you start at the present.

Humans tend to think linearly, so our intuition gets exponential growth wrong. We can overcome this using math, a technology that allows us to transcend our natural biases.

We will be focusing on the exponential growth of technological development. Exponential growth as seen in computer hardware means that every 5 years a computer is about 10 times better with no increase in price.

Technologies for better tools, especially those for learning, like writing, educational institutions and the practice of science, dramatically increase the speed of development. A couple of thousand years ago we might see important inventions every few hundred years and often they would be lost and rediscovered, then after about 1700 we started seeing significant inventions every year or two. Two hundred years later its every couple of months, twenty years ago its every month, and now, there is interesting, potentially important research being released every day. Ray Kurzweil estimated that at the rate of development in year 2000 we'll see the equivalent of 20000 years progress in the 21st century.

For reference at the turn of the century, the rough draft of the humane genome project was released, Wikipedia was started, and the following consumer gadgets were released - the first camera phone, USB flash drive, Playstation 2, Bluetooth, the Segway, The Sims, the ASIMO robot, and the world's first mass-market e-book. Add we survived the Y2K bug by spending hundreds of billions of dollars and mobilizing most governments and large businesses.

How do you turn 100 actual years into 20000 years of development? It's not just technology helping to create more technology. Cultural changes play a big role, for example, development is constrained when women, minorities and the poor are excluded from participating in, business, arts and sciences. The directions of any development are further constrained when power, personal wealth, and fear drive investment, limiting the growth in other areas. Thus there substantial headroom for increases in the speed of development through better living standards, greater diversity, accessibility and inclusivity.

This isn't to make a judgment about the accelerating rate of development, positive or negative, simply that it is increasing and will continue to do so, despite the numerous dangers from reckless, thoughtless or exploitative developments. Social media being a good case study. What matters is that we get a sense of, then acknowledge and plan for, exponential technological development.

In the following video excerpt, done for my 2015 concert *Creo Animam*, I depict images of technology ordered by the year of their invention, timed such that 1 minute of video is roughly equivalent to 100 years of historical time.

Credits

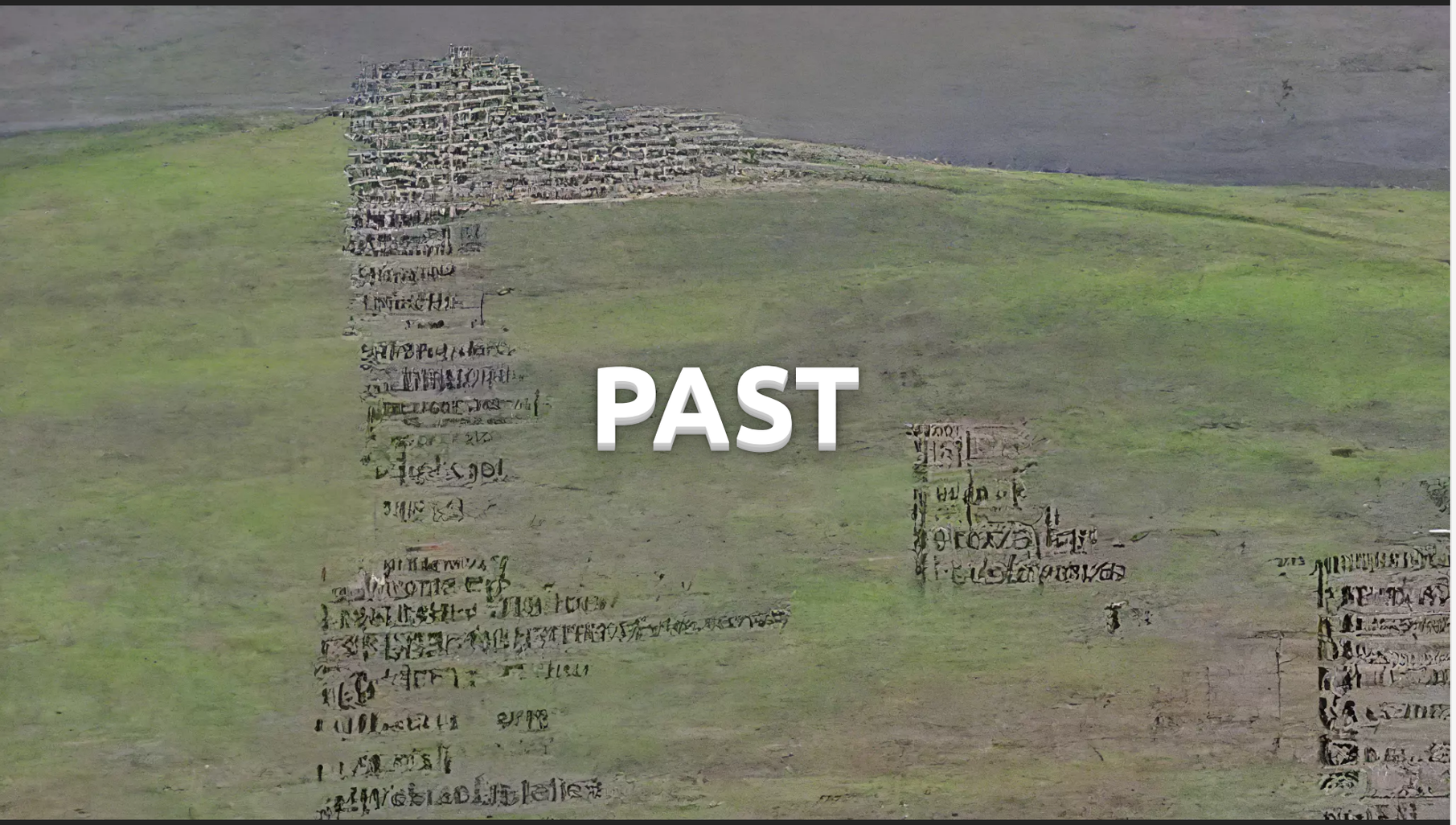
- *Machines dreaming of technology* - Ryan Kelln

Credits

- https://www.kurzweilai.net/the-law-of-accelerating-returns
- https://en.wikipedia.org/wiki/2000_in_science
- https://en.wikipedia.org/wiki/2000s_in_science_and_technology
- https://en.wikipedia.org/wiki/Wikipedia
- https://www.computerhistory.org/timeline/2000/

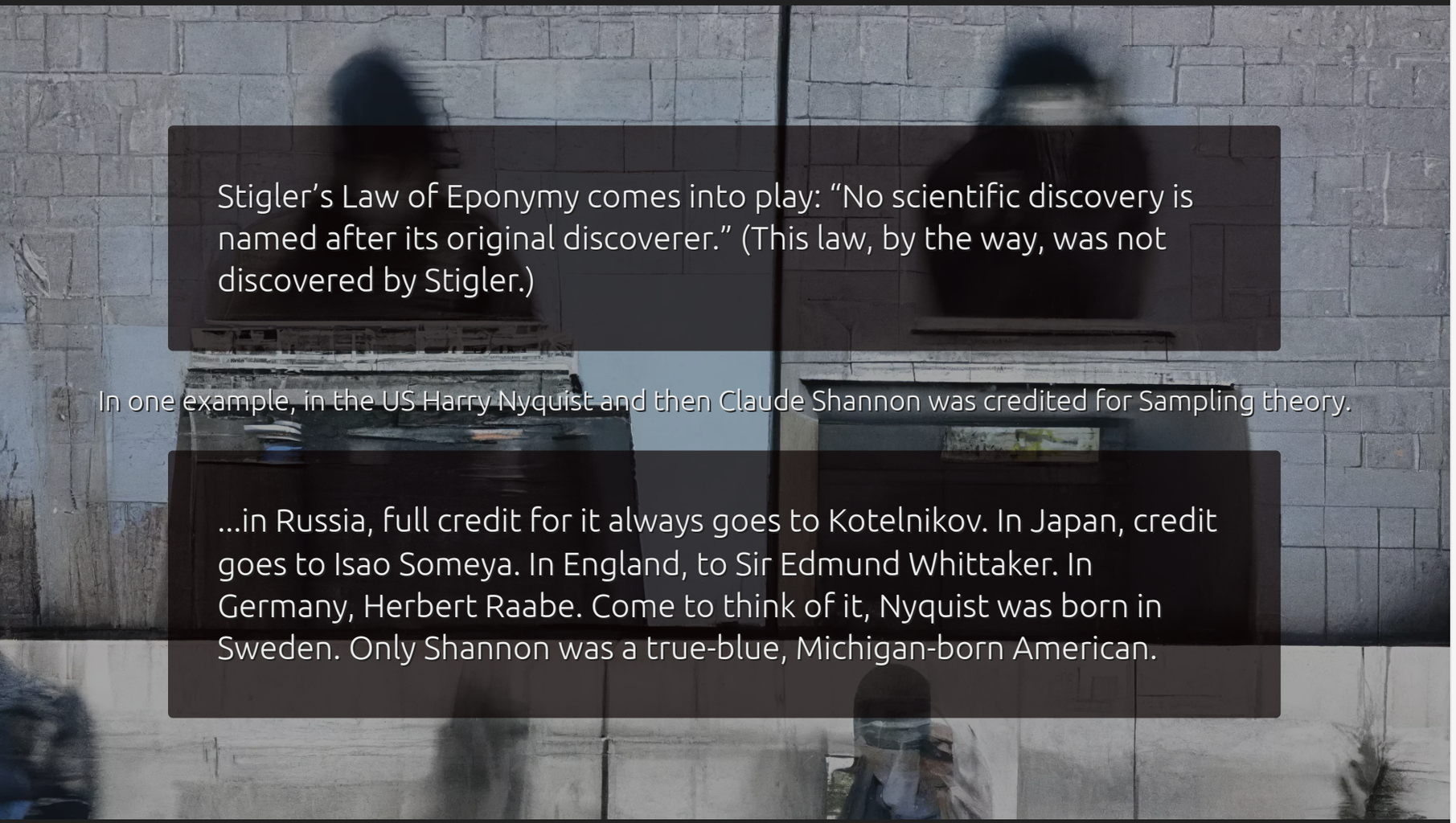
Credits

- *NNNAAAMMM* - Ryan Kelln, Eric Kovalevskyy (2015)



I'm not a historian, and the following subset of historical events is not meant to be exhaustive, but rather illustrative of the role that science, mathematics, mechanical and digital computers played in the development of machine learning and art. There were countless developments in basic math and technology before the 20th century, but I am focusing mainly on the 20th century.

It is important to remember that recorded history is heavily biased. Dates and timing are approximate and the people who are commonly associated with various breakthroughs are surrounded by people whose names aren't mentioned but contributed in some way. In most cases multiple people around the world were working on and solving the same problem independently at around the same time. In Alvy Ray Smith's *A Biography of the Pixel* he has this great example:

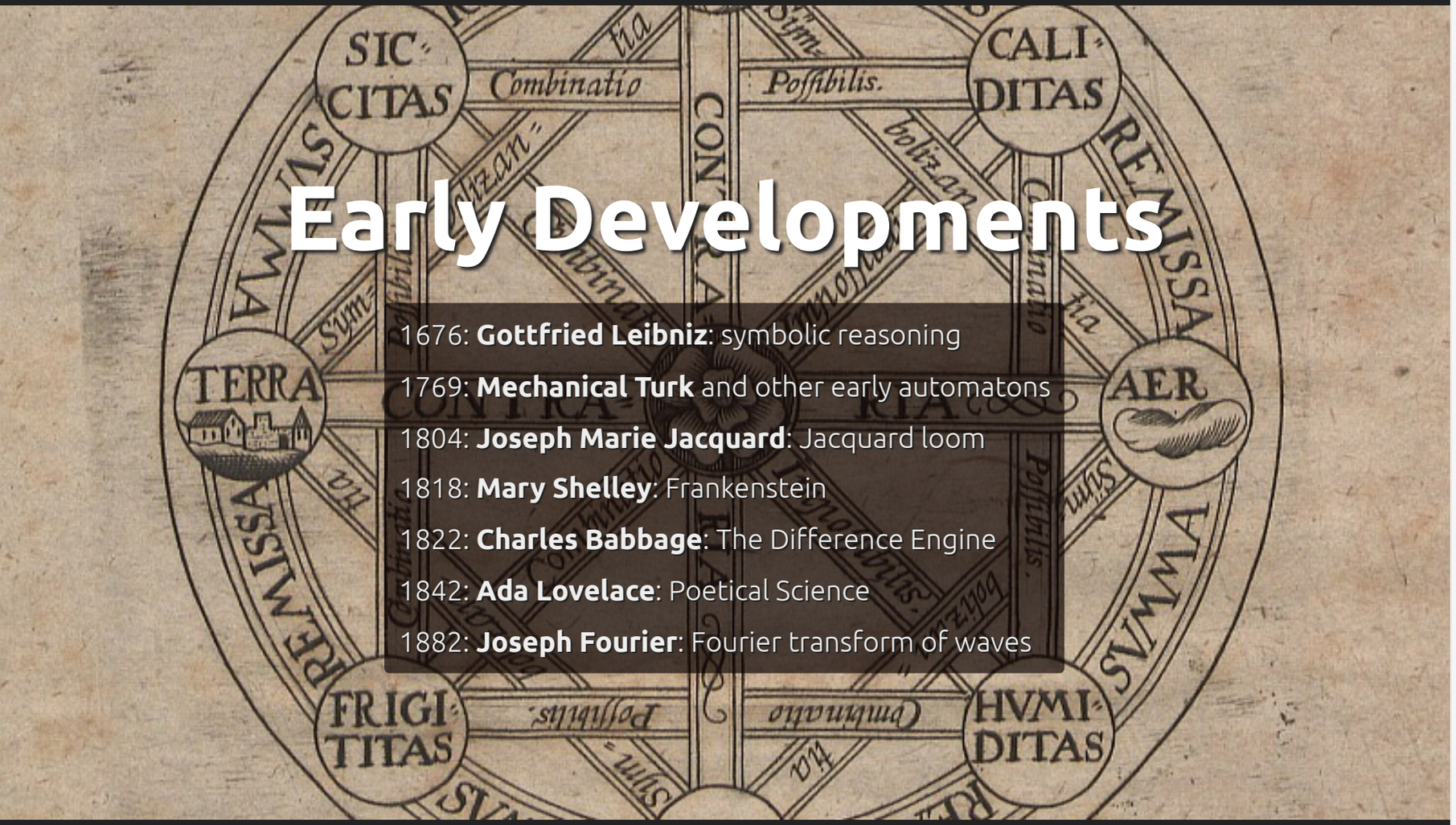


"Stigler’s Law of Eponymy comes into play: “No scientific discovery is named after its original discoverer.” (This law, by the way, was not discovered by Stigler.)"

In one example, in the US Harry Nyquist and then Claude Shannon was credited for Sampling theory.

"...in Russia, full credit for it always goes to Kotelnikov. In Japan, credit goes to Isao Someya. In England, to Sir Edmund Whittaker. In Germany, Herbert Raabe. Come to think of it, Nyquist was born in Sweden. Only Shannon was a true-blue, Michigan-born American."

So be forewarned, the people credited below did great work, but are heavily skewed towards white North American men. I encourage corrections and broader perspectives.



Early Developments

1676: **Gottfried Leibniz**: symbolic reasoning

1769: **Mechanical Turk** and other early automatons

1804: **Joseph Marie Jacquard**: Jacquard loom

1818: **Mary Shelley**: Frankenstein

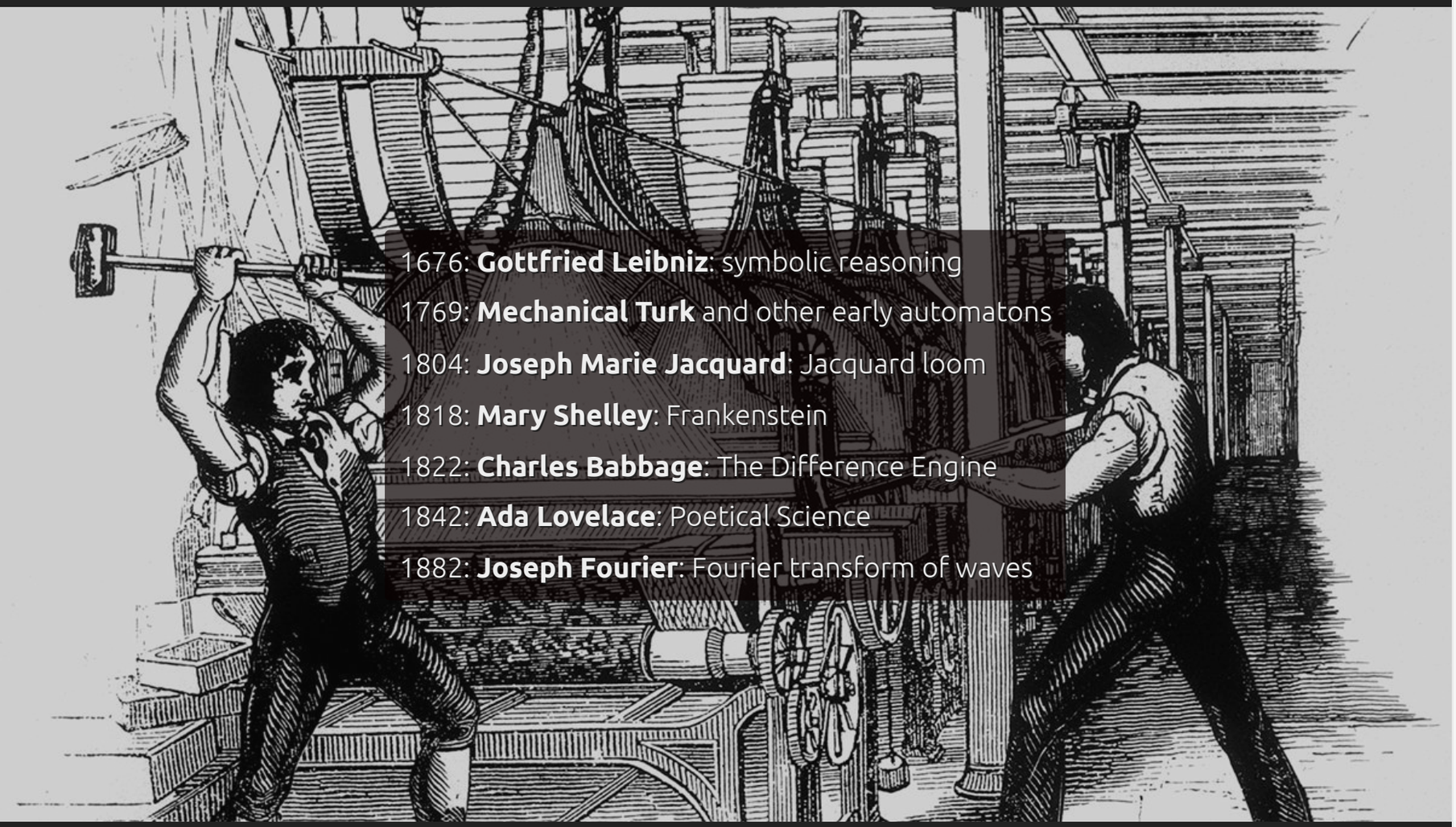
1822: **Charles Babbage**: The Difference Engine

1842: **Ada Lovelace**: Poetical Science

1882: **Joseph Fourier**: Fourier transform of waves

There are many good places to start, but I've chosen Leibniz, who is a less well known contemporary of Isaac Newton, who also is credited with the development of calculus. A deeply religious man and brilliant thinker, Leibniz imagined an alphabet of human thought that inspired scientists for centuries.

For those less interested in high concepts, mechanical marvels could inspire awe. Sophisticated mechanisms, or automatons, were created for kings and other powerful patrons as objects of wonder, the ancient version of The World's Largest Wheelbarrow. With the rise of merchant and capitalist power, these mechanisms were being made productive.



The textile industry in England saw the effects first. A punch card controlled device fitted to a loom simplified the manufacturing of complex textile patterns and brought about massive employment changes. This demonstrated the utility of programmable machinery, inspiring the invention of a mechanical computer. At the time computing was done entirely by "unskilled" human labour, men who were taught only addition and subtraction, and followed recipes to calculate tables of logarithms and nautical almanacs. This work was both error-prone and critical to science and naval power. Few imagined that mechanical calculation could be used for other pursuits.

Science in all areas was progressing rapidly, and later that century, critical understanding of the math behind waveforms, such as sound and light, was developed and would lead to the conversion between digital and analog information.

Credits

- https://en.wikipedia.org/wiki/Luddite
- https://en.wikipedia.org/wiki/Computer_(occupation)

1676

Gottfried Leibniz

In the late 17th century German polymath Gottfried Leibniz wrote 15,000 letters and 40,000 pages of other manuscripts, most of them unpublished. These thoughts anticipated notions in countless fields of inquiry including computer science. Along with René Descartes and Baruch Spinoza he was an early modern rationalist who believed that truth was not found by the senses but by deductive reasoning.

In contrast to Descartes and Spinoza, Leibniz attempted to understand the universe by finding the principal components, which he called monads, that could not be divided into smaller parts. God was the first monad or monad of monads. He applied similar thinking to ideas:

1. All our ideas are compounded from a very small number of simple ideas, which form the alphabet of human thought.
2. Complex ideas proceed from these simple ideas by uniform and symmetrical combination, analogous to arithmetical multiplication.

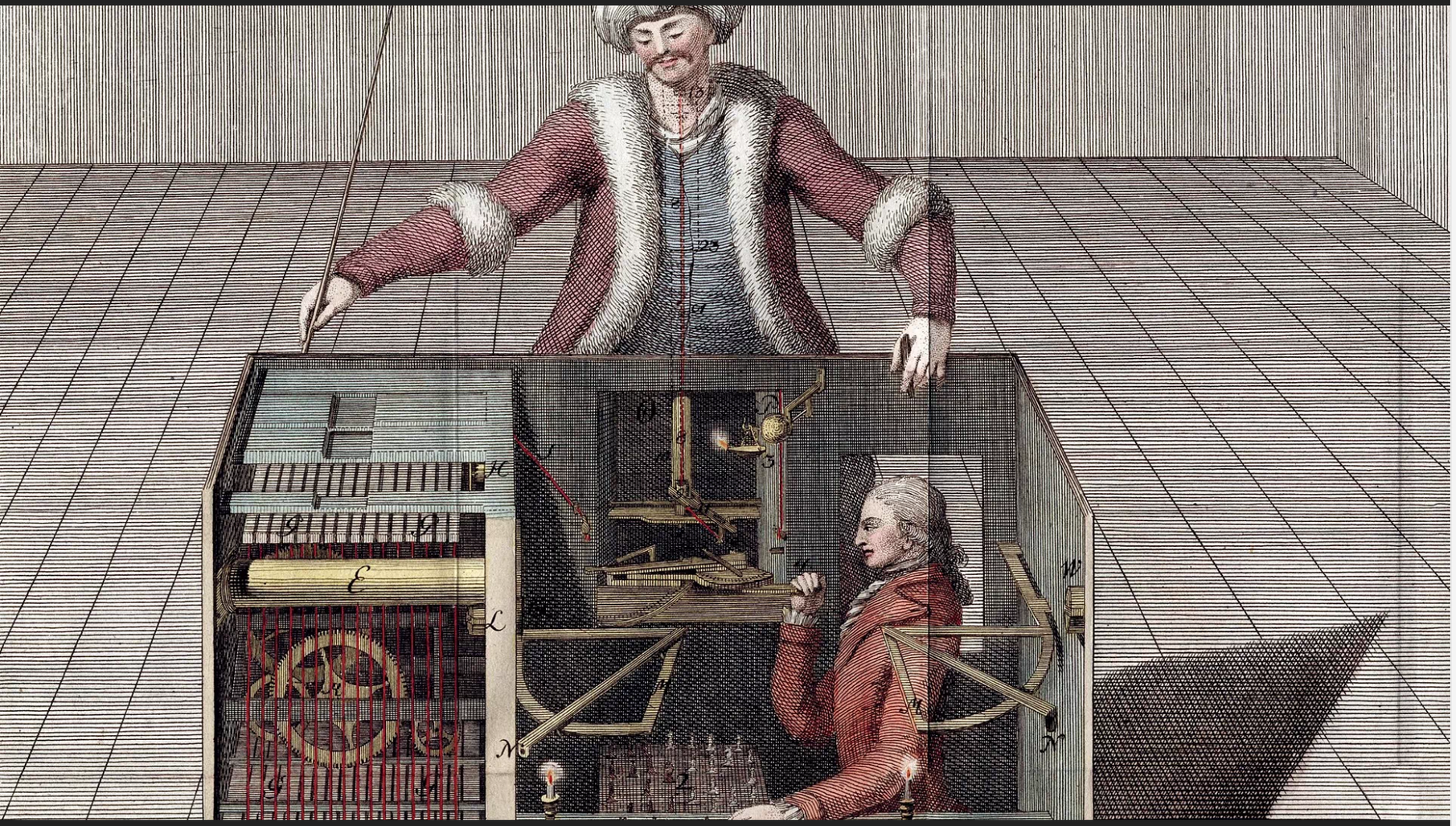
Leibniz wanted a universal and formal language able to express all mathematical, scientific, and metaphysical concepts. It could be used to prove the existence of God and the understanding of all things.

This viewpoint, partially inspired from Confucian philosophy and English sculptor Ann Conway, was to inspire numerous machine learning researchers.

Credits

- https://en.wikipedia.org/wiki/Gottfried_Wilhelm_Leibniz#Symbolic_thought
- https://en.wikipedia.org/wiki/Characteristica_universalis
- https://philosophy.stackexchange.com/questions/1801/what-is-it-that-leibniz-calls-a-monad
- https://en.wikipedia.org/wiki/Fazang
- https://stanford.library.usyd.edu.au/archives/sum2010/entries/conway/
- https://leibniz-bouvet.swarthmore.edu/
- https://en.wikipedia.org/wiki/De_Arte_Combinatoria
- https://commons.wikimedia.org/wiki/File:Gottfried_Wilhelm_Leibniz_c1700.jpg

1769 Early Automatons



Complex mechanical devices have a long history, the earliest known analog computer being the *Antikythera mechanism* now over 2000 thousand years old. There are even old accounts from China of a realistic human-shaped automaton that could walk and sing.

Mechanical marvels pop up repeatedly in many cultures, but one that I'd like to highlight is actually a fraud: Wolfgang von Kempelen's 1769 chess playing Turk automaton which secretly incorporated a human inside to control the machine. 245 years later a new online "crowdworking" business was created that that hired workers to perform online tasks that were often used to compensate for machine failings or generate machine training data - Amazon's Mechanical Turk service.

Credits

- https://en.wikipedia.org/wiki/Mechanical_Turk#/media/File:Racknitz_-_The_Turk_3.jpg
- https://en.wikipedia.org/wiki/Amazon_Mechanical_Turk

1804 Jacquard Loom

Joseph Marie Jacquard

In 1804 Joseph Marie Jacquard invented the Jacquard machine, a punch card controlled device fitted to a loom that simplified the manufacturing of complex textile patterns. This demonstrated the utility of programmable machinery, inspiring Charles Babbage to image a mechanical computer. At the time computing was done entirely by "unskilled" human labour, who were taught only addition and subtraction, and followed recipes to calculate tables of logarithms and such, which Babbage complained were full of mistakes.

Credits

- modern Jaquard Loom at Busatti Mill

1818

Frankenstein

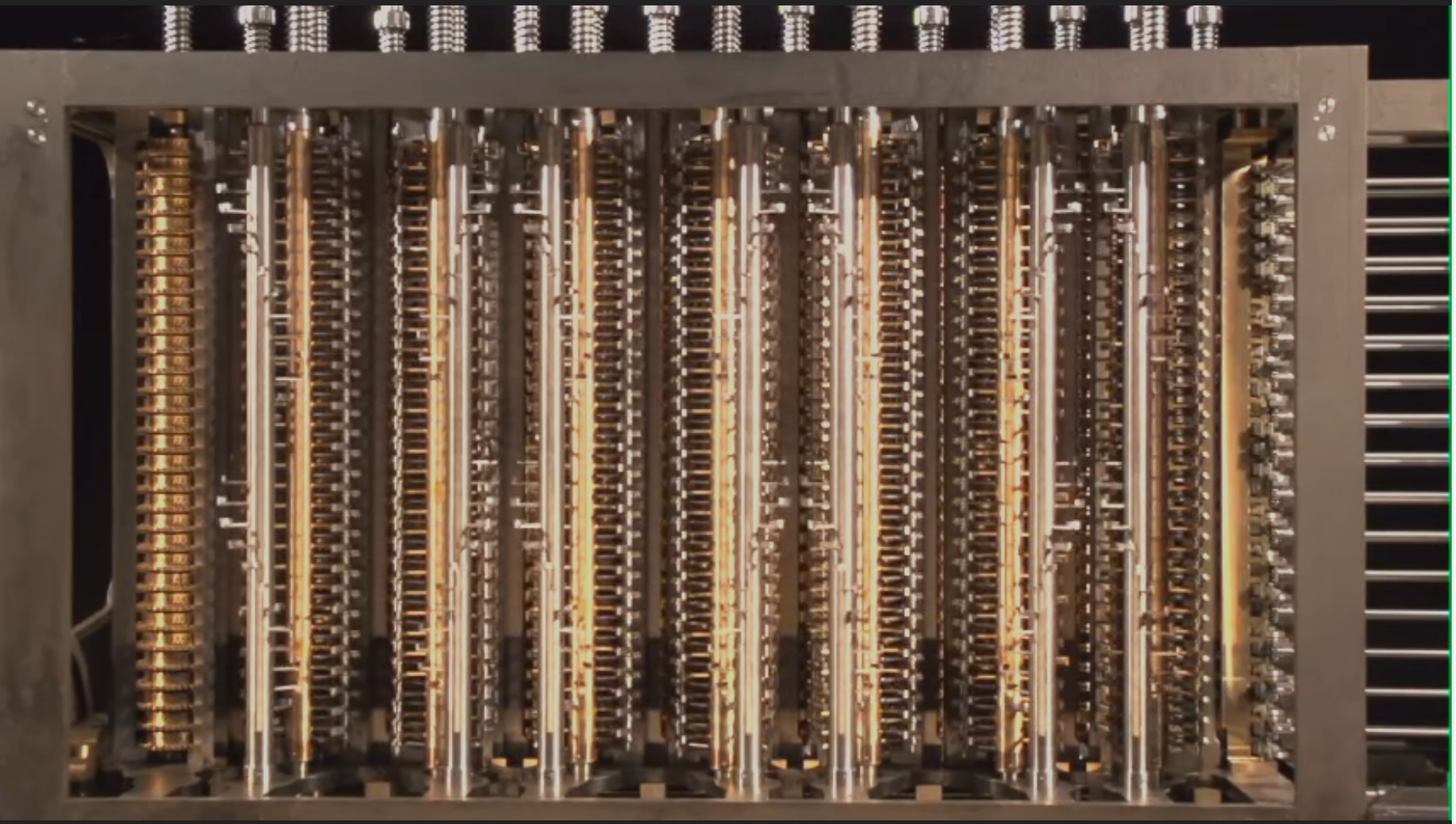
Mary Shelley

Mary Shelley was an English novelist who wrote Frankenstein. Victor Frankenstein's failure as a "parent" in the novel has been read as an expression of the anxieties which accompany pregnancy, giving birth, and particularly maternity. She feared that the irresponsible exercise of power would lead to chaos. For example, the creature in Frankenstein, reads books associated with radical ideals but the education he gains from them is ultimately useless.

Credits

- https://en.wikipedia.org/wiki/Mary_Shelley

1822 The Difference Engine



Charles Babbage

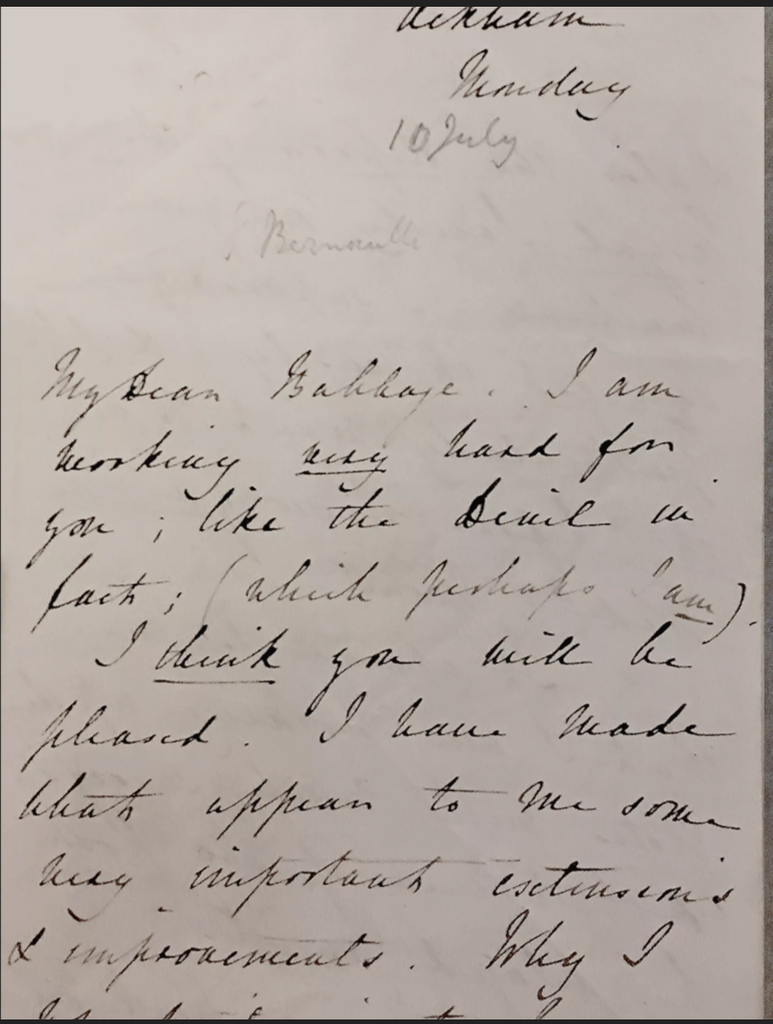
In 1822 Charles Babbage began to build *The Difference Engine* which could compute polynomial functions without using multiplication or division. Despite being well funded, Babbage never finished. He then designed a more complex machine called The Analytical Engine, programmed with punch cards and could do general purpose calculation. It was never built but would have been the first mechanical device to be a universal Turing machine.

Credits

- The Babbage Difference Engine #2 at CHM

1842

Poetical Science



Ockham
Monday
10 July

J. Bernoulli

My Dear Babbage. I am
working very hard for
you; like the Devil in
fact; (which perhaps I am).
I think you will be
pleased. I have made
what appear to me some
very important extensions
& improvements. Why I

Ada Lovelace

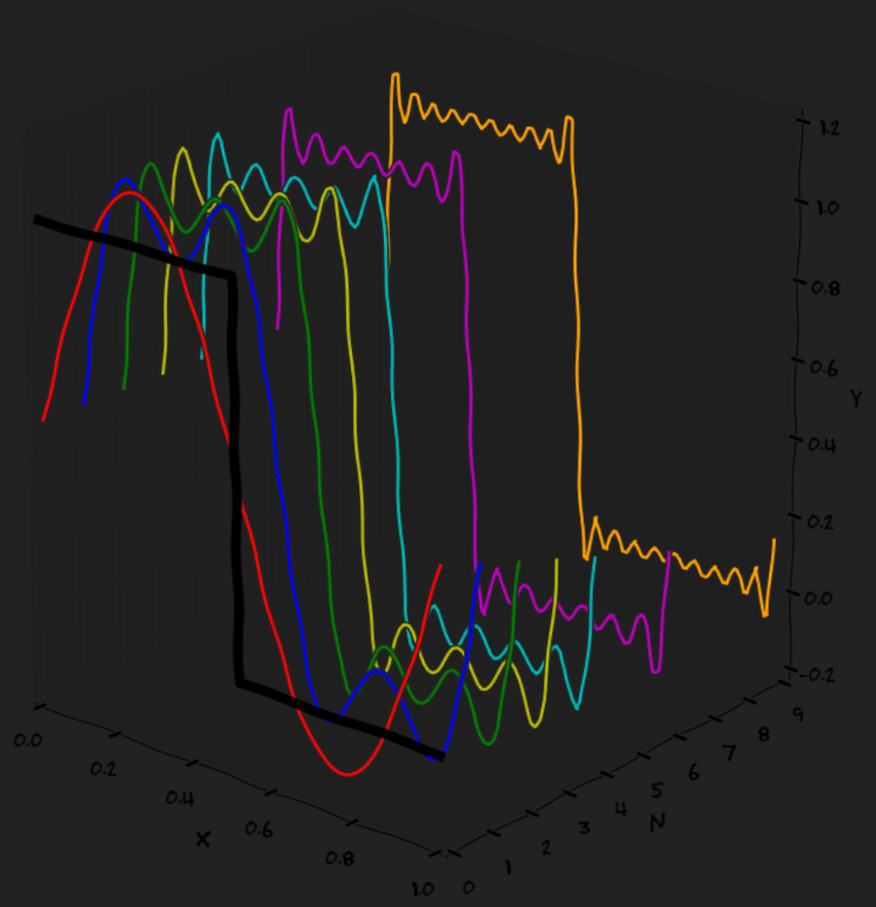
In 1842, English mathematician and writer Ada Lovelace was helping Charles Babbage with the first algorithms to be carried out by his Analytical Engine. Yet Lovelace saw opportunities beyond the math. She envisioned a computer where numbers could represent entities other than quantity. At the time it was revolutionary that computing could be used for creative pursuits. She called the idea Poetical Science.

Credits

- https://aiartists.org/ai-timeline-art
- https://en.wikipedia.org/wiki/Ada_Lovelace

1822

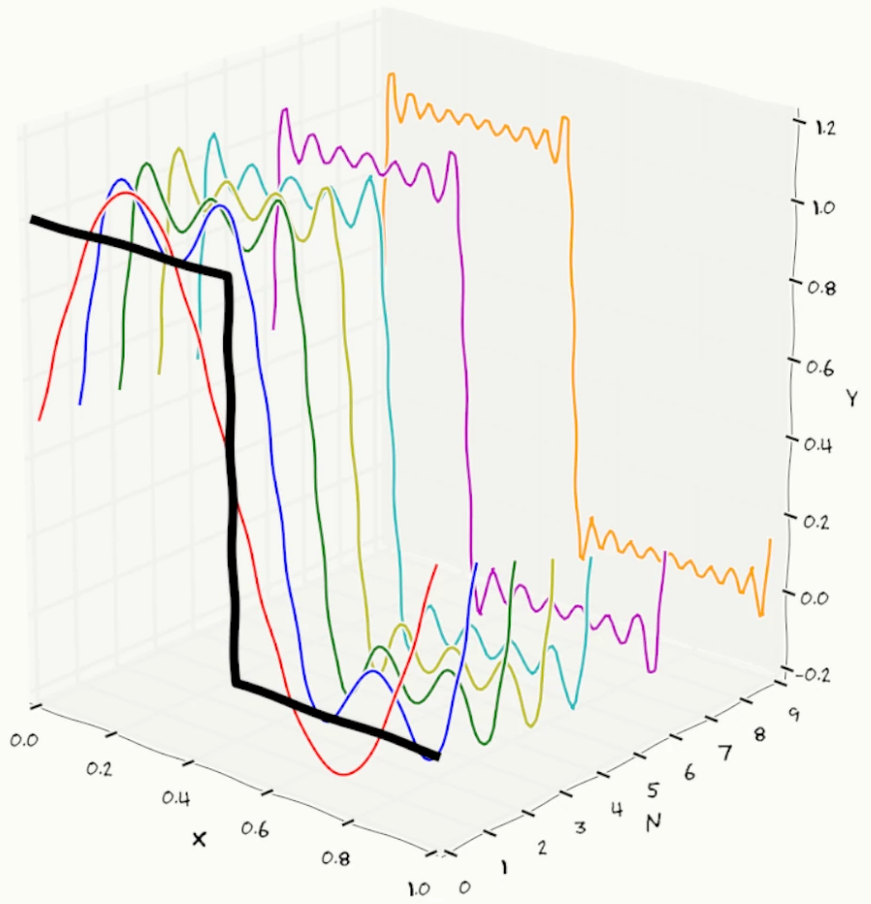
Fourier Transform



Joseph Fourier

Joseph Fourier was a french mathematician and physicist who barely survived the revolution *and* Napoleon. Among other subjects, Fourier studied mathematical waves, particularly sine waves, which trace the position along the circumference of a circle - basically an unfurled circle.

Fourier Transform



Joseph Fourier

A French mathematician and physicist who barely survived the revolution *and* Napolean. Among other subjects, Fourier studied mathematical waves, particularly sine waves, which trace the position along the circumference of a circle - basically an unfurled circle.

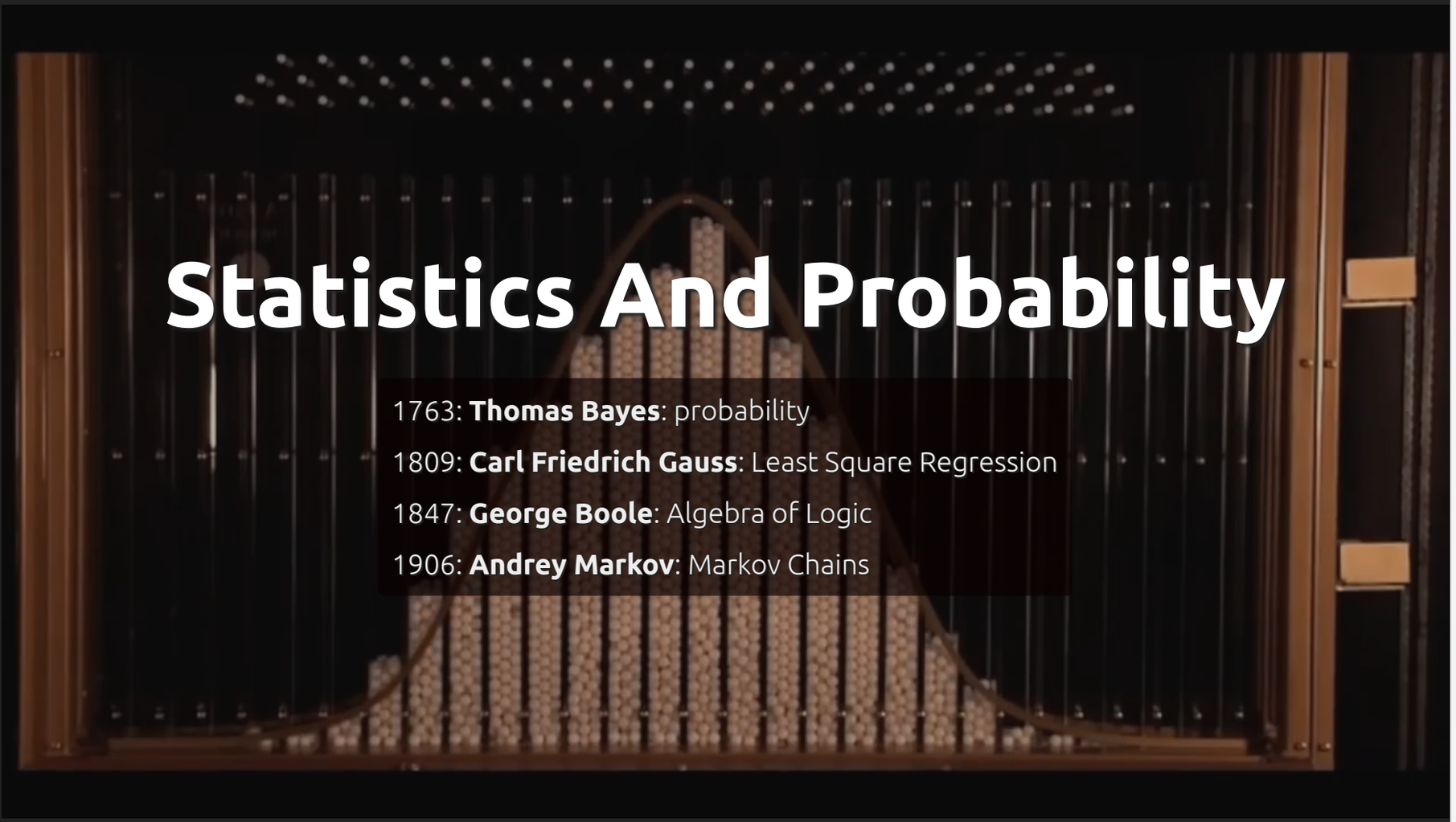
Waves have a frequency (how fast they wiggle or cycle) and an amplitude or energy (the height of the crests and troughs).

Fourier's idea was that a vast number of patterns of the world, including everything we can see or hear, can be described exactly as a sum of sine waves and nothing else. Any complex wave pattern can be decomposed into a set of simple sine waves - although it may be a very large set.

Fourier's insight underlies all conversion from analog to digital as well as data compression.

Credits

- A Biography of the Pixel by Alvy Ray Smith (2021)
- https://en.wikipedia.org/wiki/Fourier_transform
- But what is the Fourier Transform? A visual introduction. by 3Blue1Brown
- https://en.wikipedia.org/wiki/Joseph_Fourier
- https://www.oia.hokudai.ac.jp/isp/course-details/introductory-complex-function-introductory-fourier-analysis/



Machine learning and statistics are closely related, and thus rely on the same fundamental breakthroughs in probability. Statistics analyses data to get insight to the underlying distribution of probability. A probability distribution describes the frequency or count of the events or occurrences within a group or interval. In a coin flip, there are two possible outcomes and each has a 50% probability.

The statistical method of linear regression was developed for fitting a few imprecise observations of planetary movement to a mathematical function plotting the movement correctly. It is still a powerful and widely used method in machine learning today. Three basic assumptions were made:

1. Small errors are more likely than large errors;
2. Errors are balanced, just as likely to above or below the actual value; and
3. When several measurements are taken of the same quantity, the average is the most likely value.

1. Small errors are more likely than large errors;
2. Errors are balanced, just as likely to above or below the actual value; and
3. When several measurements are taken of the same quantity, the average is the most likely value.

As statistical techniques developed they were used to model more complicated phenomenon and required concepts like states and transitions.

Important contributions to number theory and logic were also critical for the development of computers and machine learning.

Credits

- https://aiartists.org/ai-timeline-art
- https://en.wikipedia.org/wiki/Machine_learning#Statistics
- https://en.wikipedia.org/wiki/Linear_regression
- https://www.actuaries.digital/2021/03/31/gauss-least-squares-and-the-missing-planet/
- https://medium.com/@kylecaron/introduction-to-linear-regression-part-1-implementation-in-python-with-statsmodels-7dbf24461072
- https://www.khanacademy.org/computing/computer-science/informationtheory/moderninfotheory/v/markov_chains

1763

Probability

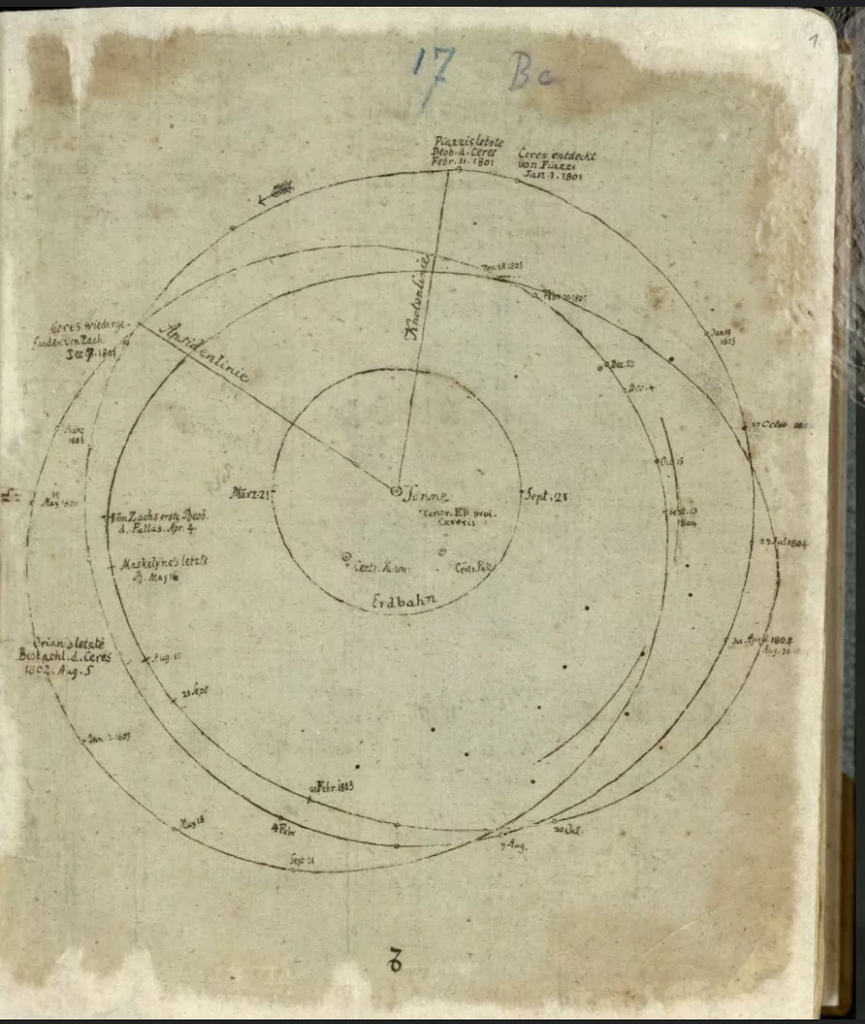
Thomas Bayes

Thomas Bayes was an English statistician, philosopher and Presbyterian minister who dramatically improved the framework for reasoning about the probability of events - allowing the probability to update as more information becomes available. Bayes never published what would become his most famous accomplishment; his notes were edited and published posthumously.

Credits

- https://en.wikipedia.org/wiki/Thomas_Bayes
- https://aiartists.org/ai-timeline-art

1809 Least Square Regression



Carl Friedrich Gauss

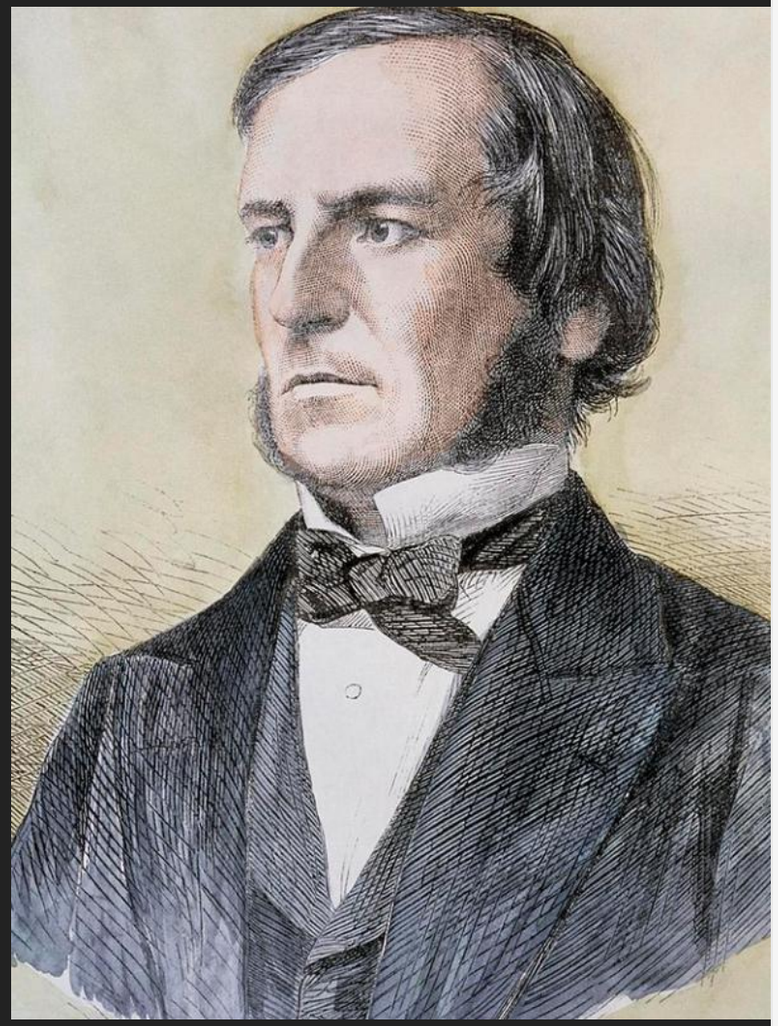
Carl Friedrich Gauss was a German mathematician and physicist ranked among history's most influential. Among numerous other contributions, he created a statistical technique that used 19 observations (less than 1% of the orbit) of the planetoid Ceres to help relocate it after it disappeared in the glare of the Sun.

Credits

- https://en.wikipedia.org/wiki/Carl_Friedrich_Gauss
- https://en.wikipedia.org/wiki/Ceres_(dwarf_planet)

1847

Algebra Of Logic



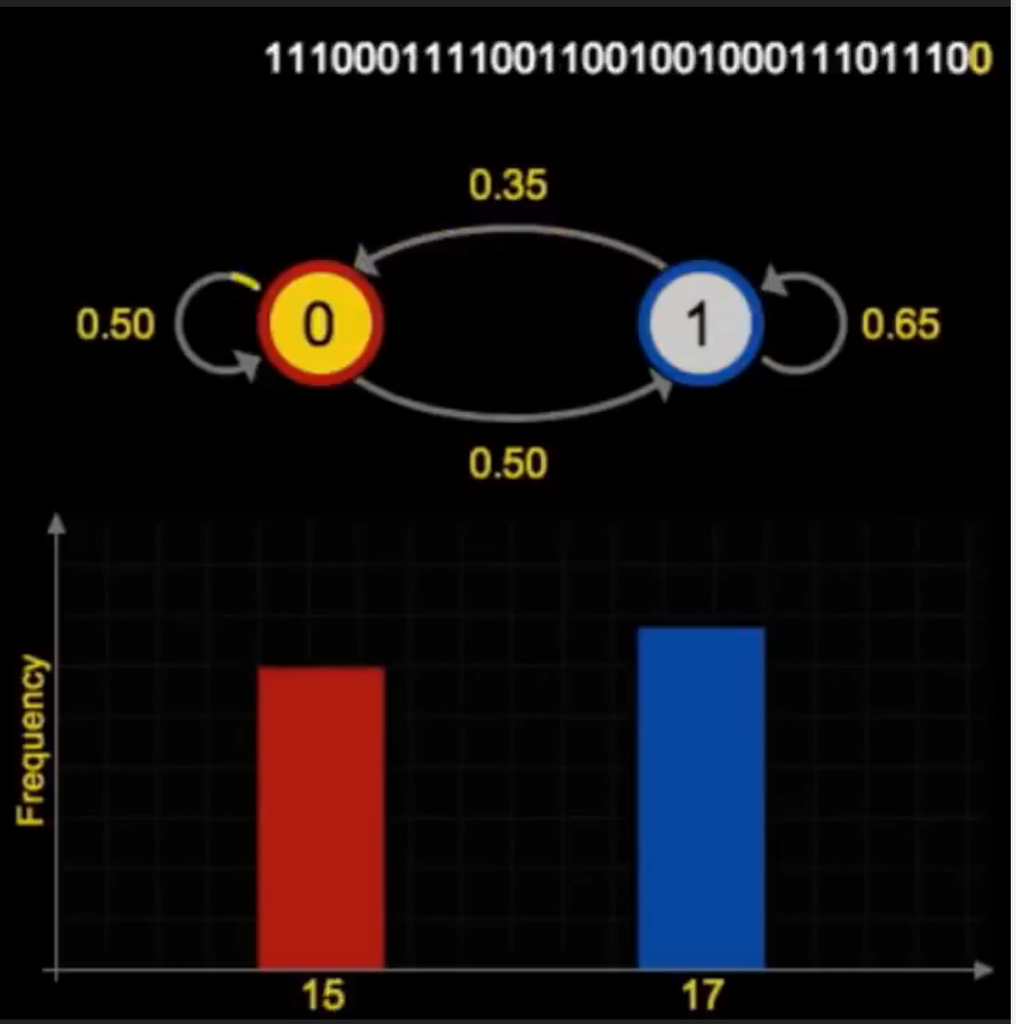
George Boole

George Boole publishes *Mathematical Analysis of Logic and An Investigation of the Laws of Thought on Which are Founded the Mathematical Theories of Logic and Probabilities* and introduces Boolean logic or “algebra of logic”. He showed that logical operations could be represented using mathematical symbols, and his work laid the foundation for the development of computers.

Credits

- https://en.wikipedia.org/wiki/George_Boole
- https://en.wikipedia.org/wiki/George_Boole#/media/File:George_Boole_color.jpg

1906 Markov Chains



Andrey Markov

Andrey Markov was a Russian mathematician best known for his work on stochastic processes. A primary subject of his research later became known as Markov chains.

A Markov chain uses states and transitions to model a random (stochastic) process where each value/event appears to vary randomly but is dependent on the previous outcome. Thus it is capable of "remembering" one previous value, but values before that do not affect the next. Each state may be connected to itself or other states and have a probability of moving to the next state.

References

- https://en.wikipedia.org/wiki/Andrey_Markov
- https://en.wikipedia.org/wiki/Markov_chain
- https://en.wikipedia.org/wiki/Stochastic_process

Credits

- *Origin of Markov chains* - Brit Cruise

Building on the developments in statistics and probability theory the general theories of information, communication, computation and games began to emerge.

Quantifying the amount of information in a system requires the use of probabilities. Learning that an unlikely event has occurred is more informative than learning that a likely event has occurred. If I already know everything you can tell me, I can receive no information. "Tell me something I don't know" means I want to be uncertain what you are going to say. All the things you could say that I don't know constitute information.

This also means that rare events that are more uncertain or more surprising then require more information to represent than common events. I don't have to say much to tell you something you expected anyways, but telling you something unexpected or unfamiliar requires a longer explanation. Thus conveying expected information requires and transmits only little information, but it requires and transmits a lot of information to convey the unexpected.

This is measured by what is called information entropy, or how much information is available in a system. It is related to, but separate from, entropy in other fields and can be thought of as "how many yes/no guesses would it take?" For example to guess the outcome of a coin flip or a die roll or a word in Wordle.

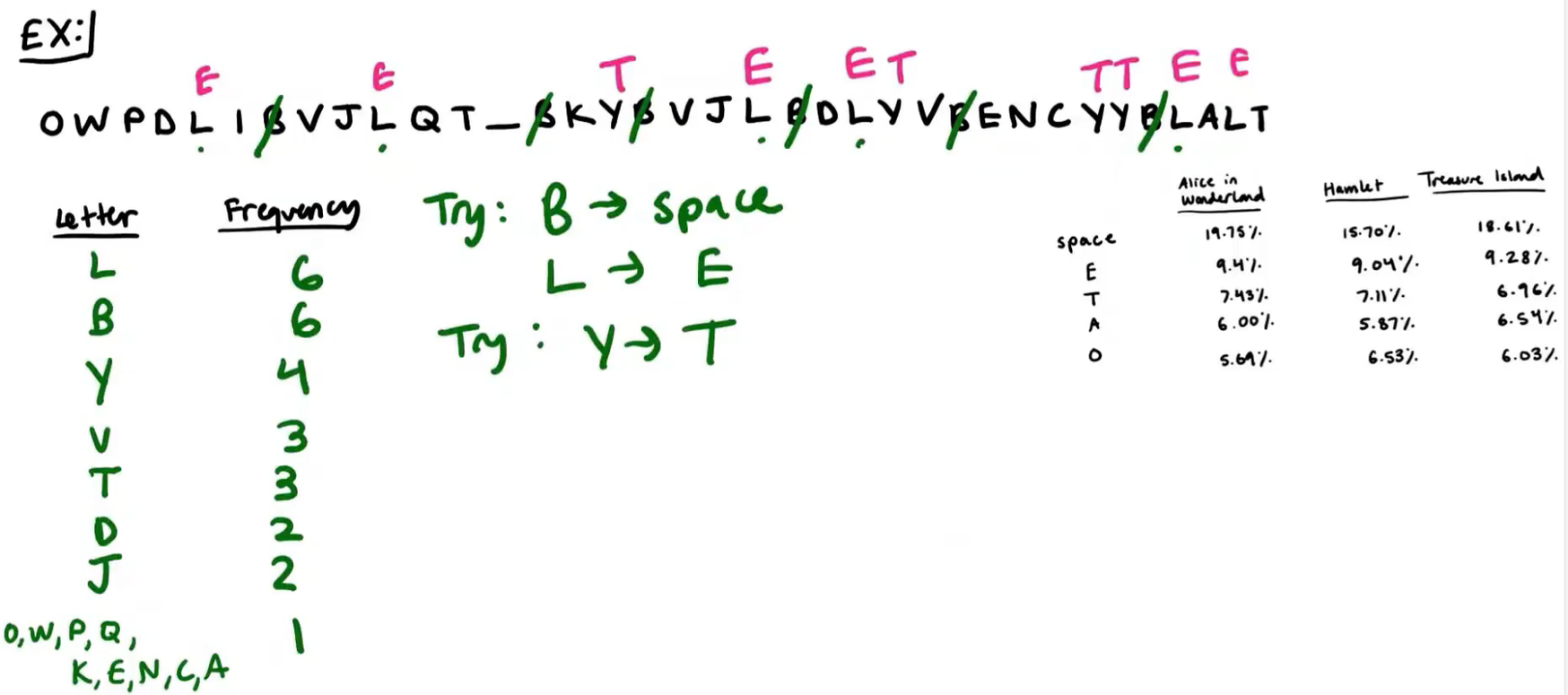
Credits

- Photo by Naser Tamimi

- Low Probability Event: High Information (surprising).
- High Probability Event: Low Information (unsurprising).
- Balanced Probability Distribution (surprising): High entropy.
- Skewed Probability Distribution (unsurprising): Low entropy.

Balanced Probability Distribution like a coin flip or die roll is surprising and high entropy. Each has an equal chance of being the secret answer, and the more options the higher the entropy.

Skewed probability distribution are less surprising with lower uncertainty and thus low entropy.



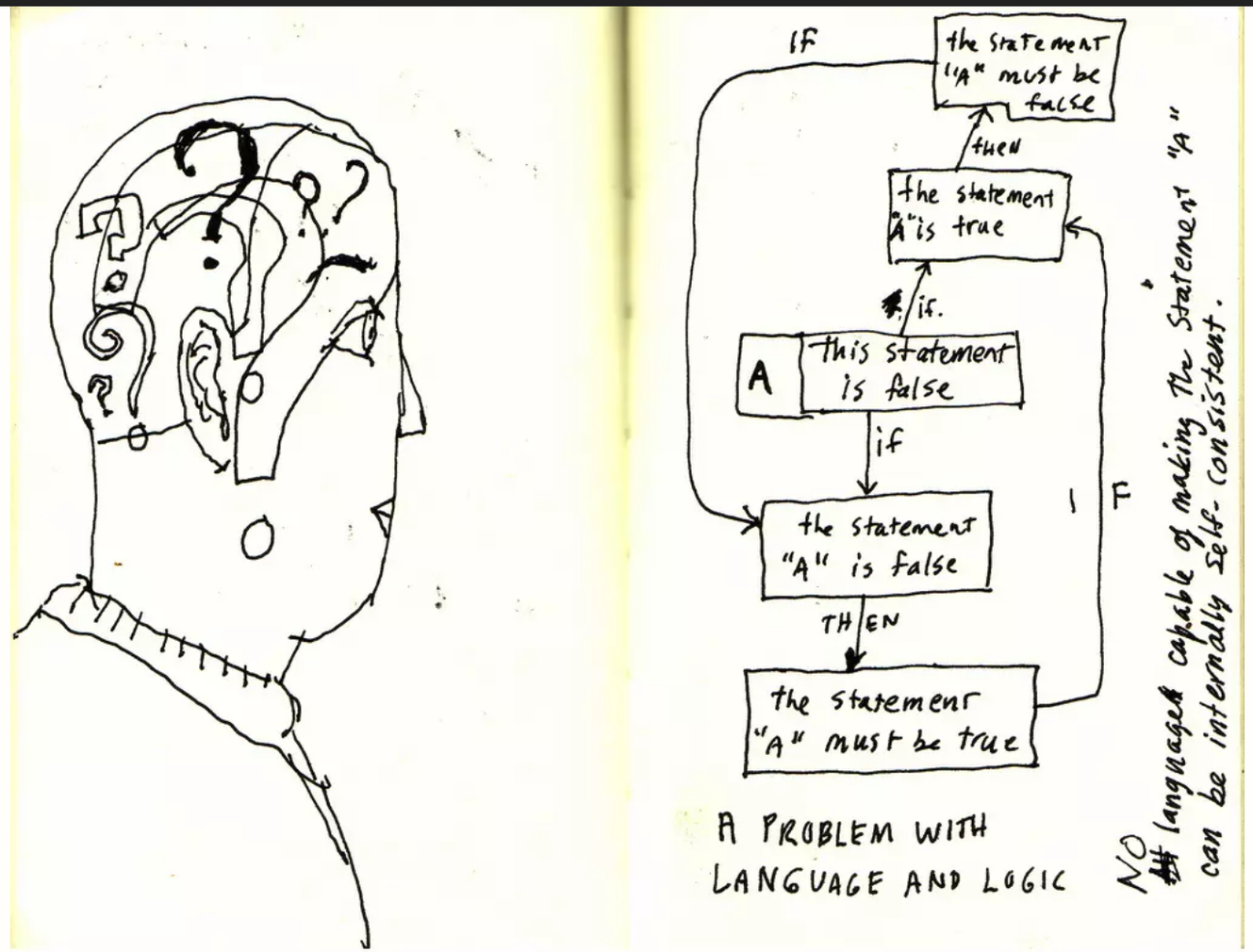
Take language, for example, where the letters and words have unequal frequencies. You can use information theory, as Alan Turing did in World War II, to decode encrypted messages. Even if all the letters have been switched or encoded as other letters the letter frequencies and probabilities of what words are in the messages and in what order can be used to guess the encoding (unless secret encoding is perfectly random). For example, messages may always start with a date.

Credits

- Cryptography: Frequency Analysis

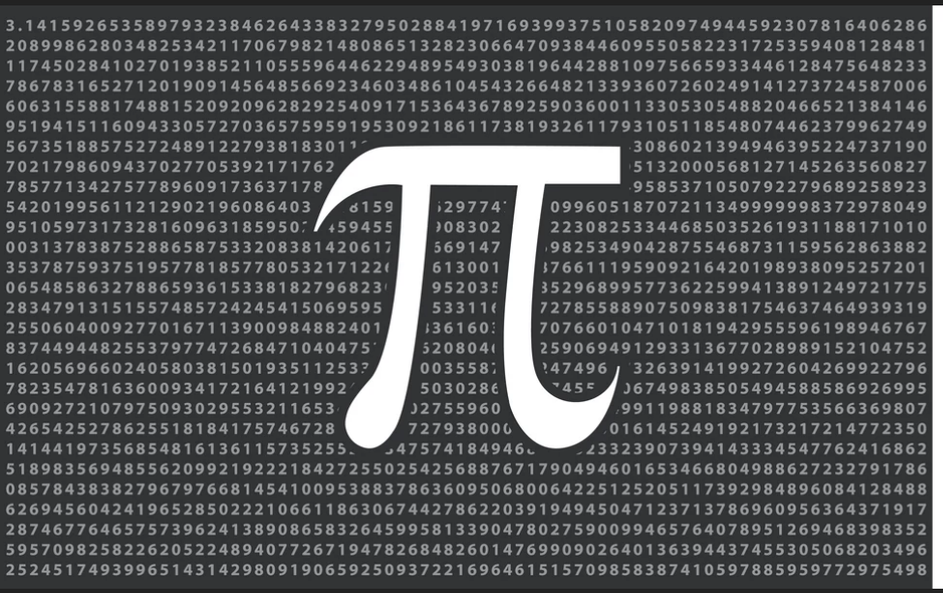
Another related concept is mutual information which measures the dependence between two variables. It tells you how much information can be obtained about one variable by observing the other. In other words, how likely it is that if one thing happens, the other thing will happen too. If two things are completely independent there is no mutual information.

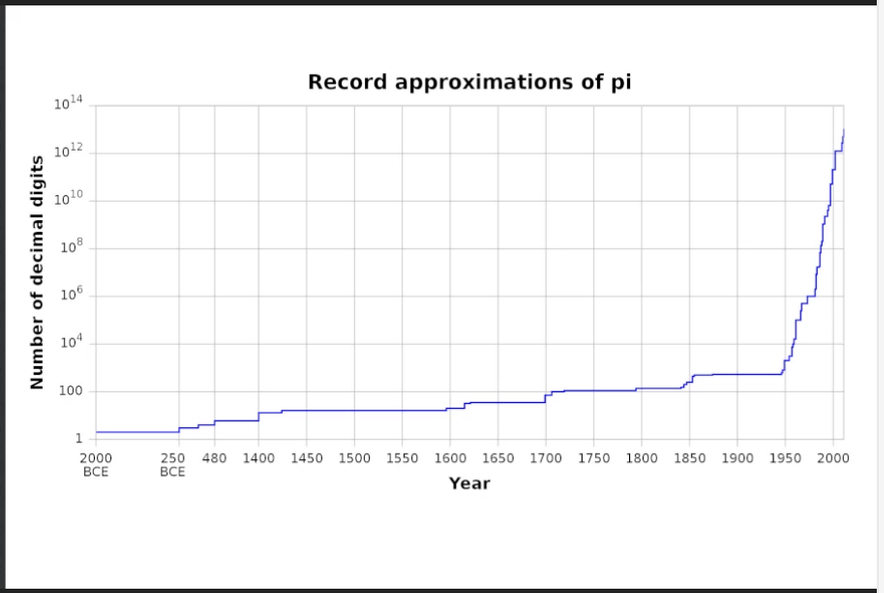
For example, once Turing broke the code, and could decrypt the German messages, he needed to hide the fact from the Germans, who would be looking for signs that their messages could be read. Thus any actions based on the information you gain from the messages must be carried out in a way that looks like it could have been random luck, rather than revealing that you know the secret information. You need to disguise the mutual information between your observed actions and the fact that you've broken the code - often by taking no action. Turing was haunted by all the people he couldn't save because it would have revealed that he had cracked the code.



The final key concept that was developed was that of undecidability. To begin to unpack that, let's look at a related example of the barber paradox: In a particular village the barber shaves everyone, and only those, who does not shave themselves. Who shaves the barber? The barber cannot shave himself as he only shaves those that do not shave themselves, and if he doesn't shave himself, then the barber shaves ...himself.

Problems that involve self-references can have serious theoretical problems about being decidable. In particular, self-reference and negation or "not"-ness is a paradoxical quagmire. It was shown that for certain problems there is no consistent, effective algorithm that can answer every question in the problem: it is undecidable. For centuries philosophers and mathematicians had been struggling to create a complete logical framework for all mathematics, but it was now proven an impossible task. So too was it impossible to create computer algorithms for these sorts of undecidable problems.





This ties into a further concept where something may not be predictable, but can be computed. You just have to run the program to see the output. To know the Nth digit of pi, you need to calculate it, there is no formula that can reach into the as-yet-computed future to reveal future digits.

- https://en.wikipedia.org/wiki/Information_theory
- https://machinelearningmastery.com/what-is-information-entropy/
- https://www.youtube.com/watch?v=lLWnd6-vSGo&list=PLzH6n4zXuckpIQPv8hiHpJkSyv0fmXEYr&index=3
- https://en.wikipedia.org/wiki/Undecidable_problem
- https://en.wikipedia.org/wiki/Barber_paradox

1924 Information Theory

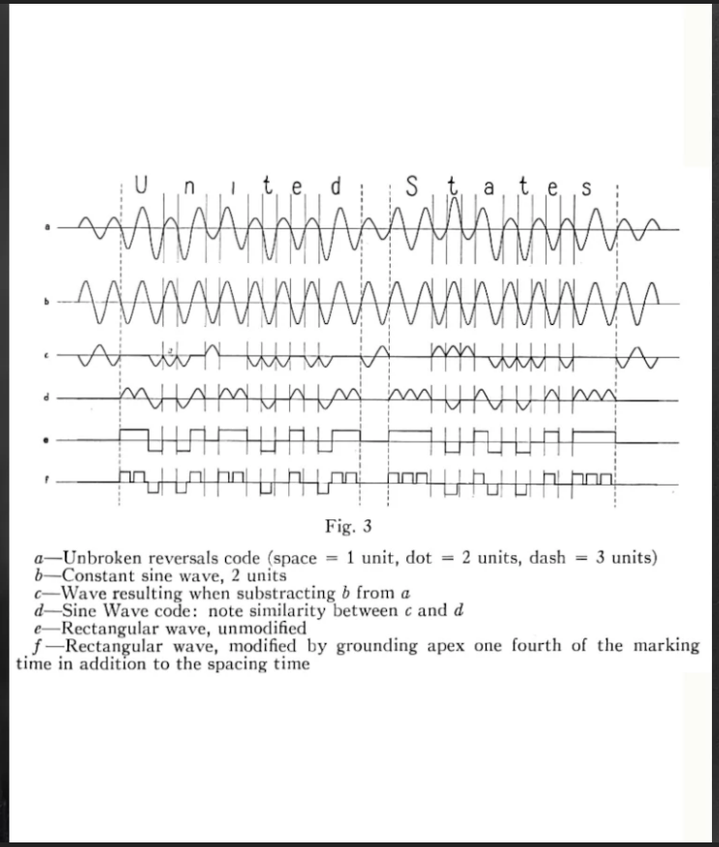


Fig. 3

a—Unbroken reversals code (space = 1 unit, dot = 2 units, dash = 3 units)
b—Constant sine wave, 2 units
c—Wave resulting when substracting *b* from *a*
d—Sine Wave code: note similarity between *c* and *d*
e—Rectangular wave, unmodified
f—Rectangular wave, modified by grounding apex one fourth of the marking time in addition to the spacing time

Harry Nyquist And Ralph Hartley

Harry Nyquist's 1924 paper, *Certain Factors Affecting Telegraph Speed*, contains a theoretical section quantifying "intelligence" and the "line speed" at which it can be transmitted by a communication system.

Ralph Hartley's 1928 paper, *Transmission of Information*, uses the word information as a measurable quantity, reflecting the receiver's ability to distinguish one sequence of symbols from any other.

1928

Game Theory

John Von Neumann

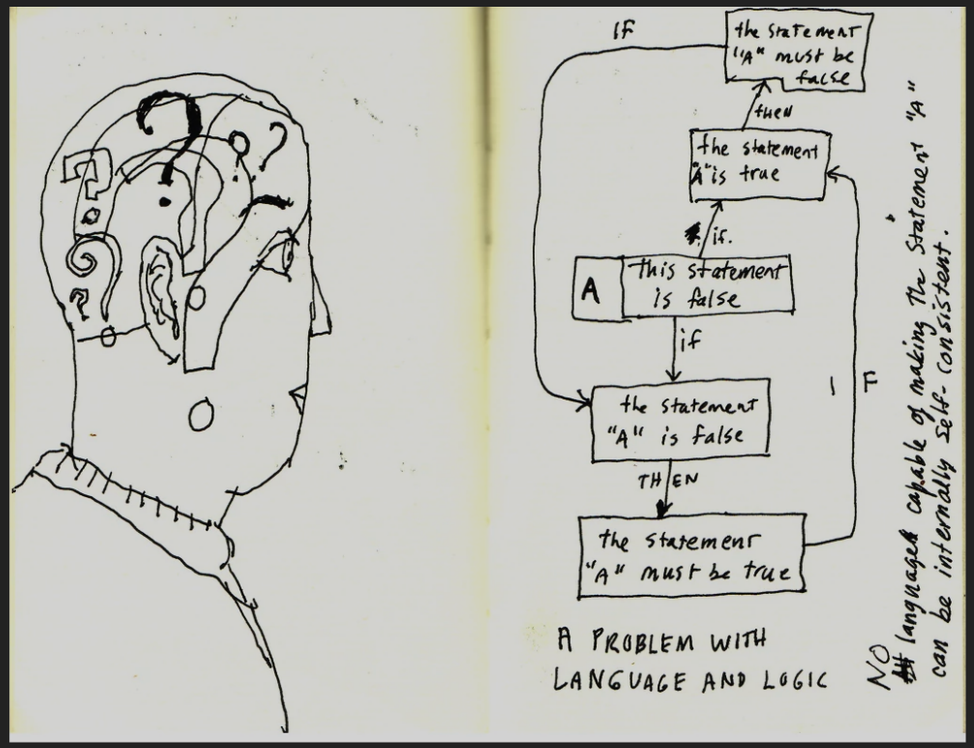
John von Neumann, a Hungarian-American polymath, made contributions to many fields, including being a key figure in cellular automata and the digital computer. His paper *On the Theory of Parlor Games*, one of thirty-two published before he was 26 years old, began the field of games theory and eventually his research would lead to revolutionizing the mathematics of economics.

The minimax decision rule he came up with is still used for minimizing the possible loss for the worst case (i.e. maximum loss) scenario.

Credits

- https://en.wikipedia.org/wiki/John_von_Neumann
- https://en.wikipedia.org/wiki/Minimax#Minimax_theorem

1931 Incompleteness Theorem



Kurt Gödel

Kurt Friedrich Gödel was a logician, mathematician, and philosopher. He developed the incompleteness theorems that created a paradox in formal systems: there will always be at least one true but unprovable statement.

1933

Sampling Theorem

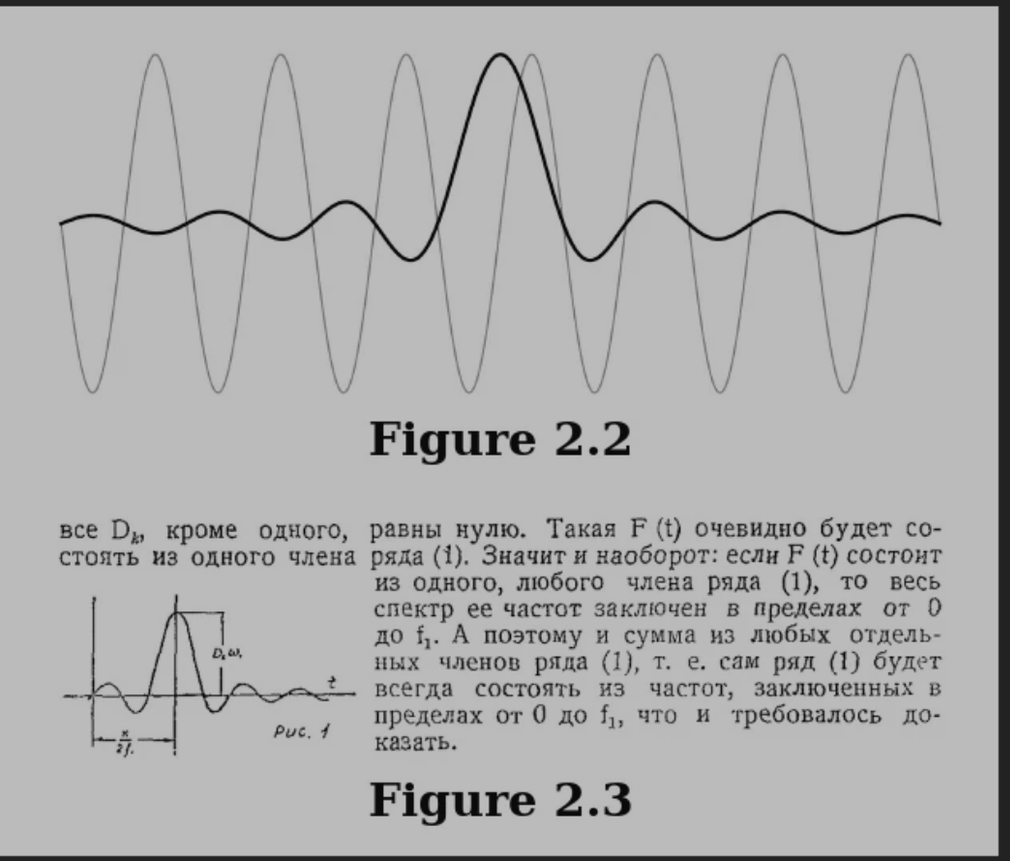


Figure 2.2

все D_k, кроме одного, равны нулю. Такая F (t) очевидно будет состоять из одного члена ряда (1). Значит и наоборот: если F (t) состоит из одного, любого члена ряда (1), то весь спектр ее частот заключен в пределах от 0 до f_1. А поэтому и сумма из любых отдельных членов ряда (1), т. е. сам ряд (1) будет всегда состоять из частот, заключенных в пределах от 0 до f_1, что и требовалось доказать.

Рис. 1

Figure 2.3

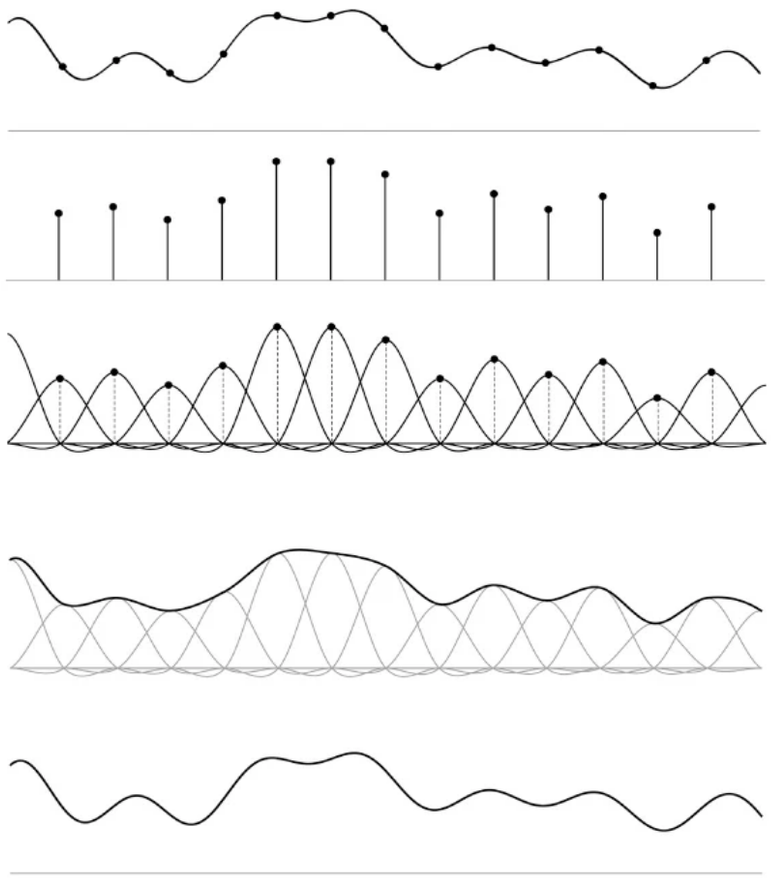
Vladimir Kotelnikov

Vladimir Kotelnikov (Владимир Котельников), an information theory and radar astronomy pioneer from the Soviet Union discovered the sampling theorem that allows digital samples to precisely reconstruct analog waves as long as the sampling rate is twice the highest wave frequency.

This forms the basis of digital encoding of audio, music, and light (i.e. pixels) and digital compression in general.

Credits

- A Biography of the Pixel by Alvy Ray Smith (2021)



I think it will be helpful to take a moment and talk about just how important the combination of Fourier and Kotelnikov's ideas are and what they say about the nature of digital and analog.

As we discussed in Part 1, an analog signal can be thought of as a wave. A digital representation of a wave takes samples along the wave at particular moments in time. Between each digital sample there are no other values. But there are actually infinite values on the continuous wave between the two. Digital has nothing in between two neighbouring samples, analog has infinite values between any two values.

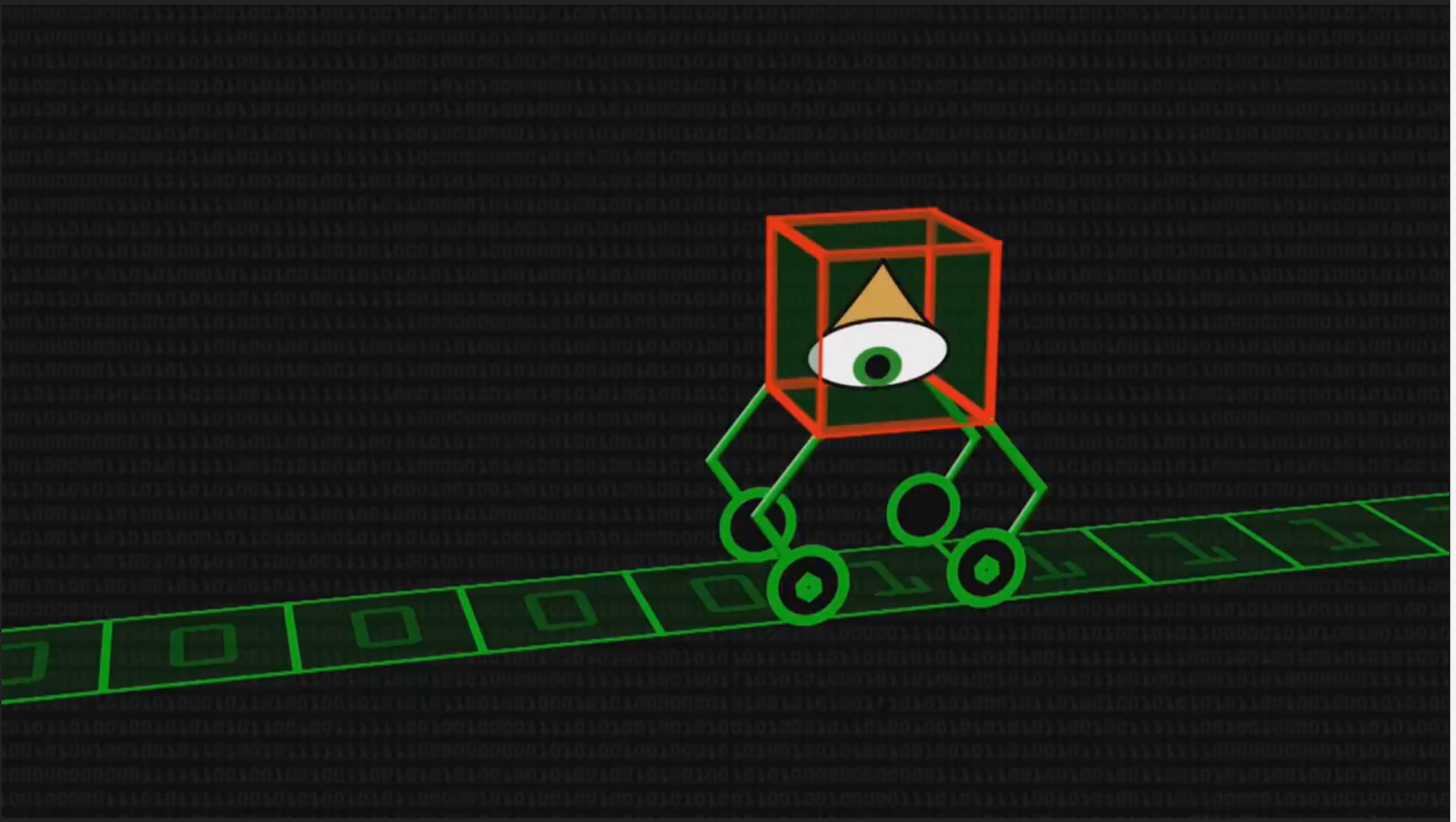
However, using the sampling theorem it is possible to reconstruct the infinite values in the wave perfectly, essentially by multiplying each digital value by another analog wave and then adding all the waves together. Fourier shows us that waves can be decomposed and recomposed and if the digital sampling is frequent enough then you can get back the original wave. Don't worry about the details, the important point made by Alvy Ray Smith in his book, *A Biography of the Pixel* is that:

Digital is not somehow less than analog. Taking samples seems to imply that the infinite amount of information between samples is lost — that digital is only an approximation — but it’s not so. The Sampling Theorem, if correctly applied, proves that digital discards nothing. Instead, it’s an extremely clever repackaging of infinity.

Credits

- A Biography of the Pixel by Alvy Ray Smith (2021)

1937 Universal Computing Machine



Alan Turing

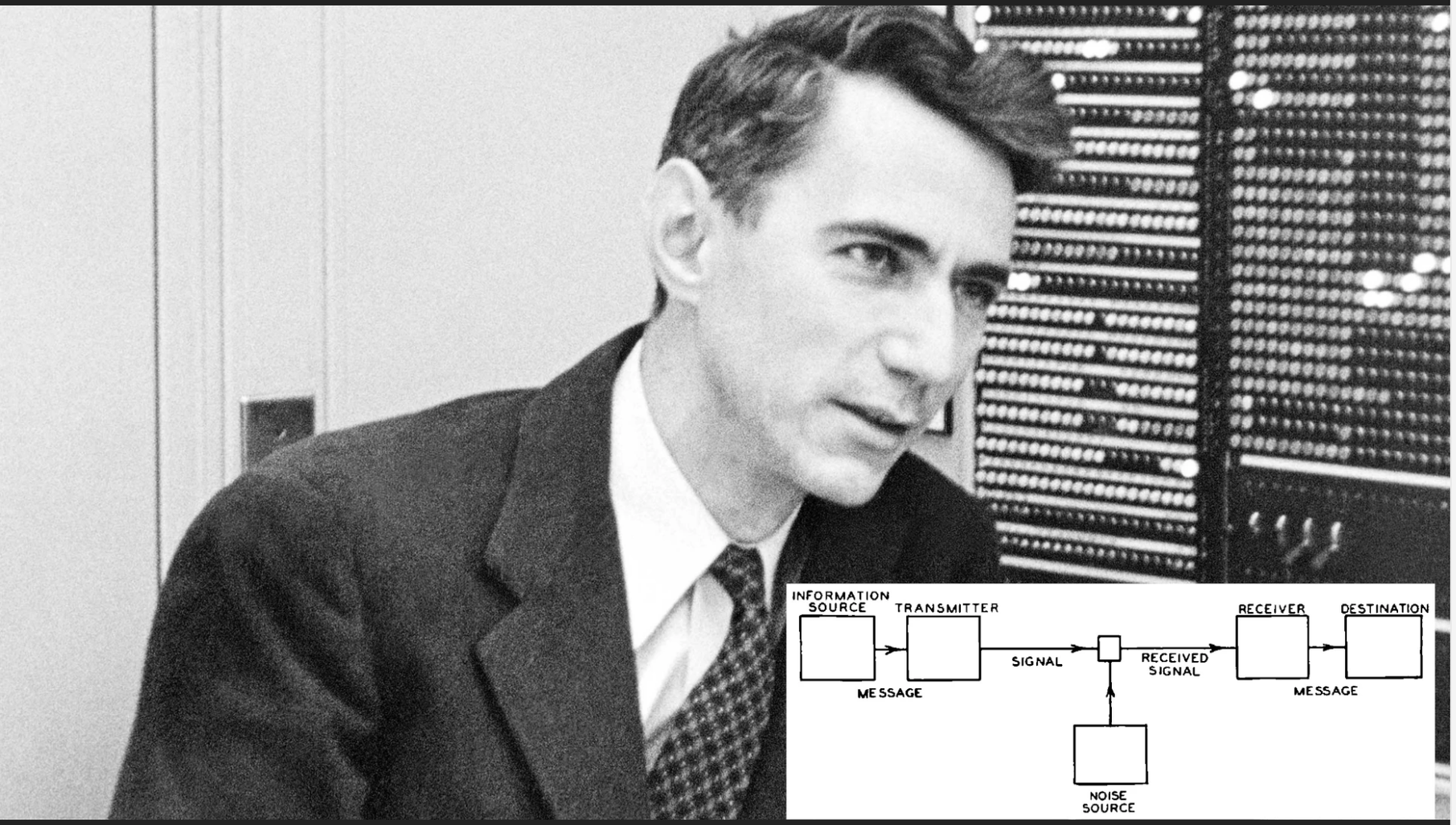
Alan Turing, an English mathematician and computer scientist, introduced the idea of the Universal Computing Machine. His thought experiment involved a simple machine that could read, write, and compare symbols on a length of paper or tape with no understanding of the symbols being read and written other than changing the "state" of the machine. Each state has a rule for how to read, write, compare, move the tape and change to another state.

Given an infinitely long tape Turing proved that this machine could compute anything.

Credits

- *Turing Machines Explained* - Computerphile

1948 Theory Of Communication



Claude Shannon

Claude Shannon, an American mathematician and engineer, his 1937 master's thesis demonstrated how Boolean algebra could be used to build digital computers.

His *A Mathematical Theory of Communication* article tackled how to best encode a message a sender wants to transmit. Using Norbert Weiner's probability tools he developed information entropy to measure the information content (based on the amount of uncertainty) of a message.

Credits

- https://en.wikipedia.org/wiki/Claude_Shannon
- https://www.quantamagazine.org/how-claude-shannons-information-theory-invented-the-future-20201222/

Credits

- *Machina Speculatrix* : William Grey Walter (1949)

Early Neural Networks And Cybernetics

1943: **Warren McCulloch** & **Walter Pitts**: artificial neuron model can model Boolean algebra

1950: **Alan Turing**: *Intelligent Machinery* paper introduces the *Imitation Game*

1948: **Norbert Wiener**: *Cybernetics: Or Control and Communication in the Animal and the Machine*

1949: **Donald Hebb**: Hebbian learning: neurons that fire together, wire together

1949: **William Grey Walter**: *Machina Speculatrix:* analog tortoise robots with two neuron brain

1951: **Marvin Minsky**: *SNARC* (Stochastic Neural Analog Reinforcement Calculator)

1952: **Friedrich Hayek**: *The Sensory Order:* connectionist theory of mind

1954: **Marvin Minsky**: *Neural Nets and the Brain Model Problem*

1956: **Nicolas Schöffer**: *CYSP 1* robot and first artwork to explicitly employ cybernetic principles

1964: **Roy Ascott**: *Behaviourist Art and the Cybernetic Vision:* first extensive theory of cybernetic art

1968: **Jasia Reichardt**: *Cybernetic Serendipity:* exhibition of computer and cybernetic art

Early work into neural models and evolutionary models of intelligence began before the first digital computers. William Grey Walter's tortoises were built with a light sensor, touch sensor, propulsion motor, steering motor, and a two vacuum tube analogue computer.

Robots, a term coined in 1920 to describe imagined artificial workers, like Walter's and Nicolas Schöffer's *CYSP 1* were being used to experiment with connectionist and cybernetic models of intelligence. These theories are precursors to artificial neural networks and involve connections and feedback loops as well as direct sensing of the environment as primary components of intelligence.

Norbert Weiner characterized cybernetics as concerned with "control and communication in the animal and the machine". The goal of control was often homeostasis or self-organization, or more generally, a resistance to change away from optimal conditions. When these optimal conditions lay on the borders of stasis and chaos, those unpredictable boundary areas where computation can form, like we found in the first tutorial in Rule 110, then life and intelligence can arise.

Donald Hebb put forth the unexpected and hugely influential idea that knowledge and learning occurs in the brain primarily through the formation and change of synapses between neurons - concisely stated as Hebb's Rule: "neurons that fire together, wire together".

Cybernetic thinking expanded to many different fields and into the art world, with the first major exhibit about 20 years after Walter's first tortoises.

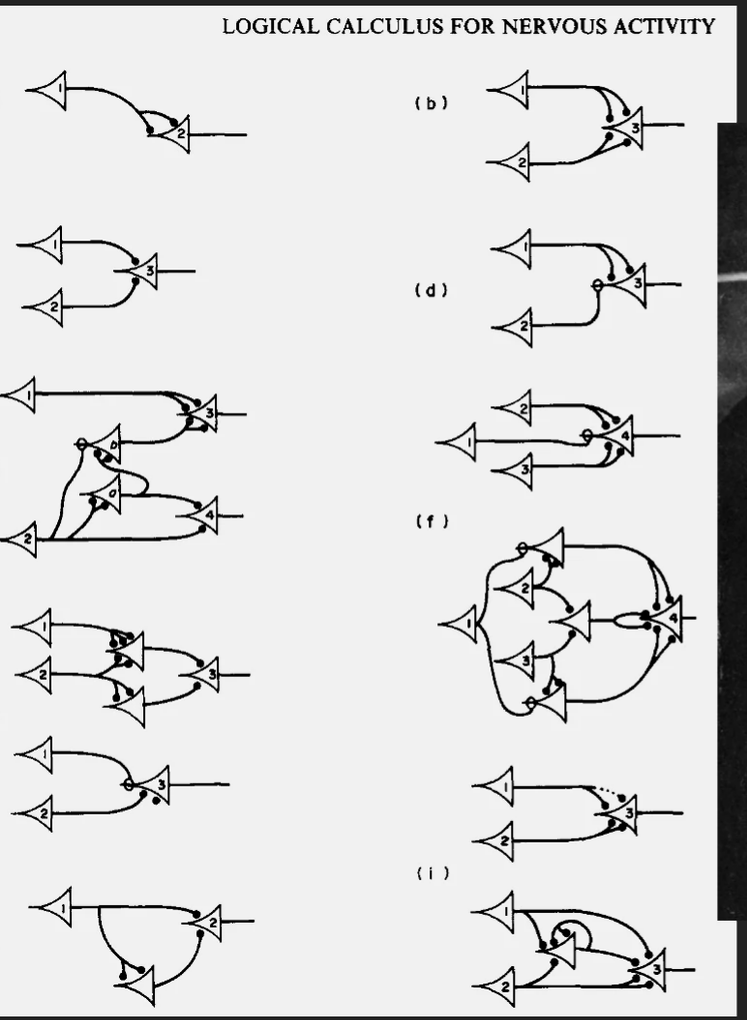
Credits

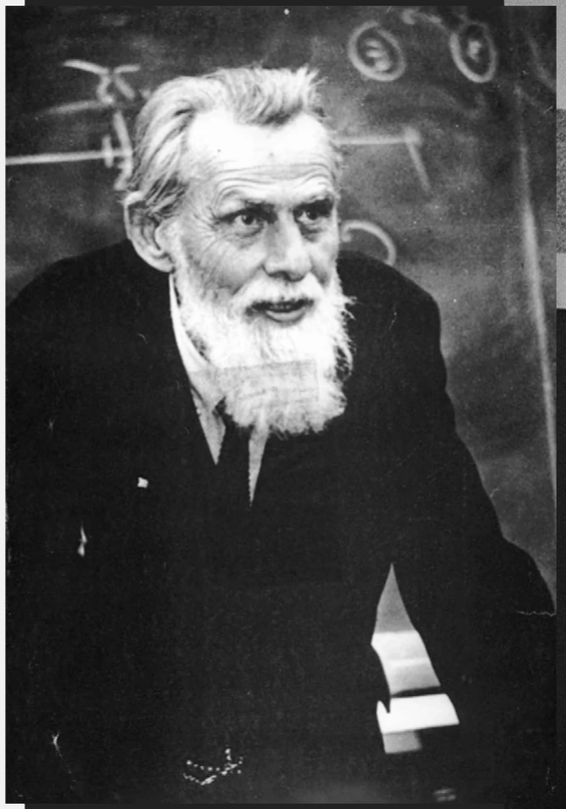
- *Cyspe* : Nicolas Schöffer (1956)

Credits

- https://en.wikipedia.org/wiki/Cybernetics
- https://en.wikipedia.org/wiki/Neural_network#History

1943 Neuron Model





Warren McCulloch & Walter Pitts

Although they were almost a generation apart and had dissimilar scientific backgrounds, neuropsychiatrist Warren McCulloch and mathematician Walter Pitts had similar intellectual concerns, simultaneously motivated by issues in philosophy, neurology, and mathematics. This lead to their landmark publication *Logical Calculus of Ideas Immanent in Nervous Activity* which described a neuron model that could perform Boolean algebra.

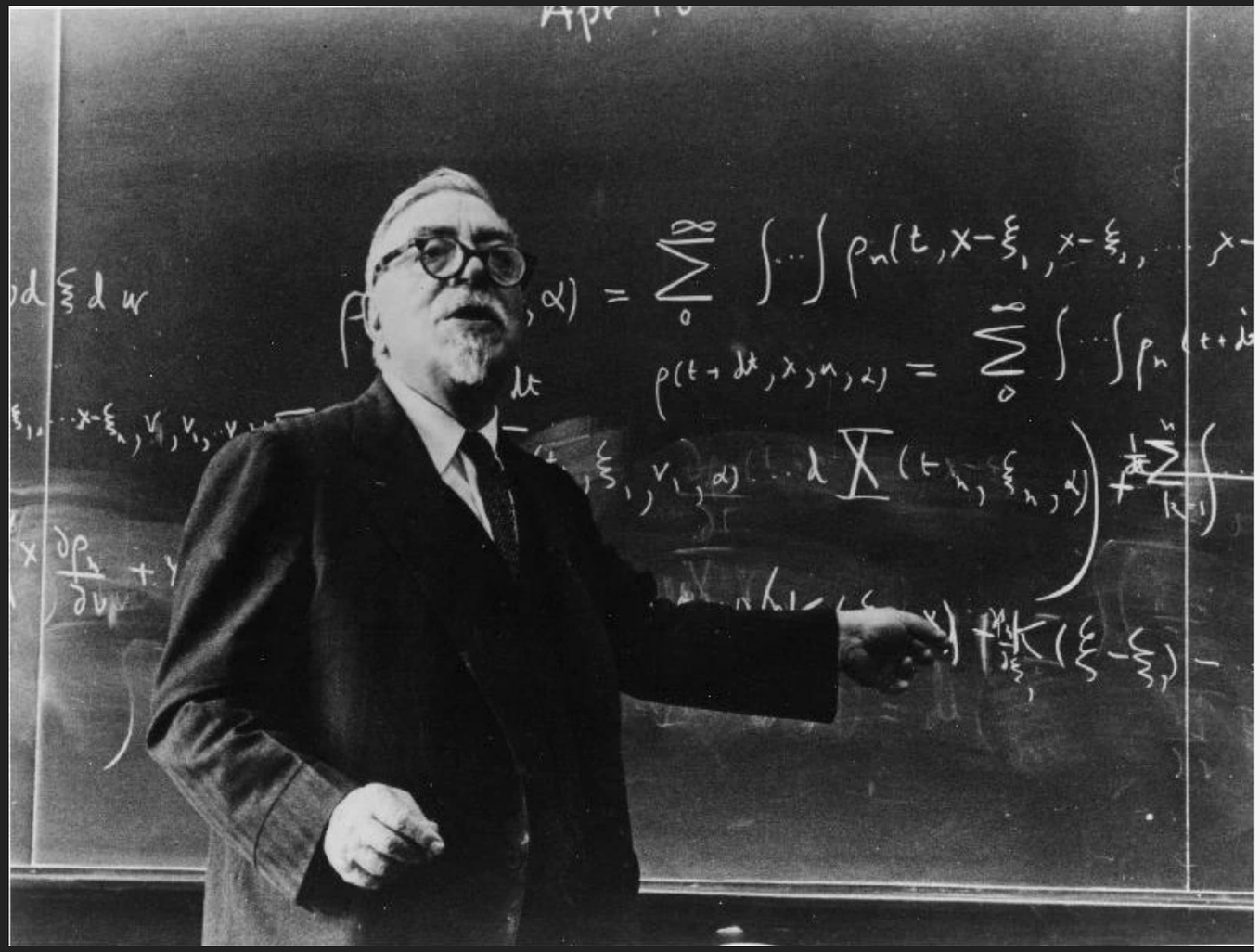
A neuron that sums binary inputs and then outputs a 1 if the sum exceeds a certain threshold value, and otherwise outputs a 0, can model the basic OR/AND/NOT functions.

Credits

- https://onlinelibrary.wiley.com/doi/epdf/10.1002/jhbs.1094

1948

Cybernetics



Norbert Wiener

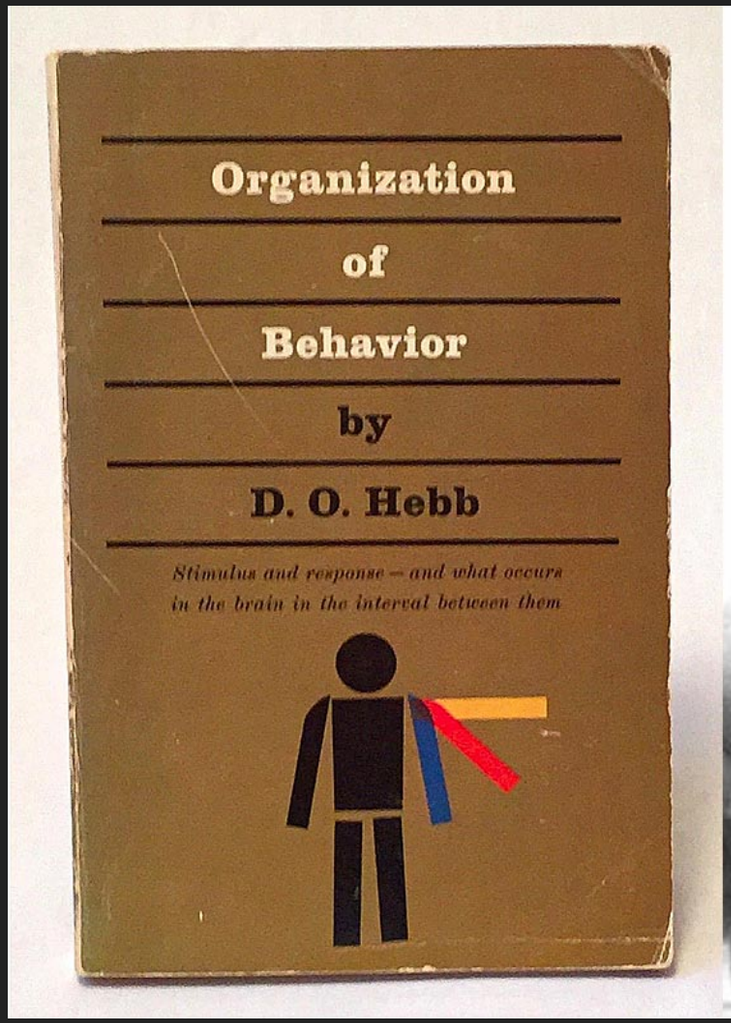
Norbert Wiener was an American mathematician and philosopher. His *Cybernetics: Or Control and Communication in the Animal and the Machine* was the beginning cybernetics.

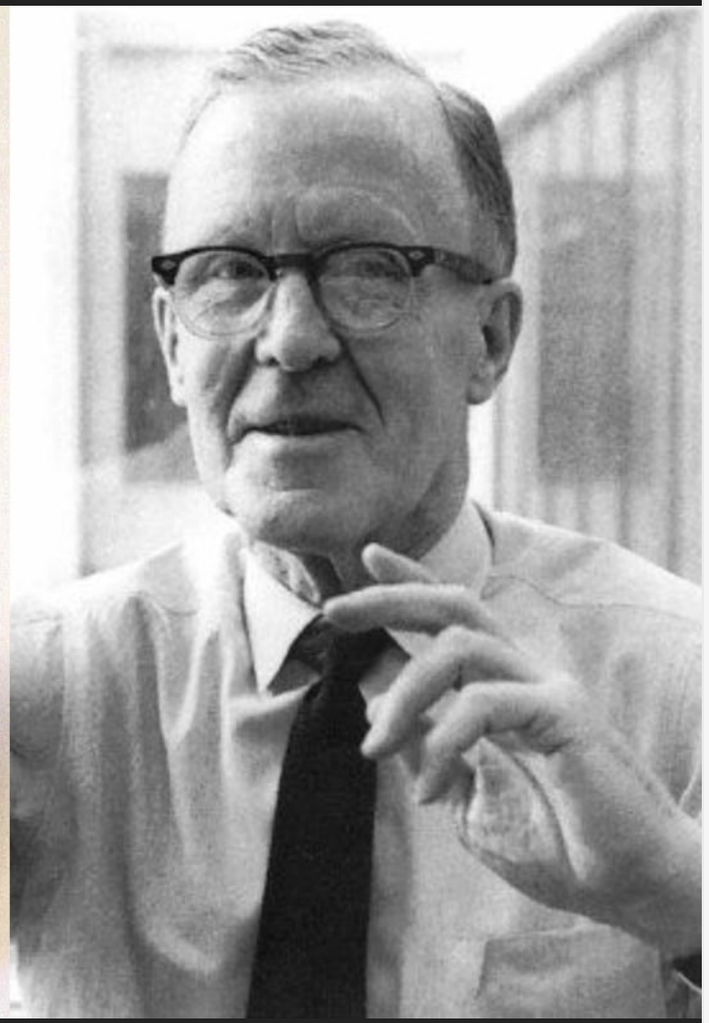
The core concept is circular causality or feedback where the observed outcomes of actions are taken as inputs for further action, generally towards self-organization or homeostasis.

Credits

- https://www.privatdozent.co/p/the-absent-minded-father-of-cybernetics-db9

1949 Hebbian Learning





Donald Hebb

Donald Hebb was a Canadian psychologist who sought to understand the process of learning. In *The Organization of Behavior* he introduced Hebbian learning, paraphrased as, "neurons that fire together wire together".

Credits

- https://labelyourdata.com/articles/history-of-machine-learning-how-did-it-all-start

1949 *Machina Speculatrix*

William Grey Walter

William Grey Walter was an American-born British neurophysiologist, cybernetician and robotician.

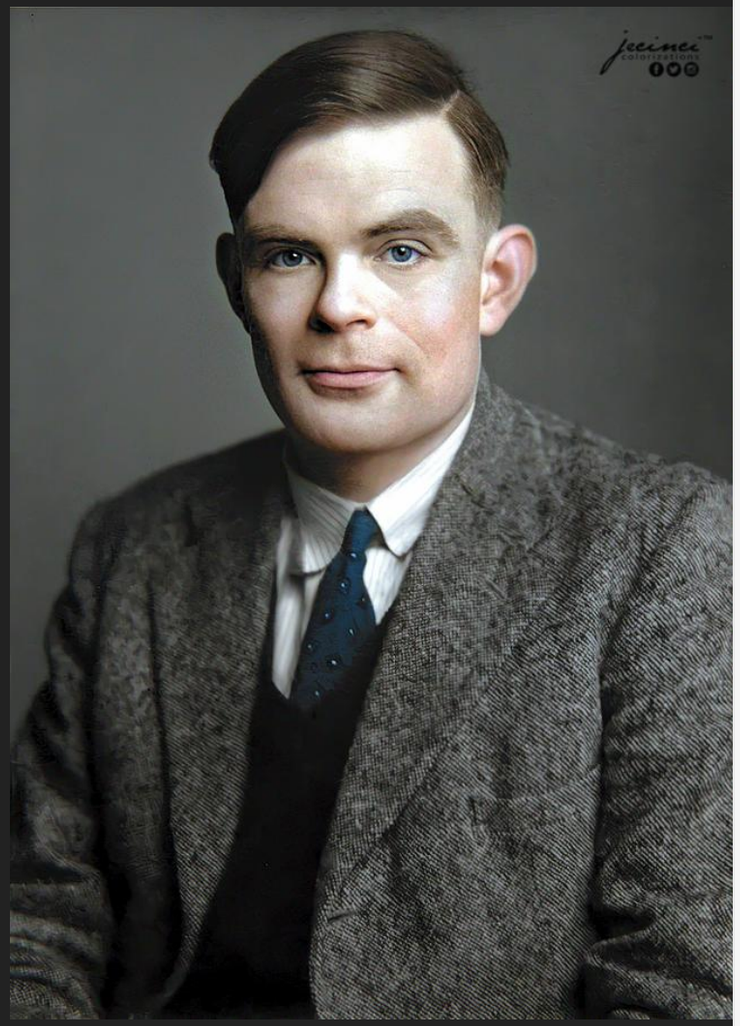
Walter's *Machina Speculatrix* tortoises were built with a light sensor, touch sensor, propulsion motor, steering motor, and a two vacuum tube analogue computer.

Credits

- *Machina Speculatrix* : William Grey Walter

1950

Imitation Game



Alan Turing

In 1950 Alan Turing devised a famous test called the Imitation Game, now called the Turing Test, a three-person game in which an interrogator asks questions of a man and a woman in another room in order to determine the correct sex of the two players, who, in turn, are trying to fool the interrogator.

Turing believed the question ‘Can machines think?’ was ambiguous, and instead was interested in what happens if a machine replaces a human player in the game.

1951

SNARC

Marvin Minsky

Inspired by McCulloch and Pitts, Marvin Minsky, who was American computer and cognitive scientist, built the SNARC (Stochastic Neural Analog Reinforcement Calculator), a device with 40 randomly connected Hebb synapses made from vacuum tubes, which simulated a rat finding its way through a maze.

1952 Connectionist Theory Of Mind

CONTENTS

PREFACE Page v

INTRODUCTION xv

I. THE NATURE OF THE PROBLEM 1

1. *What is Mind?* 1.1–1.5
2. *The Phenomenal and the Physical World* 1.6–1.21
3. *Stimulus, Impulse, and the Theory of the Specific Energy of the Nerves* 1.22–1.37
4. *Differences in Quality are Differences in the Effects* 1.38–1.55
5. *The Unitary Character of the Sensory Order* 1.56–1.67
6. *The Order of Sensory Qualities not Confined to Conscious Experience* 1.68–1.74
7. *The Denial or Disregard of Our Problem by Behaviourism* 1.75–1.89
8. *The 'Absolute' Qualities of Sensations a Phantom-Problem* 1.90–1.107

II. AN OUTLINE OF THE THEORY 37

1. *The Principle of the Explanation* 2.1–2.19
2. *The Sensory Order in its Static and in its Dynamic Aspects* 2.20–2.31
3. *The Principle of Classification* 2.32–2.38
4. *Multiple Classification* 2.39–2.44
5. *The Central Thesis* 2.45–2.51

III. THE NERVOUS SYSTEM AS AN INSTRUMENT OF CLASSIFICATION 55

1. *An Inventory of the Physiological Data* 3.1–3.14
2. *Simplifying Assumptions on which the Operation of the Principle will be Discussed* 3.15–3.24
3. *Elementary Forms of Classification* 3.25–3.34
4. *Complex Forms of Classification* 3.35–3.51

Friedrich Hayek

Friedrich Hayek was an Austrian-British economist and philosopher best known for his contributions to economic theory. His book *The Sensory Order* was an early connectionist theory of mind that challenged behaviorisms focus on stimuli and response, suggesting instead that principles by which thinking operated should be studied and that a reductionist approach was insufficient. For Hayek interpretation was part of every step in sensation.

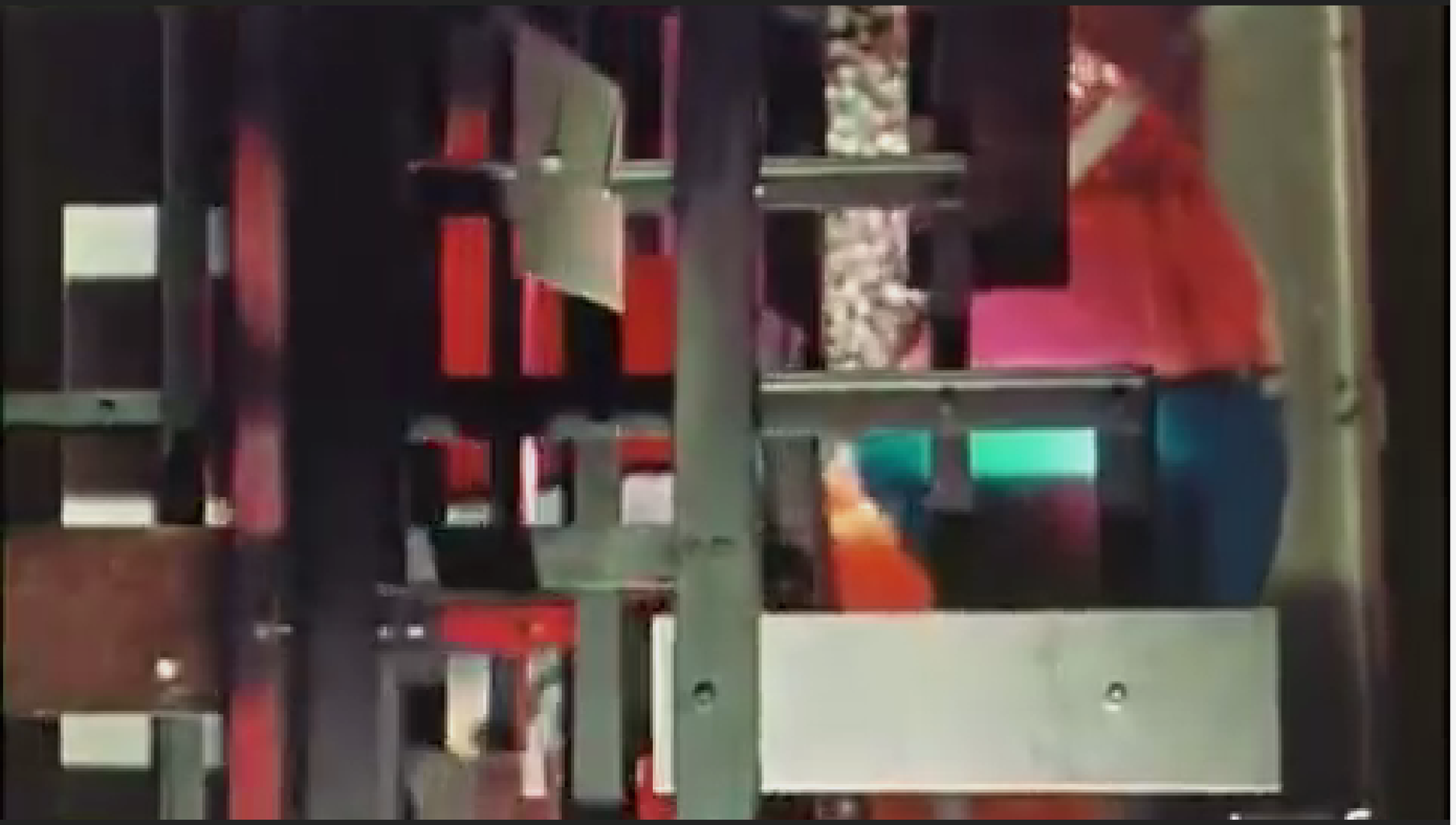
The mind, like other complex phenomenon, could only be understood by "explanations of the principle" or "pattern predictions" and could not be fully predicted. This mirrors his distaste for planned economies.

Credits

- http://public.econ.duke.edu/~bjc18/docs/Reflections%20on%20Hayek%27s%20Sensory%20Order.pdf

1956

CYSP 1 Robot



Nicolas Schöffer

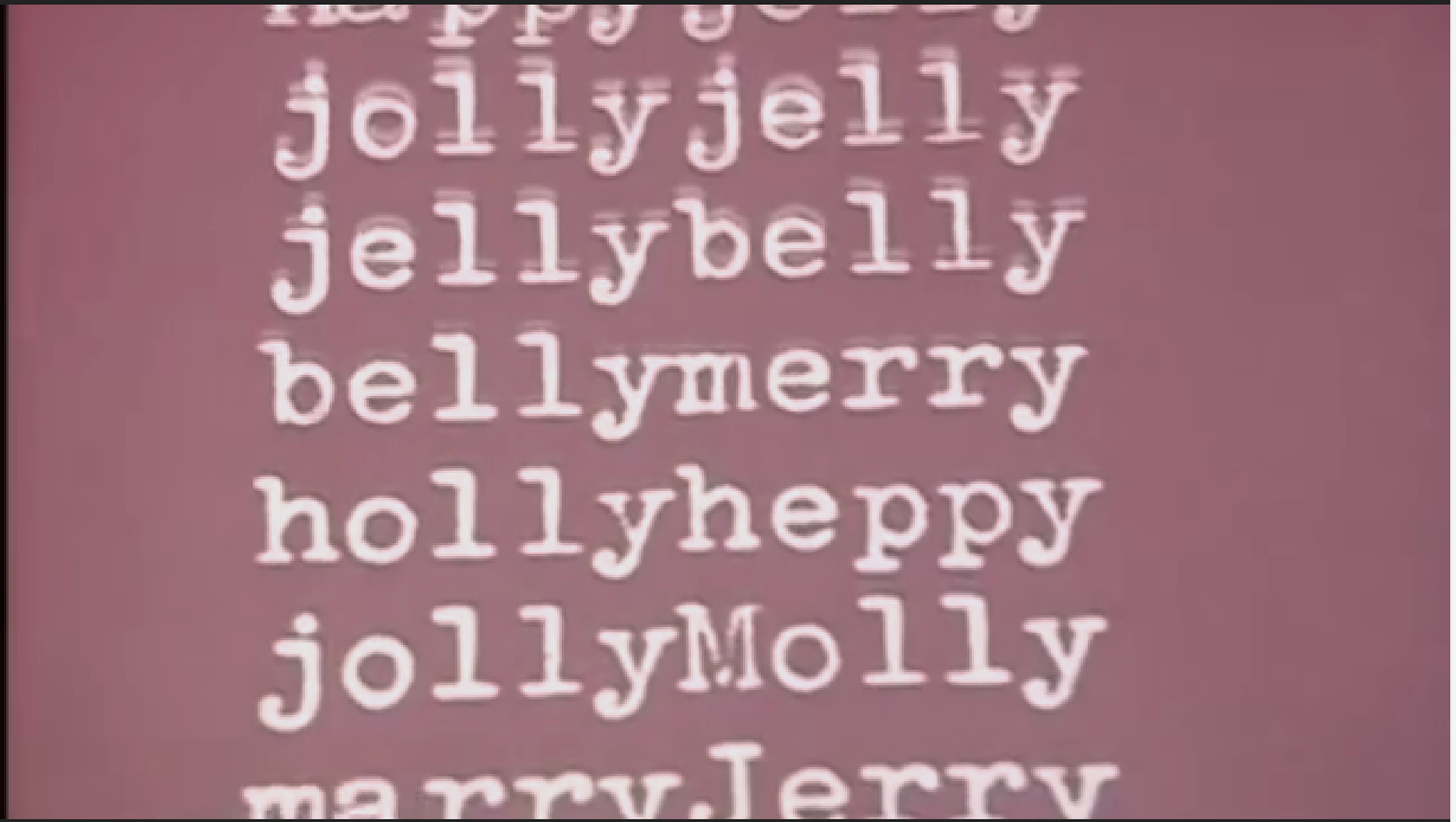
Nicolas Schöffer was a Hungarian-born French cybernetic artist. His *CYSP 1* robot was the first artwork to explicitly employ cybernetic principles. Phototubes and a microphone detected changes in color, light and sound intensity and elicited reactions by the robot.

Schöffer also belived art is a cultural asset that should be available equally to everyone without any limitation.

Credits

- *Cyspe 1* - Nicolas Schöffer

1968 *Cybernetic Serendipity* Exhibition



Jasia Reichardt

Jasia Reichardt is a British art critic, writer, and curator who is a specialist in the emergence of computer art. *Cybernetic Serendipity* was the first widely attended exhibition of computer and cybernetic art in the UK.

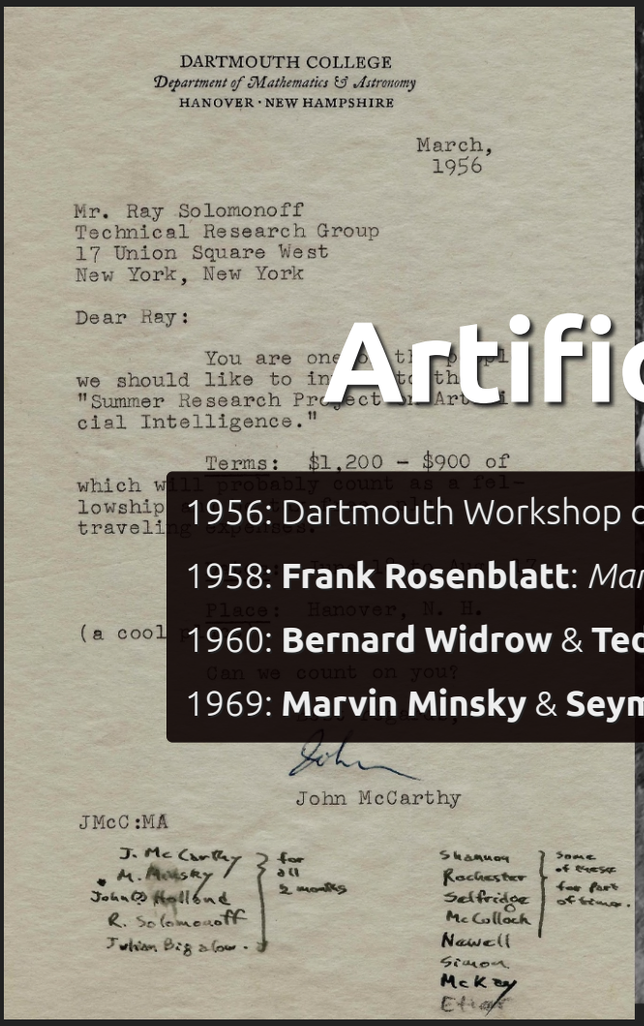
Credits

- *Cybernetic Serendipity* - Jasia Reichardt

Credits

- Perceptron Research from the 50's & 60's, clip

Artificial Intelligence


DARTMOUTH COLLEGE
Department of Mathematics & Astronomy
HANOVER · NEW HAMPSHIRE

March,
1956

Mr. Ray Solomonoff
Technical Research Group
17 Union Square West
New York, New York

Dear Ray:

You are one [illegible]
we should like to in[illegible]
"Summer Research Project [illegible]
cial Intelligence."

Terms: $1,200 - $900 of
which will probably count as a fel-
lowship
traveling
Place: Hanover, N. H.
(a cool
Can we count on you?

John McCarthy

JMcC:MA

J. McCarthy
M. Minsky
John Holland
R. Solomonoff
Julian Bigelow
} for all 2 months

Shannon
Rochester
Selfridge
McCulloch
Nawell
Simon
McKay
Etc.
} some of these for part of time.

1956: Dartmouth Workshop on Artificial Intelligence

1958: **Frank Rosenblatt**: *Mark I Perceptron:* first single-layer neural network

1960: **Bernard Widrow** & **Tedd Hoff**: *ADELINE* and *MADELINE* multi-layered neural networks

1969: **Marvin Minsky** & **Seymour Papert**: *Perceptrons* book published

The Dartmouth Summer Research Project on Artificial Intelligence is noted as the founding event of artificial intelligence as a field. In attendance were Minsky and Shannon, as well as others in the cybernetics, automata and information theory fields.

"The study is to proceed on the basis of the conjecture that every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it."

Interestingly, that phrasing foreshadows a future battle between two camps of AI researchers, the neural net based cybernetic connectionists and the logic and expert systems symbolic rationalists.

The McCuloch-Pitts neural model lacked a mechanism for learning, so by using neuroscience research by Hebb, a new model called a Perceptron was created that could learn simple classification. An alternate model was proposed not long after that included multiple layers but had other trade-offs. Progress began to stall.

An infamous event in AI history is the book *Perceptrons* Marvin Minsky and Seymour Papert. While the Perceptron had been a remarkable first step, it was only a single layer deep and no one could figure out how to train multi-layer Perceptrons. Additionally, computational limitations were holding back larger networks. The books pessimistic predictions for connectionist systems and support for symbolic systems ushered in the first "AI winter" where funding for AI basic research evaporated.

Credits

- https://www.skynettoday.com/overviews/neural-net-history

1956

DARTMOUTH COLLEGE
Department of Mathematics & Astronomy
HANOVER · NEW HAMPSHIRE

March,
1956

Mr. Ray Solomonoff
Technical Research Group
17 Union Square West
New York, New York

Dear Ray:

You are one of the people we should like to invite to the "Summer Research Project on Artificial Intelligence."

Terms: $1,200 - $900 of which will probably count as a fellowship and be tax free, plus traveling expenses.

Dates: June 18 to Aug. 17

Place: Hanover, N. H. (a cool place).

Can we count on you?

Best regards,

John McCarthy

JMcC:MA

J. McCarthy
M. Minsky
John Holland
R. Solomonoff
Julian Bigelow
} for all 2 months

Shannon
Rochester
Selfridge
McCulloch
Newell
Simon
McKay
Etc.
} some of these for part of time.

The Dartmouth Summer Research Project on Artificial Intelligence is noted as the founding event of artificial intelligence as a field. In attendance were Minsky and Shannon, as well as others in the cybernetics, automata and information theory fields.

1958 *Mark I Perceptron*

Frank Rosenblatt

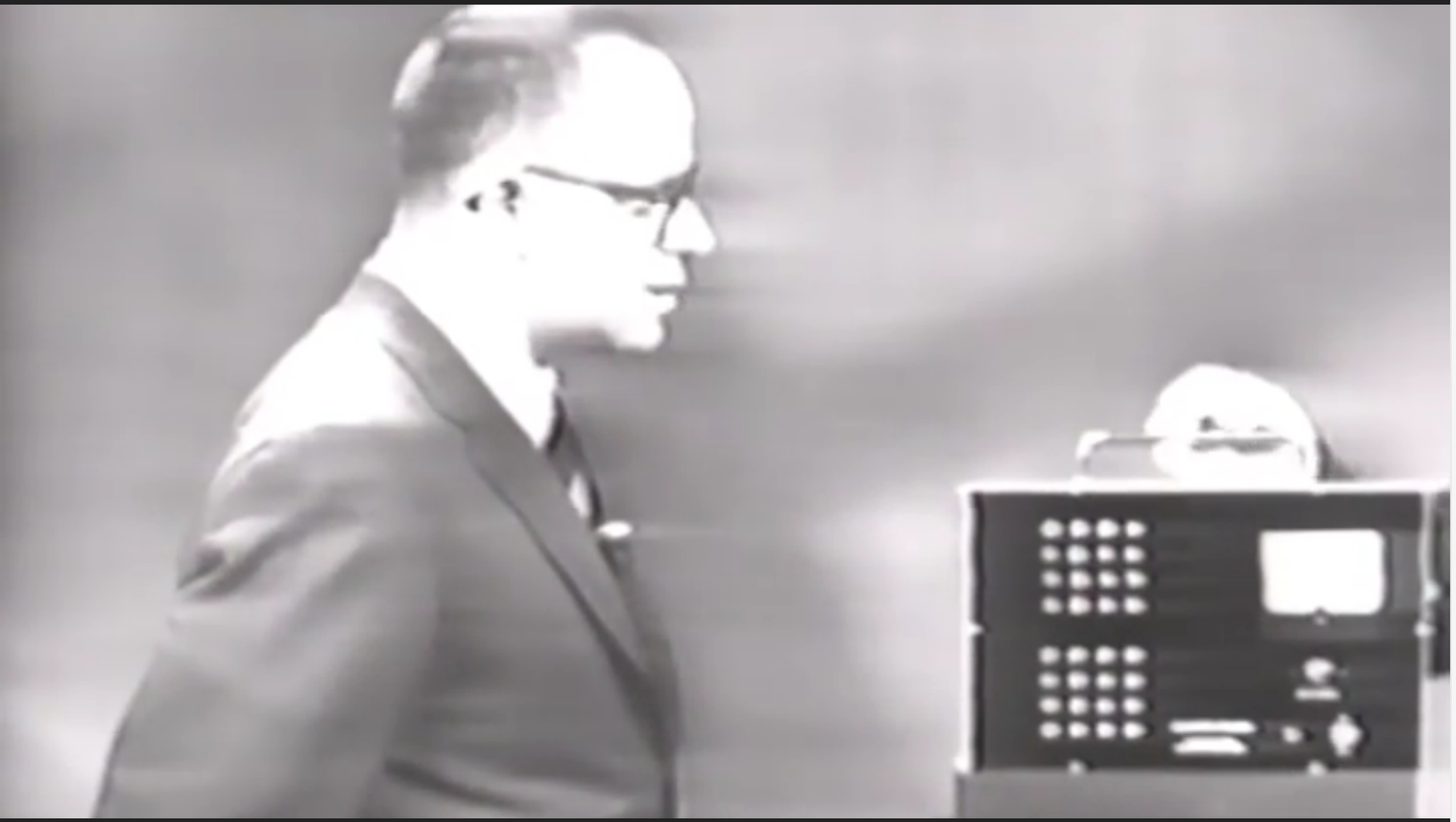
Frank Rosenblatt was an American psychologist who helped developed the *Mark I Perceptron*, the first computer that could learn by trial and error.

A percepton has an input, a single layer of artificial neurons and an output and was able to classify a small number of patterns.

Credits

- Perceptron Research from the 50's & 60's, clip

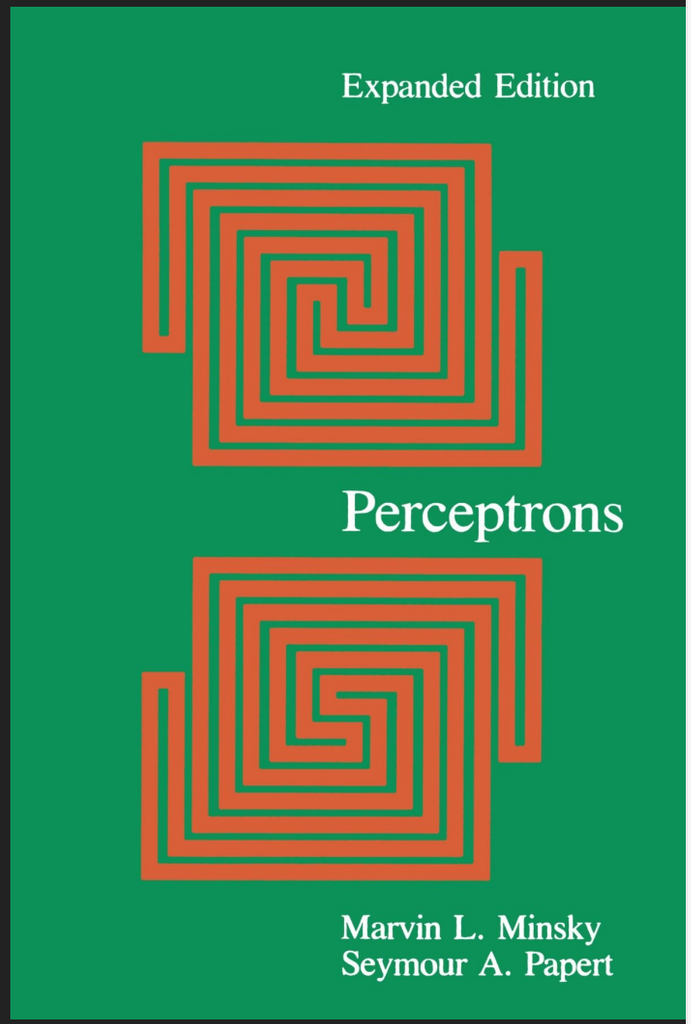
1960 Multi-Layer Neural Net



Bernard Widrow And Tedd Hoff

Bernard Widrow an American professor of electrical engineering and his grad student Tedd Hoff, later one of the inventors of the microprocessor, created the MADALINE system a 1000-weight trainable multi-layered neural networks using memistors. Initially not capable of training the middle layers, it was improved in 1988 to do so.

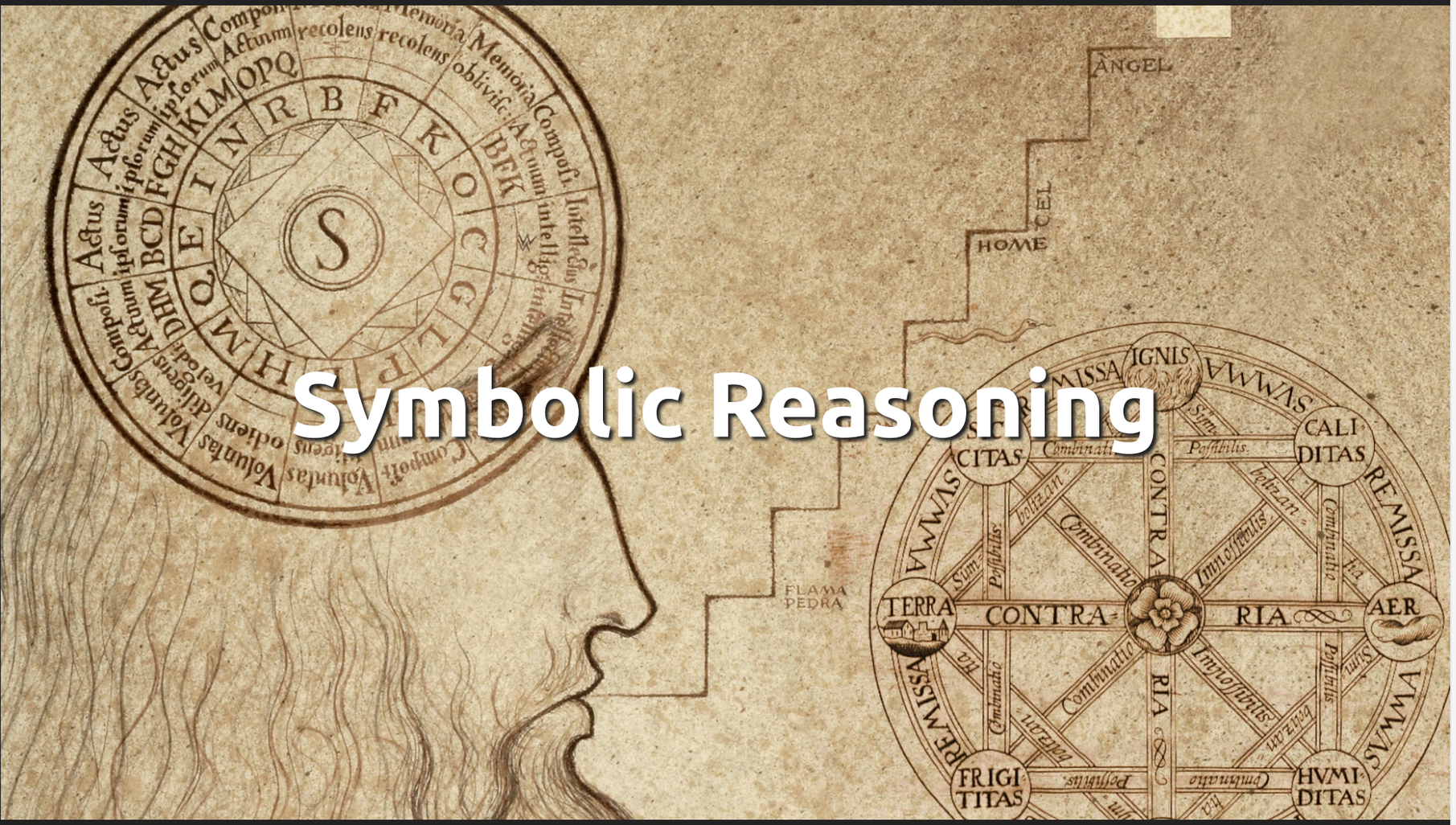
1969 *Perceptrons* Book



Marvin Minsky And Seymour Papert

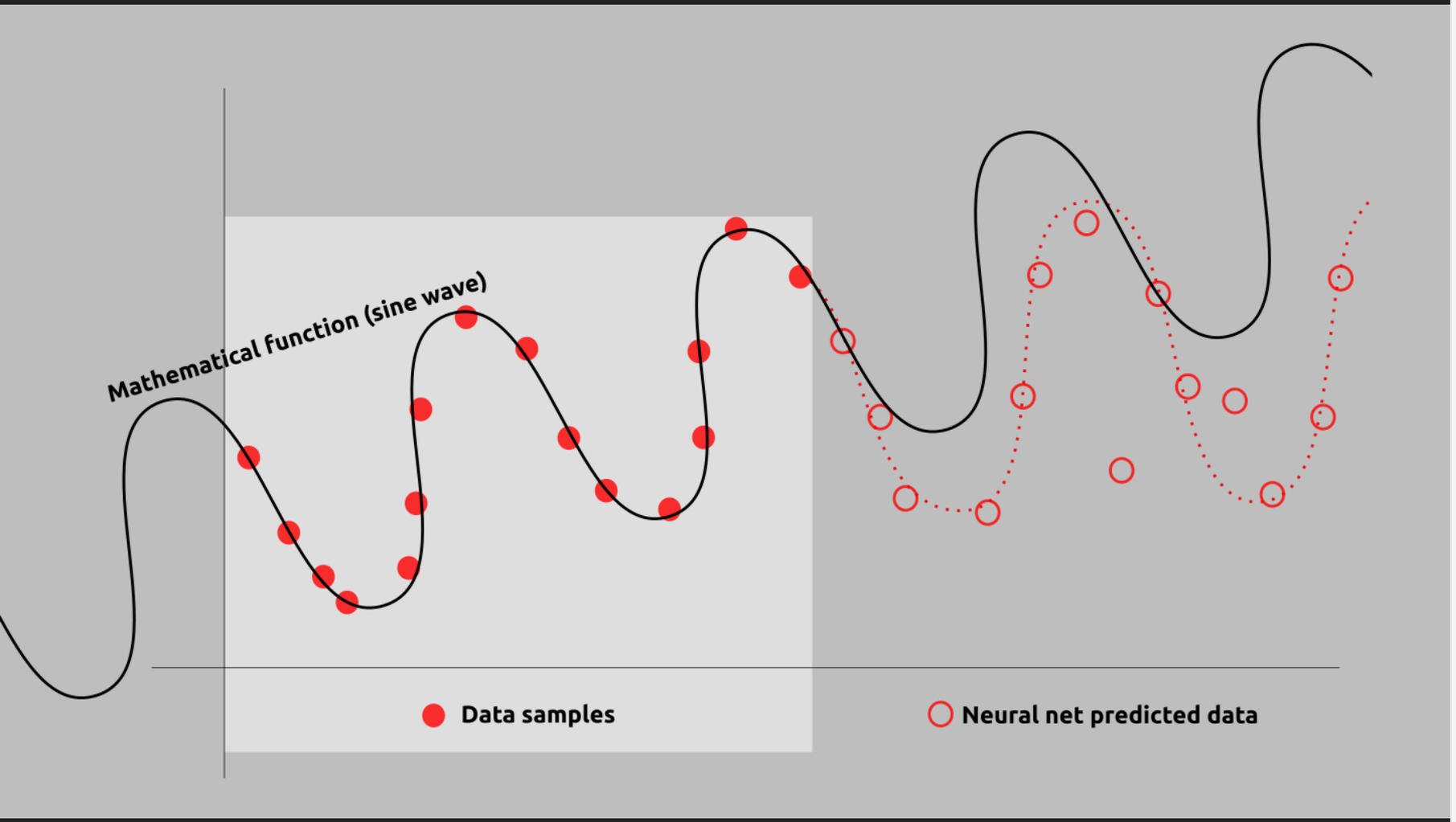
In 1969 Minsky and Seymour Papert published *Perceptrons*.

The book was critical of neural net research and potential and was considered the cause of a switch to symbolic expert systems.



Symbolic Reasoning

Building on the tradition of Leibniz and other great mathematicians one set of AI researchers imagined intelligence as rational thought where sensory perception was not required for intelligence. Instead, intelligence was defined by an idealized construction of semantic symbols, that could be construed as facts or truths, and the manipulation of those symbols through the rules of logic. Thinking was done with abstract concepts, not input, which merely needed to be converted to symbols so that it could be thought about.



Symbolic reasoning has one important crucial property that neural networks seemed to lack: once data was represented symbolically then predictions could be made about future or any other unseen data and the predictions could be accurate. This is the difference between a mathematical formula versus mimicy of the patterns of input data. Using a formula you can derive values at any point in time, rather than just matching the patterns seen in training data. Symbolic reasoning was "real" understanding versus pattern matching and mimicry done by the nets.

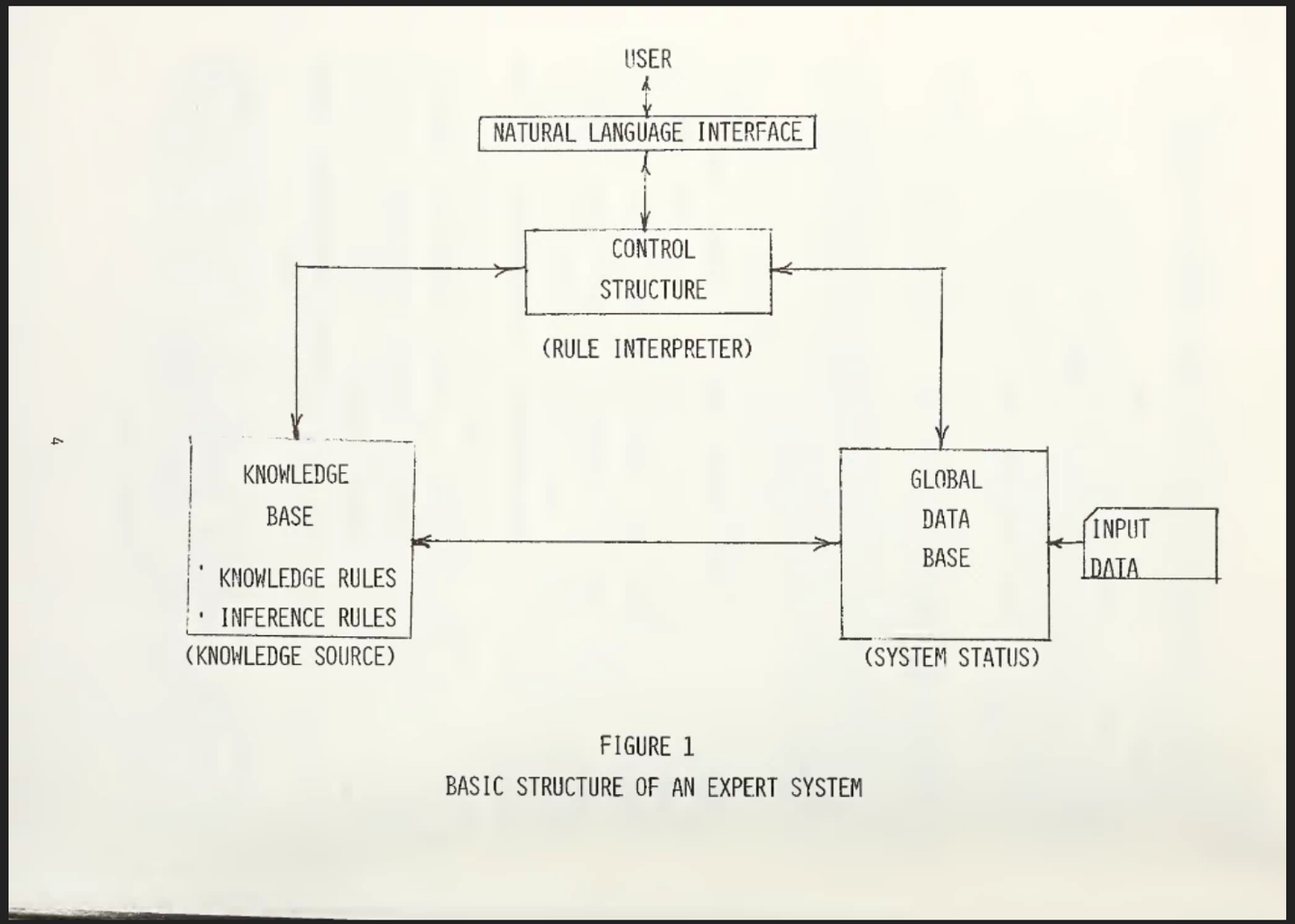


FIGURE 1
BASIC STRUCTURE OF AN EXPERT SYSTEM

Thus, the proponents of symbolic reasoning believed that reasoning was only possible by recording a "knowledge base" of facts generally agreed upon by experts as well as distilling the good judgment and expert decision-making based on the facts into rules or heuristics. This approach became known as expert systems and were popular during the 1980s.

Despite the pitfalls of uncertainty and undecidability, the symbolic logic researchers believed they could work around these problems using "fuzzy logic" and "truth maintenance". With hindsight, it seems incredibly utopian to imagine there are generally agreed upon facts for all things and that consistent logical rules could be applied to them.

Practically, building an expert system was difficult. Recording expert knowledge is much more difficult than it sounds and in many cases that knowledge is continually updating. CADUCEUS, an internal medicine expert system finished in the 1980s took a decade to build a knowledge base and was able to diagnose up to 1000 different diseases.

Credits

- https://en.wikipedia.org/wiki/Expert_system

Artificial Life And Cellular Automata

- 1948: **John von Neumann**: cellular automata
- 1969: **Alvy Ray Smith**: *Cellular Automata Theory*: proof of universal computation
- 1970: **John Conway**: *Game of Life*: cellular automaton
- 1975: **John Holland**: *Adaptation in Natural and Artificial Systems*: establishes genetic software algorithms
- 1984: **Christopher Langton**: *Langton's Loops*: first self-replicating computer organism with "genetic" code
- 1987: **Craig Reynolds**: *Boids* flocking behaviour
- 2002: **Stephen Wolfram**: *A New Kind of Science*: Rule 110, etc

Despite the funding drought for connectionist AI, there was still work being done on artificial life and genetic algorithms.

These systems originate from the cellular automata imagined by von Neumann. A cellular automata is regular grid of cells, each with a set of states, and an algorithm that determines the next generation of cells. Carefully constructed cellular automata can evolve complex, unpredictable but long-lived patterns. von Neumann was interested in self-replicating robots, but self-replication can be medium independent and is arguably a basic requirement for life.

It was almost 20 years later when the most famous cellular automaton, The Game of Life, helped illustrate the interplay of evolution, life and computation. However, cellular automata were poor at modelling neural networks and were considered ore of a toy than serious researcher.

Genetic algorithms or artificial evolution became a widely recognized optimization method in the 1960s and 1970s. It uses a process similar to evolution: create a large number of test subjects that are constructed using a system of composable parts and test them on the problem. The subjects are initially constructed randomly, and are terrible at solving the problem. Those that are best at solving the problem are kept, the others discarded, and then mutation or breeding of the survivors creates a new generation of test subjects.

The hyper competition and high mutation rates combined with millions of cycles of evolution allow the initially incapable problem solvers to evolve effective solutions.

The ultimate goal of artificial life according to Langton was, "to extract the logical form of living systems."

Credits

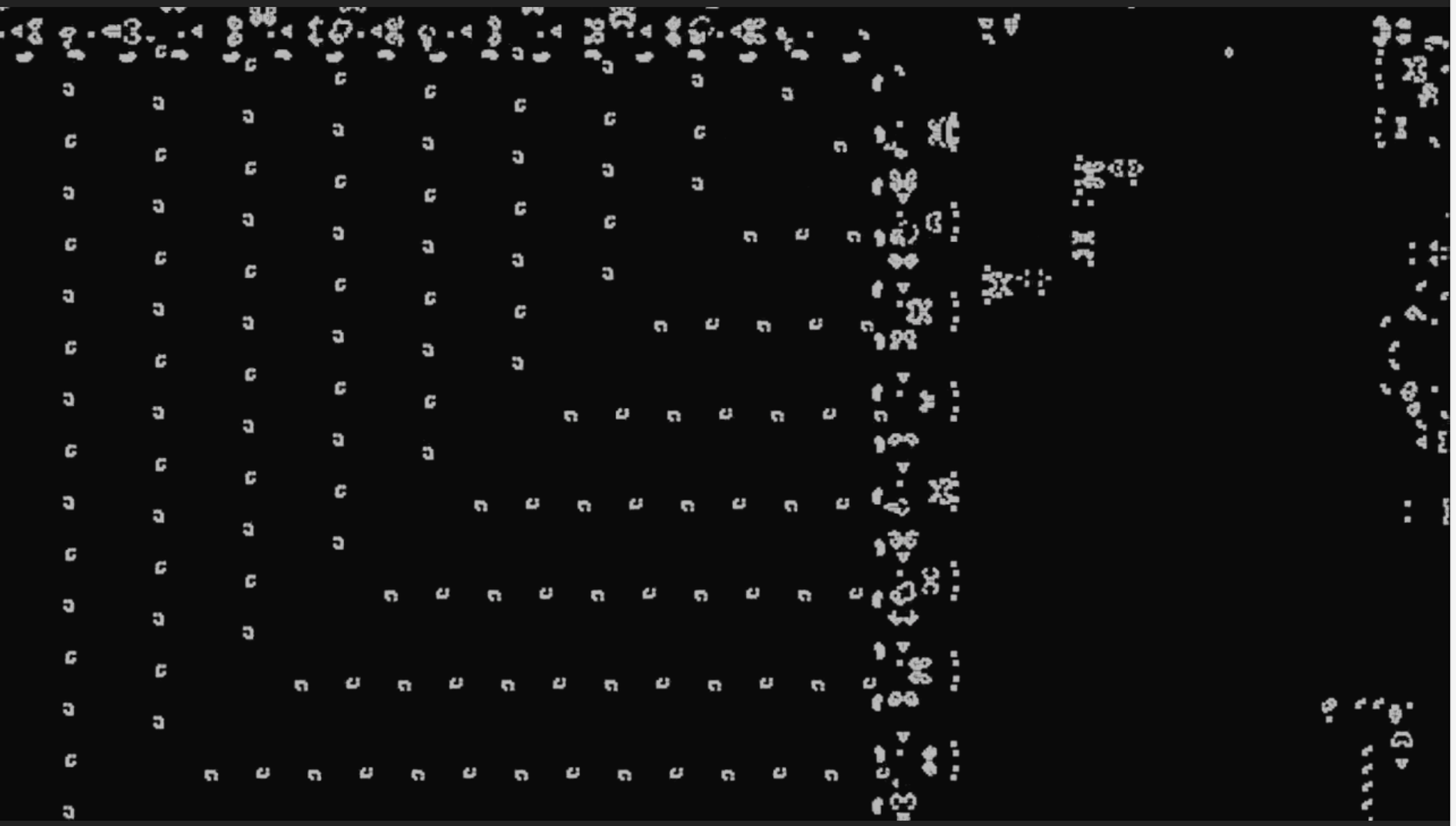
- *Breed* - Driessens & Verstappen

Credits

- https://en.wikipedia.org/wiki/History_of_artificial_life

1970

Game Of Life



John Conway

John Conway was an English mathematician who contributed to cellular automata theory particular in creating *The Game of Life*.

This software is a "zero player" game, its evolution determined entirely from its starting configuration, with no further input. It is Turing complete and follows three rules:

1. Any live cell with 2 or 3 live neighbours survives.
2. Any dead cell with 3 live neighbours becomes alive.
3. All other live cells die in the next generation.

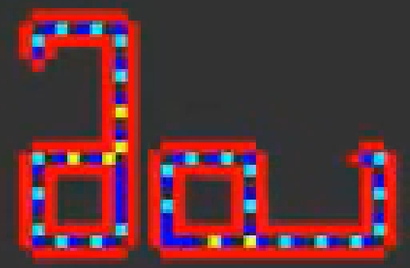
Play with the game and learn more at conwaylife.com.

Credits

- *Life in life* - Phillip Bradbury

1984

Langton’s Loops



Christopher Langton

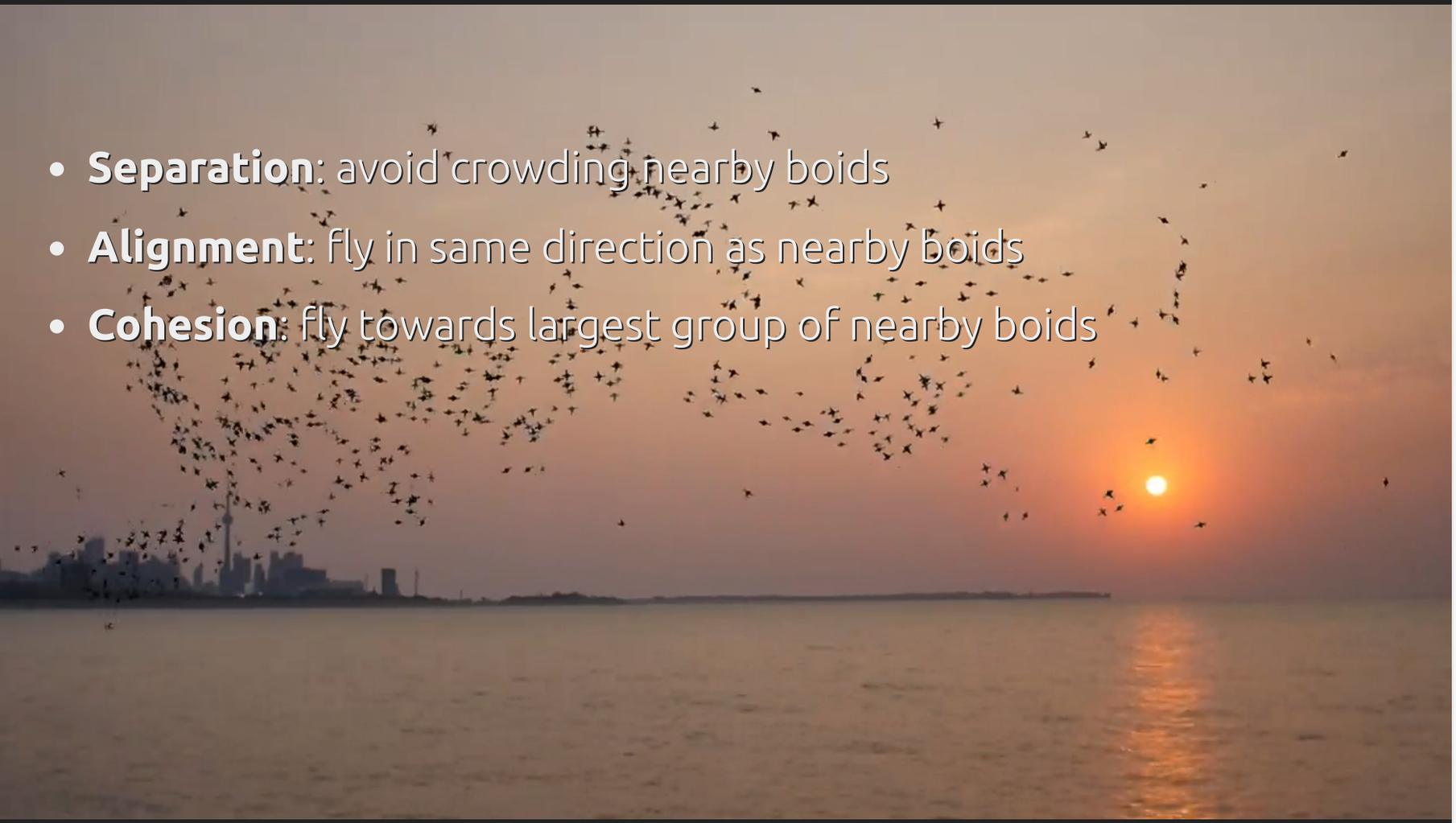
Christopher Langton an American computer scientist who studied cellular automata and artificial life.

Langton's Loops was the first self-replicating computer organism with “genetic” code.

Credits

- *Langton Loops* - Phillip Compeau

1987 *Boids*



Craig Reynolds

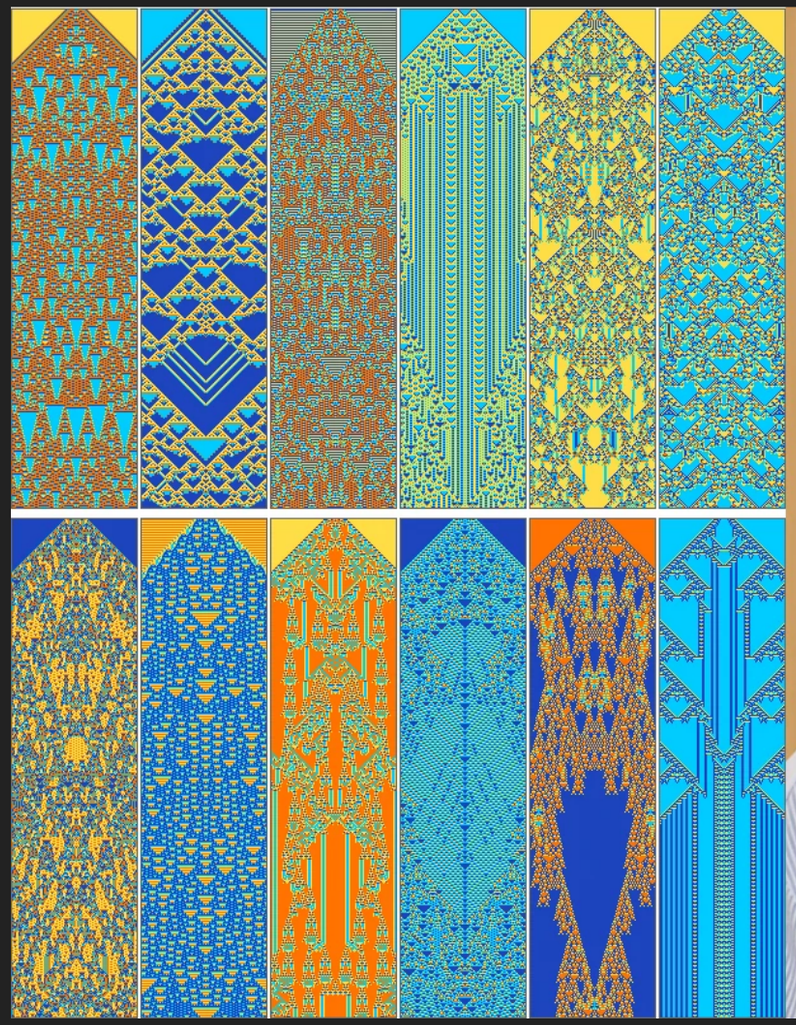
Craig Reynolds is an artificial life and computer graphics expert who created the *Boids* artificial life simulation. Boids obey three rules:

- **Separation**: avoid crowding nearby boids
- **Alignment**: fly in same direction as nearby boids
- **Cohesion**: fly towards largest group of nearby boids

Credits

- *Bird Brained* - Ryan Kelln & Jackson Welchner

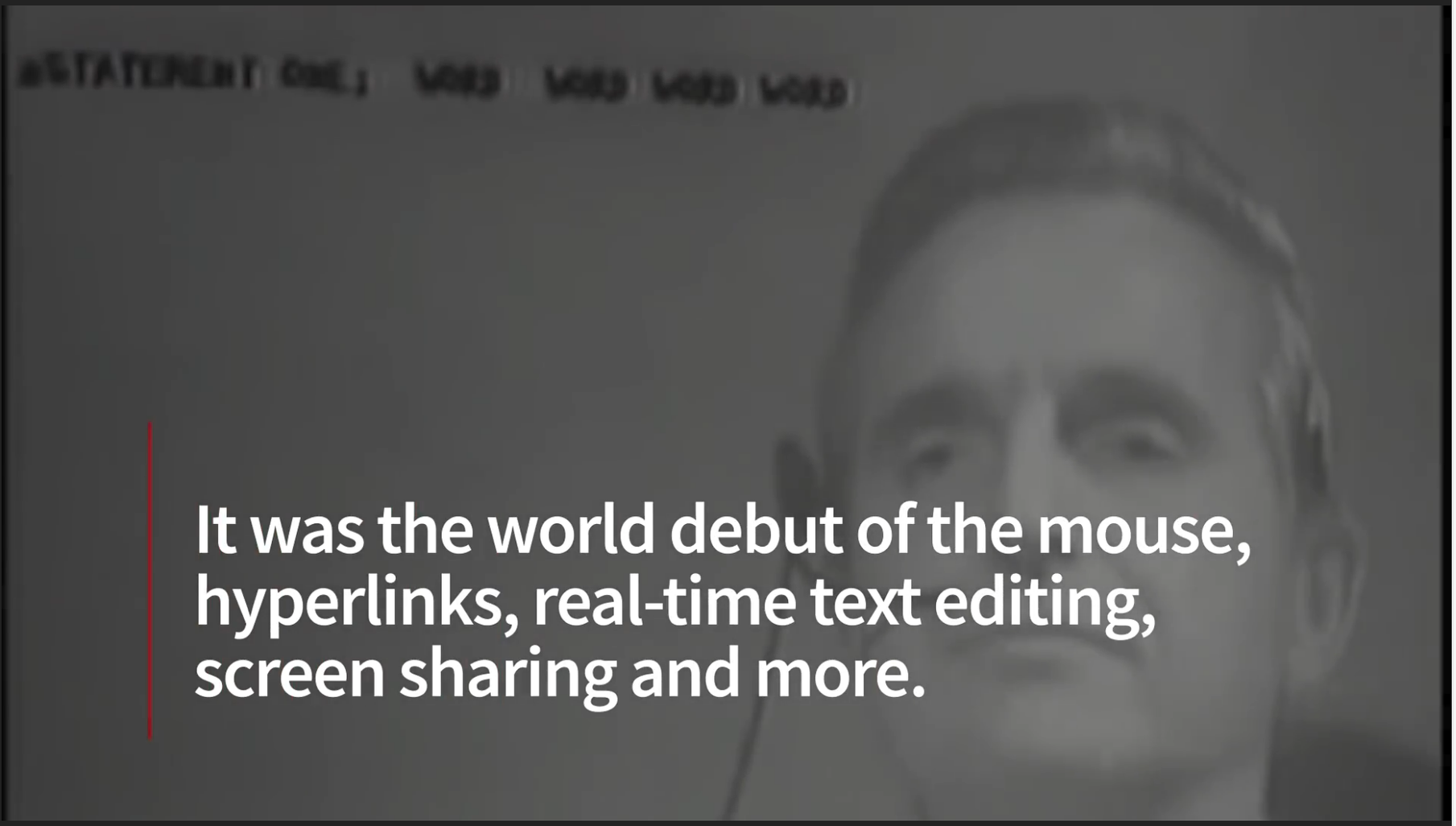
2002 *A New Kind Of Science*



Stephen Wolfram

Stephen Wolfram is a British-American computer scientist, physicist, and businessman who was the designer of the Mathematica software. His *A new Kind of Science* book presented an empirical study of simple computational systems: cellular automata and argued they were needed to model and understand complexity in nature.

For example, Rule 110, is a pattern that is unpredictable, on the boundary of stability and chaos and proven to be a Turing machine.



On Dec. 9, 1968, Doug Engelbart of the Stanford Research Institute (now SRI International) performed the “Mother of All Demos” — the world debut of personal and interactive computing. It was the realization of a vision he’d developed as a graduate student in electrical engineering: that we’d one day be able to “fly” through information spaces, and that the computer would become an extension of human communication capabilities and an augmentation of intellect.

Credits

- *The Mother of All Demos* - Doug Engelbart (1968)
- Courtesy of Stanford & SRI International

Origins Of Networked Personal Computers

1940s: first digital computers

1945: **Vannevar Bush**: memex

1952: **Grace Hopper**: first conception of programming using English language

1963: **Ted Nelson**: coins hypertext and imagines *Project Xanadu* a repository for the world's knowledge

1963: **Ivan Sutherland**: interactive display graphics program *Sketchpad*

1968: **Douglas Engelbart**: *"The Mother of All Demos":* mouse, bitmapped screens, hypertext, screensharing

1969: *ARPANET*: technical foundation for the internet

1990: **Tim Berners-Lee**: first web browser and World Wide Web

2001: Wikipedia: used for training most text-based neural nets

2006: **Fei-Fei Li**: *ImageNet* dataset collected from millions of web images

2007: 1st generation *iPhone*

2009: Graphics Processing Units (GPUs) made for video games used for neural net training

I'm going to skip over the developments of digital computers for the most part, even though they are crucial, the exact development of the hardware is relatively unimportant for machine learning, except to note that general purpose computers were always too slow for the massively parallel computations needed for neural nets. It wasn't until modern consumer graphics cards, developed for 3D video games became cheap and fast enough that training neural nets became feasible on affordable hardware.

So too today's modern internet and personal computing devices are not directly related to machine learning, but the amount of data that they enable has been instrumental in the success of machine learning. Machines need to learn from data, and without the mass adoption of personal computing, cheap storage and ubiquitous networks all made possible through digital technology this data would be unavailable.

As early as 1945 it was starting to be imagined that information could be stored and linked together, and that vision continued to be developed first in service to organizing data for the government but quickly being adopted by pioneers who imagined how it would transform society if everyone had access to easy to use hyperlinked knowledge tools connected to each other over networks.

The growth of all forms of digital tools, from digital photography, to GPS satellites and the world wide web, combined in devices like modern mobile phones created the environment where data and metadata is now so abundant that machines with no sensory perceptions of their own could learn about the world through these digital recordings.

Credits

- https://en.wikipedia.org/wiki/History_of_personal_computers

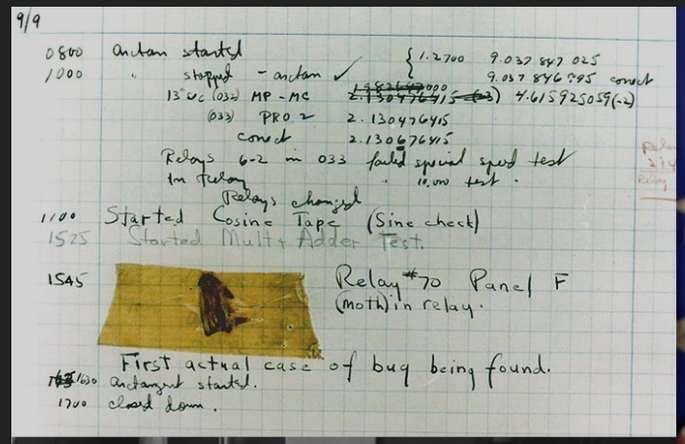
1945

Memex Machine

Vannevar Bush

Vannevar Bush, who founded the company that became Raytheon Technologies and initiated the Manhattan Project, also inspired generations of computer scientists in his 1945 essay *As We May Think* that described the *memex*, an electromechanical device in which individuals would compress and store all of their books, records, and communications, record new information such as photos, make comments and create and follow links between all the documents.

1952 High Level Programming Languages

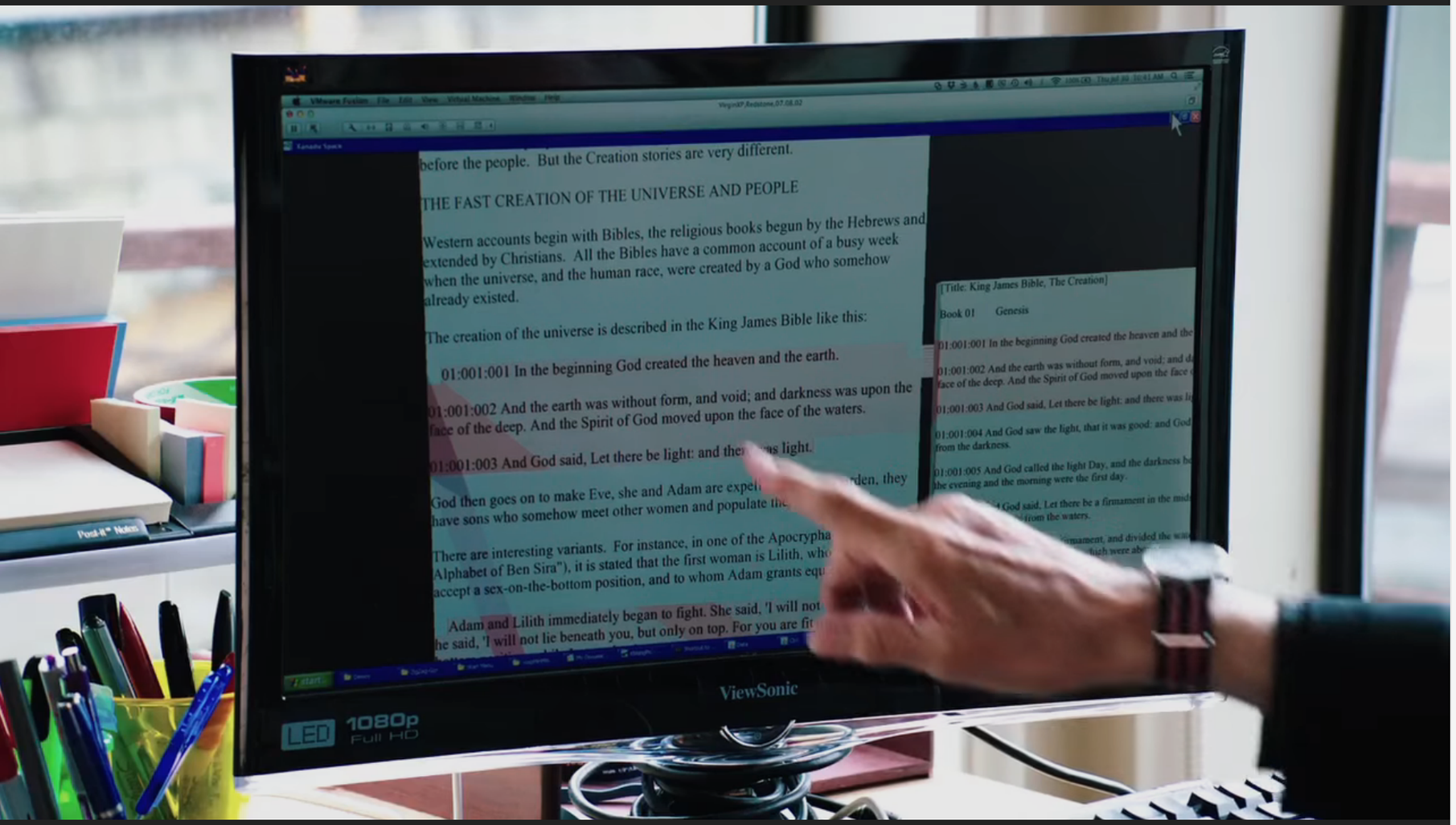


Grace Hopper

Grace Hopper was an American computer scientist and US Navy rear admiral. She was the first to conceive of programming using english language and compiling to machine language.

1963

Hypertext



Ted Nelson

Ted Nelson is an American philosopher and sociologist. He coined the terms hypertext and hypermedia in 1963 and founded *Project Xanadu* in 1960 to be a repository for the world's knowledge where data was connected by two-way (rather than the current web's one-way) links. *Xanadu* was under development until at least 2016, but hasn't published its code and file formats, dooming it to obscurity.

Credits

- Ted Nelson in Werner Herzog's "Lo and Behold"

1963 *Sketchpad*

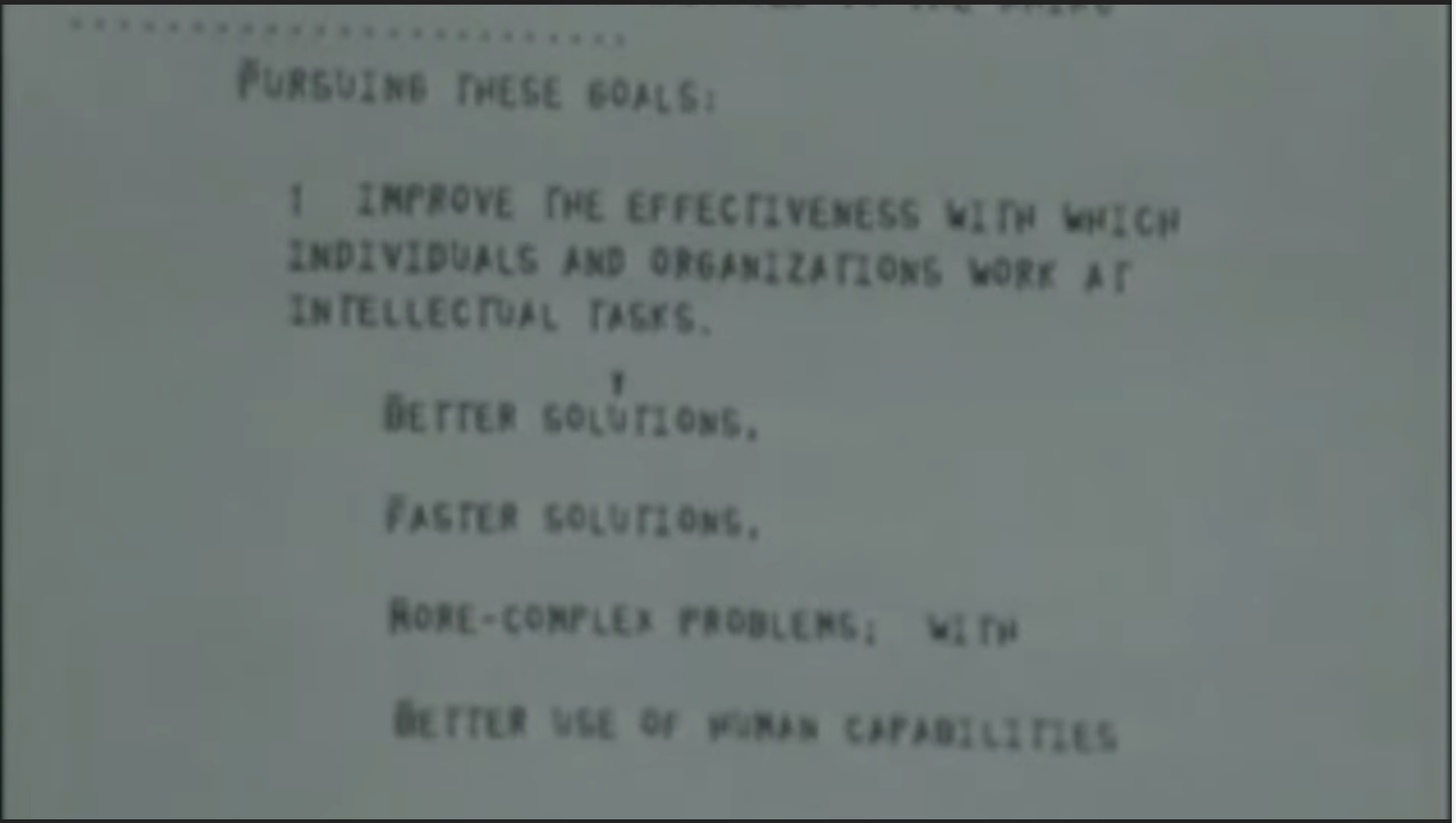
Ivan Sutherland

Ivan Sutherland is an American computer scientist. His interactive display graphics program *Sketchpad* was among the first graphical user interfaces (GUIs) and demonstrated its use for both artistic and technical work.

Credits

- *Computer Sketchpad* - Ivan Sutherland

1968 *The Mother Of All Demos*



Douglas Engelbart

Douglas Engelbart was an American engineer and inventor. He was part of a group at SRI International that developed the mouse, bitmapped screens, hypertext, and screensharing.

Credits

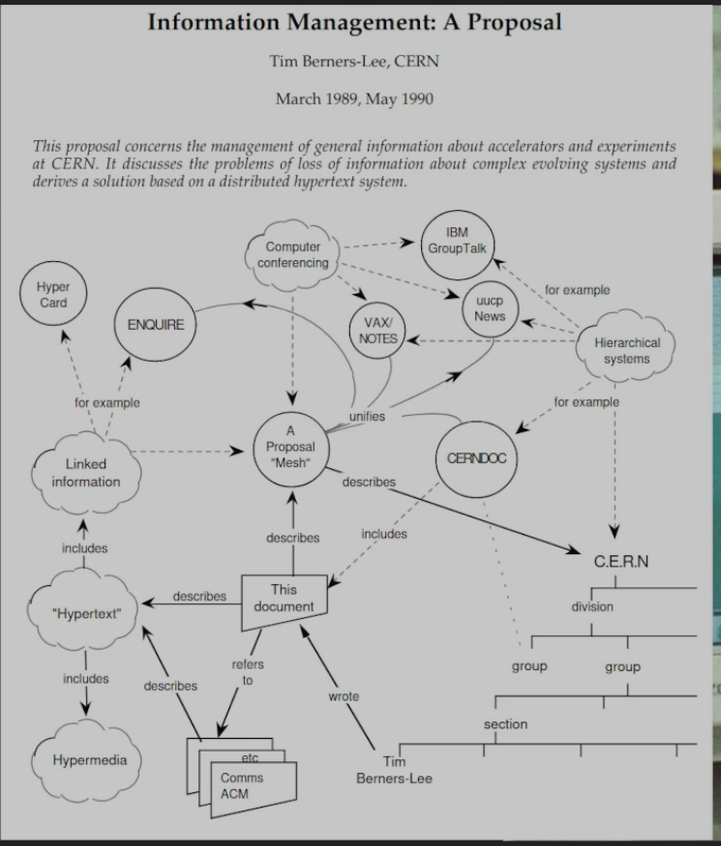
- Goals of the Program - from Doug's 1968 Demo Highlights

Credits

- https://dougengelbart.org/
- https://dougengelbart.org/content/view/194/

1990

World Wide Web

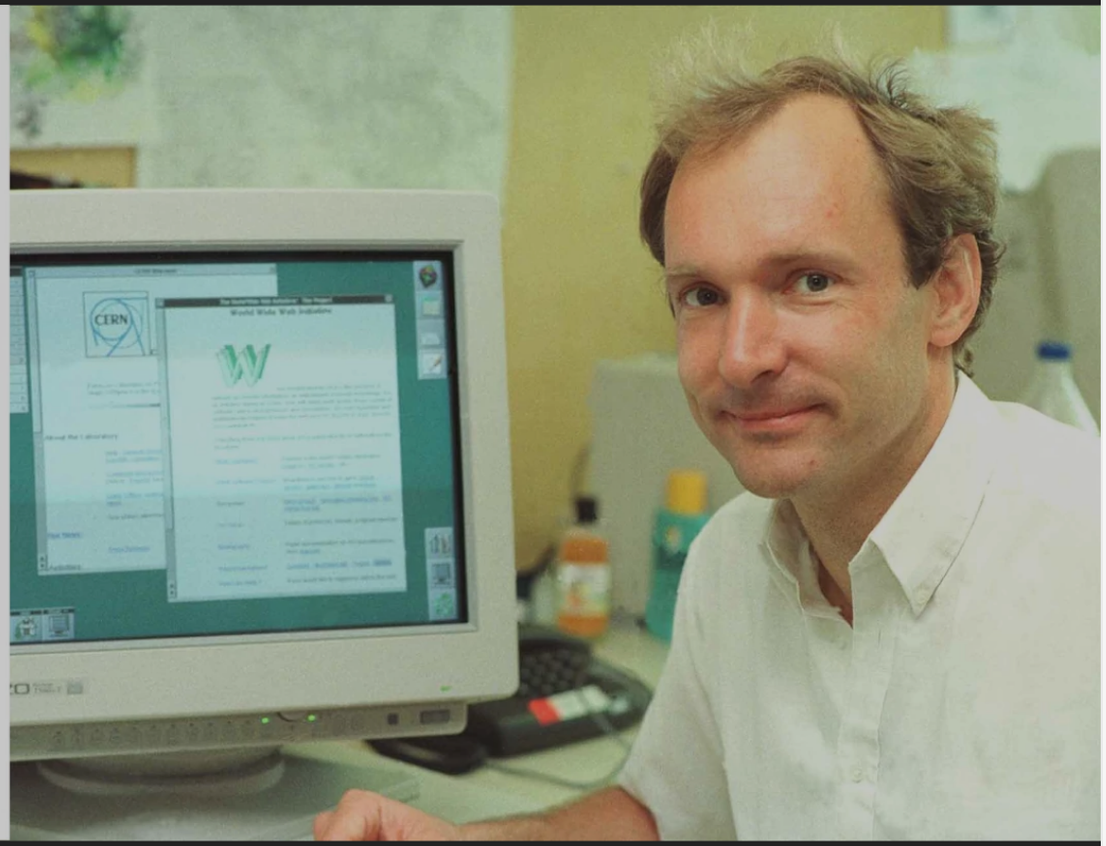


Information Management: A Proposal

Tim Berners-Lee, CERN

March 1989, May 1990

This proposal concerns the management of general information about accelerators and experiments at CERN. It discusses the problems of loss of information about complex evolving systems and derives a solution based on a distributed hypertext system.



Tim Berners-Lee is an English computer scientist. In 1989, while working at CERN, he proposed and implemented the first web browser (and HTML editor) and web server and helped establish the World Wide Web. This connection of hypermedia to the early Internet shaped the next 30 years.

Tim Berners-Lee

Credits

- https://webfoundation.org/about/vision/history-of-the-web/
- https://cds.cern.ch/record/369245/files/dd-89-001.pdf

2006 *ImageNet*

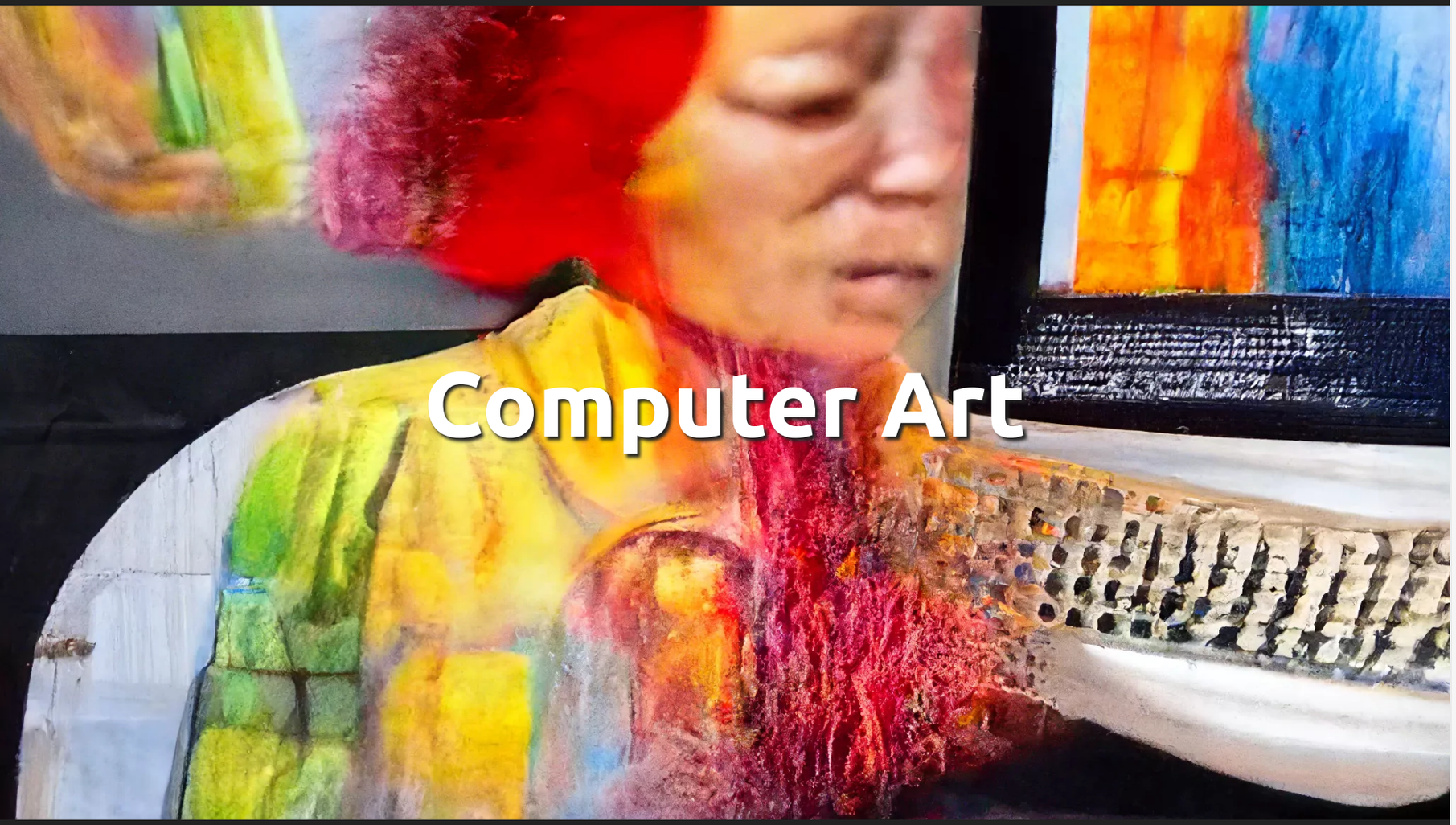
Fei-Fei Li

Fei-Fei Li (李飞飞) is a Chinese-American computer scientist who studies vision. Her lab created the *ImageNet* dataset of images collected from the Internet, using queries and searches and then used Amazon Mechanical Turk to clean the dataset.

ImageNet is still growing and changing (recently dealing with issues of privacy and bias) but currently contains 14 millions images in 20000 categories.

Credits

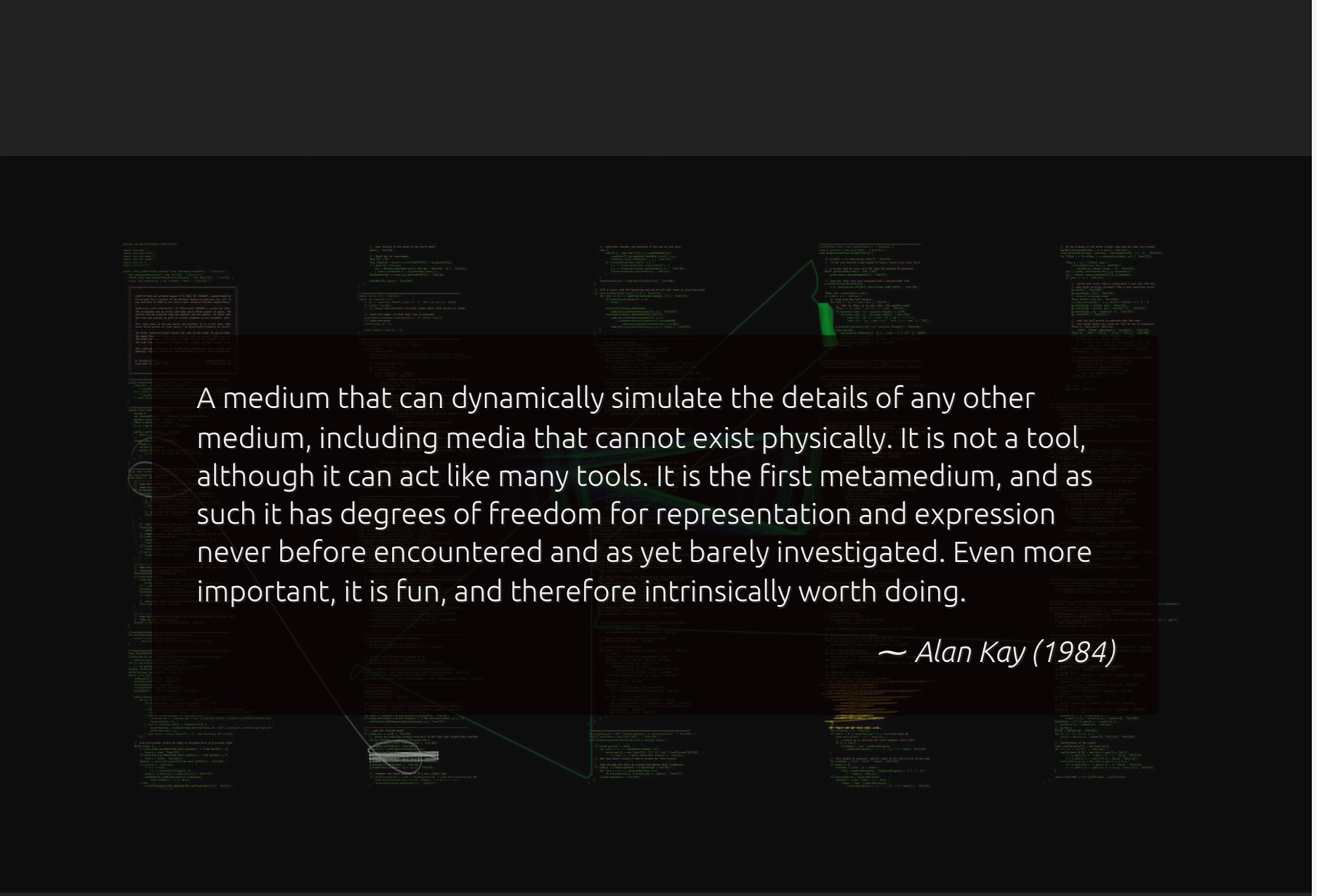
- *Computer Orchestra* - Peter Zinovieff (1968)



Computer Art

Let's take a look at a very incomplete sample of different forms of art made using computers, some with neural nets, many with some form of evolving system, others just generated by software. These are chosen based on available documentation, impact on the art world, or demonstration of diverse techniques, often some mix of the three, but are very arbitrary!

All of these artists are using software to create their art. Software can be thought as a meta-medium, envisioned by Alan Kay in 1984, as:



A medium that can dynamically simulate the details of any other medium, including media that cannot exist physically. It is not a tool, although it can act like many tools. It is the first metamedium, and as such it has degrees of freedom for representation and expression never before encountered and as yet barely investigated. Even more important, it is fun, and therefore intrinsically worth doing.

Credits

- *Code Profiles* - W. Bradford Paley

Credits

- Alan Kay, “Computer Software,” Scientific American 251, no. 3 (1984): 52–59, quote on 59.
- https://www.jstor.org/stable/24920344?refreqid=excelsior%3Afd0842ab4ac53b944798ae30f36e7564

Early Computer Art

1968: **Peter Zinovieff**: *Computer Orchestra:* early electronic music using PDP8

1968: **Georg Nees**: *Schotter (Gravel):* software generated and printed by plotter

1970: **Edward Ihnatowicz**: *Senster:* robotic sculpture with sensors to react to the behaviour of audience

1972: **Lillian F. Schwartz**: *Mutations:* computer aided visuals set to music

1973: **Harold Cohen**: *AARON:* software for creation of artistic images using an “expert system”

1974: **Vera Molnár**: *(Dés)Ordres:* software generated and printed

1976: **Analiva Cordeiro**: *Cambiantes:* algorithmic dance choreography

Early software pioneers often worked with robots as kinetic sculptures or with printed media using plotters or early printers of some type. As televisions and other video works became more common in the 70s other pioneers incorporated software into video production techniques. As techniques developed artists began incorporating more evolutionary algorithms into their work, playing with giving up control.

Computer Art

1987: **William Latham** & **Stephen Todd**: *Mutator 1:* beginning evolved 3D graphics used as art

1991: **Karl Sims**: *Primordial Dance:* animation of textures and colors from evolved mathematical equations

1992: **Nicolas Baginsky**: *Aglaopheme:* robotic electric guitar using Self Organizing Maps

1999: **Scott Draves**: *Electric Sheep:* distributed artificial life visualization

2003: **Jared Tarbell**: *Substrate:* opensource algorithmic visual art

2004: **Jon McCormack**: *Eden:* evolutionary sonic ecosystem

2005: **Jaap Blonk** & **Golan Levin**: *Ursonography:* audiovisual spoken poetry with reactive subtitles

2006: **Alain Lioret**: *Painting Beings:* evolved brushstrokes that interact with each other

2011: **David Rokeby**: *Plot Against Time:* video installation, drawing movement trajectories

2018: **Allison Parrish**: *Articulations:* generated poems learned from 2 million lines of public poems

This took on new significance when the computer could "render" out 3 dimensional images onto the 2D screen, but these forms did not have to be objects, instead could be ethereal blends of light and colour, visualized mathematics. Alain Loriet even explored paint-like evolving creatures. Scott Draves' ongoing *Electric Sheep* project demonstrates what years of work in this direction can deliver using a genetic-style language to describe visual elements and distributed rendering by thousands of participants who can also craft and breed their own visual genetic codes.

Jared Tarbell's *Substrate* exemplifies what similar algorithmic unfolding of shape and colour can achieve, but with a completely different aethstetic from the hallucinegetic high intensity visuals of the sheep.

Early video techniques have evolved into sublime computer aided video works by artists like David Rokeby and been combined with other types of performances. The diversity of software expression is growing. I will cover much more contemporary machine learning in fifth tutorial, but I'll start you with an appetizer of Allison Parrish's delightful use of ML language models in her poetry.

Credits

- https://www.artnome.com/news/2018/8/8/why-love-generative-art
- https://www.invaluable.com/blog/generative-art/
- http://iasl.uni-muenchen.de/links/GCA-IV.3e.html
- http://iasl.uni-muenchen.de/links/GCA-II.3e.html
- https://itp.nyu.edu/adjacent/issue-3/articulations-a-fragment-fragment-fragment/

TODO:

- 2012: Francisco Vico: *Melomics*: music composition algorithms using simulated evolution
- *0music* - Iamus by Francisco Vico

1968 *Partita For Unattended Computer*

Peter Zinovieff

Peter Zinovieff was a British engineer and composer, son of Russian aristocrats who fled the Russian Revolution. He bought his first computer (he claimed "the first computer in a private house") by auctioning his first wife's tiara.

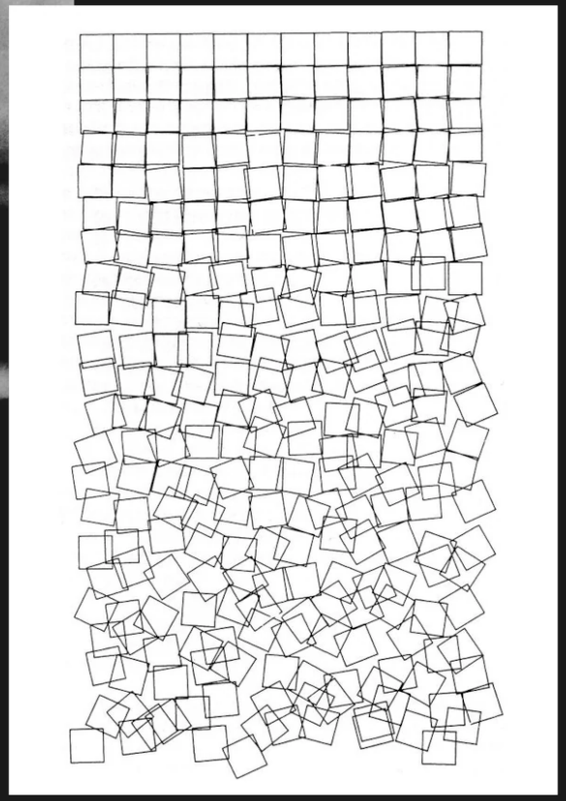
His *Partita for Unattended Computer* was the first ever unaccompanied performance of live computer music, with no human performer involved, and the piece read from paper tape. Later that year, as part of *Cybernetic Serendipity* exhibition, he used a PDP-8 to analyze a tune whistled by a visitor to the show and improvise upon it.

Credits

- https://120years.net/wordpress/ems-synthesisers-peter-zinovief-united-kingdom-1969/

1968

Schotter (Gravel)



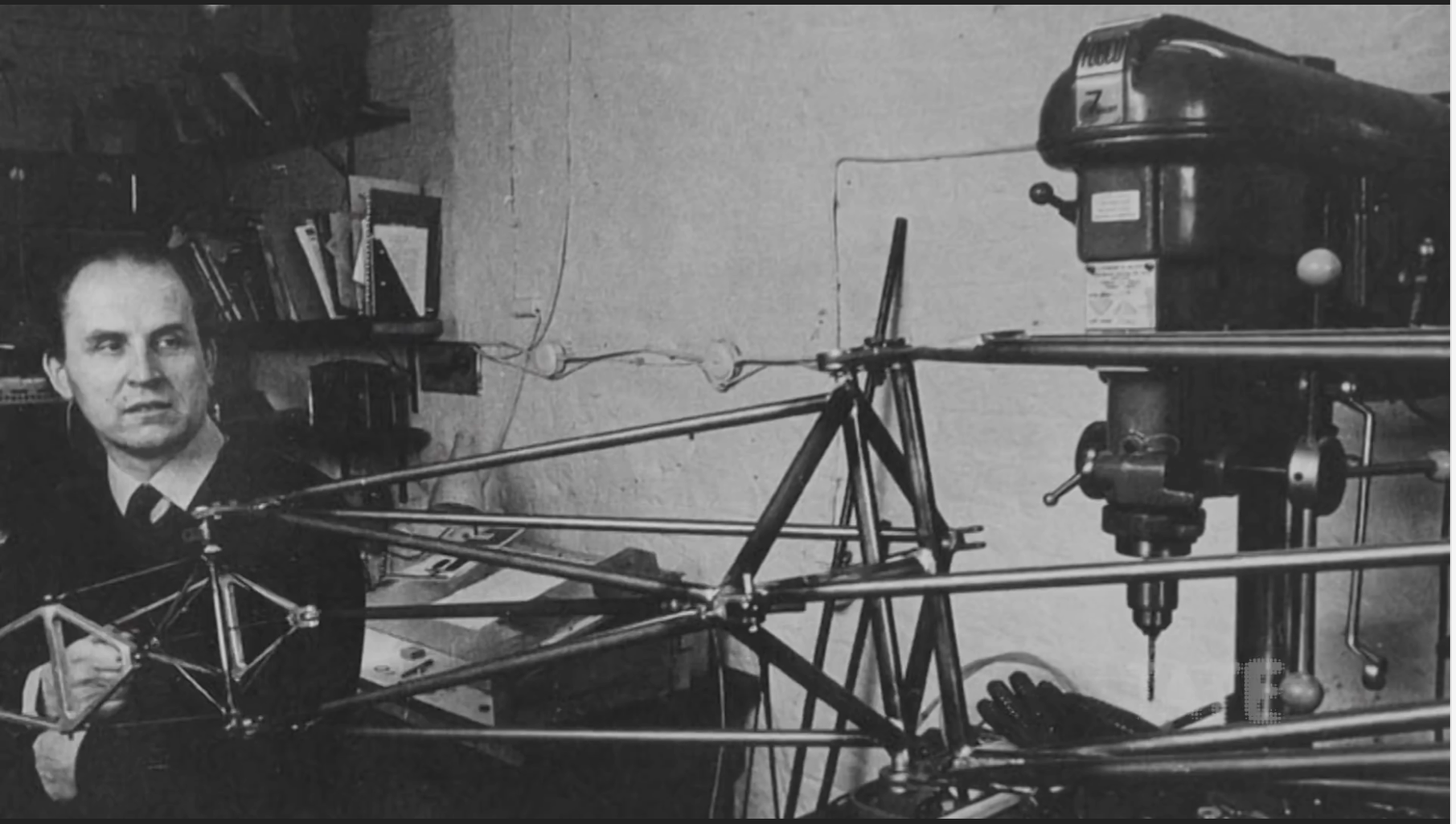
Georg Nees

Georg Nees was a German academic who studied mathematics, physics and philosophy. In 1965, Nees showed the world's first computer graphics art works created with a digital computer. He worked with random numbers fed into math equations to have flatbed plotters or milling machines create physical output.

Credits

- https://en.wikipedia.org/wiki/Georg_Nees

1970 *Senster*



Edward Ihnatowicz

Edward Ihnatowicz was a Polish cybernetic art sculptor who fled during the war and eventually immigrated to Britain.

Senster was the first robotic sculpture to be controlled by a computer. It was a large hydraulically actuated robot that followed the sound and motion of the people around it, giving the impression of being alive. It used an array of four microphones to detect the direction of the sound around it and two Doppler radar arrays to measure the motion of people.

Credits

- Art-ificial Intelligence - Lost Art

1972 *Mutations*

Lillian F. Schwartz

Lillian F. Schwartz is an American artist who studied to be a nurse and was stationed in postwar Japan. She started working with Bell Labs engineers and Ken Knowlton to produce a series of computer animated films.

Credits

- *Mutations* - Lillian F. Schwartz

1973 *AARON*

Harold Cohen

Harold Cohen was a British-born artist who created the *AARON* expert system software that could create line drawings that Cohen would colour after printing.

Started in 1973 in the C programming language but converted later to Lisp, *AARON* created abstract line drawings. Representational imagery was then added; first rocks, then plants, then people, and interior scenes. *AARON* cannot learn new styles or imagery on its own; each new capability must be hand-coded.

Credits

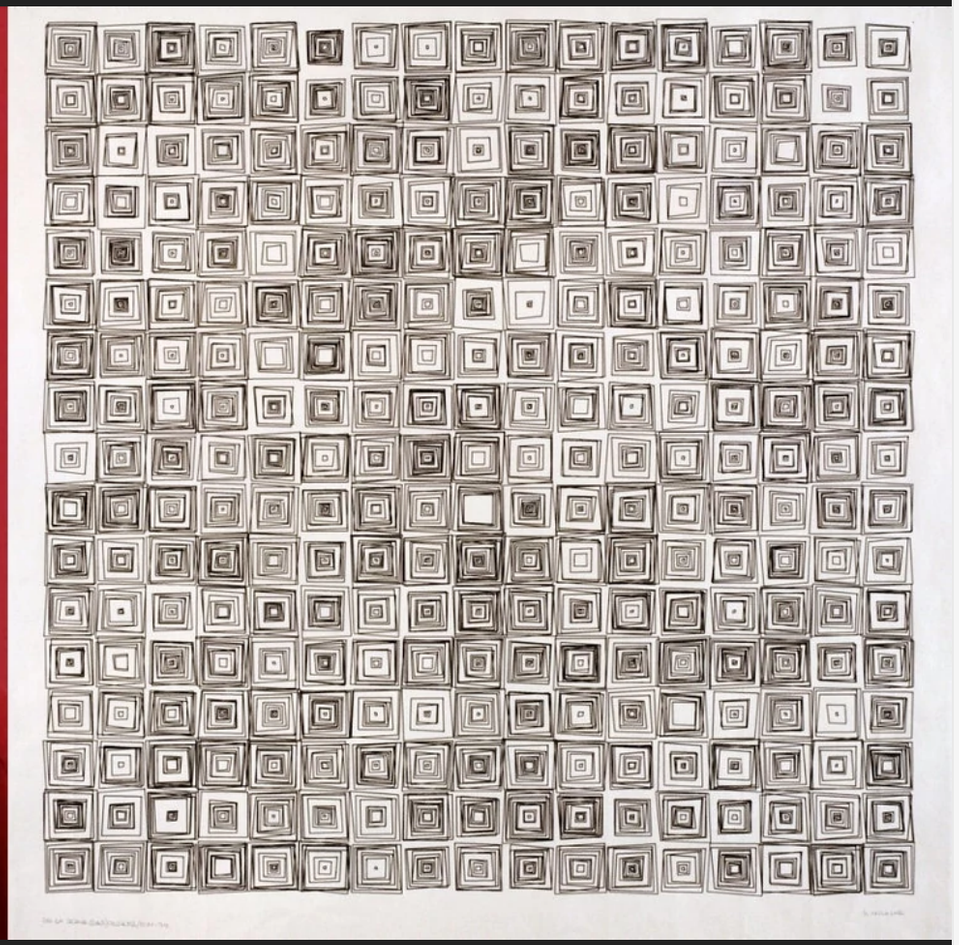
- Harold Cohen - The Age of Intelligent Machines - 1987 (Clip)

Credits

- https://en.wikipedia.org/wiki/AARON

1974

(Dés)Ordres



Vera Molnár

Vera Molnár is a Hungarian media artist living and working in France. Molnar learned the early programming languages of Fortran and Basic, and gained access to a computer at a research lab in Paris where she began to make computer graphic drawings on a plotter.

veramolnar.com

Credits

- http://www.veramolnar.com/

1976 *Cambiantes (Changing)*

Analivia Cordeiro

Analivia Cordeiro a Brazilian dancer, choreographer and architect was one of the first artists to use software for dance.

Using the Fortran programming language to program positions of body limbs at 45 degree and right angles, Cordeiro wanted to make a geometrical dance. This is reinforced by the costumes and sets.

analivia.com.br

Credits

- *Cambiantes* - Analivia Cordeiro

Credits

- https://static1.squarespace.com/static/5d93621cc1ed84125185a315/t/5f9093b9ad10751ed2958f8e/1603310551642/Analivia+Cordeiro_Cambiantes_Aninat+Galeri%CC%81a.pdf

1987 *Mutator 1*

William Latham, Stephen Todd

William Latham and Stephen Todd worked at IBM together and used genetic algorithm techniques to generate and animate 3-d models. Latham, an artist, and Todd, a mathematician and programmer, were interested in psychedelic evolving organic art.

In 1992 they released a book *Evolutionary Art and Computers* and have continued to develop Mutator into a VR experience.

mutatorvr.co.uk

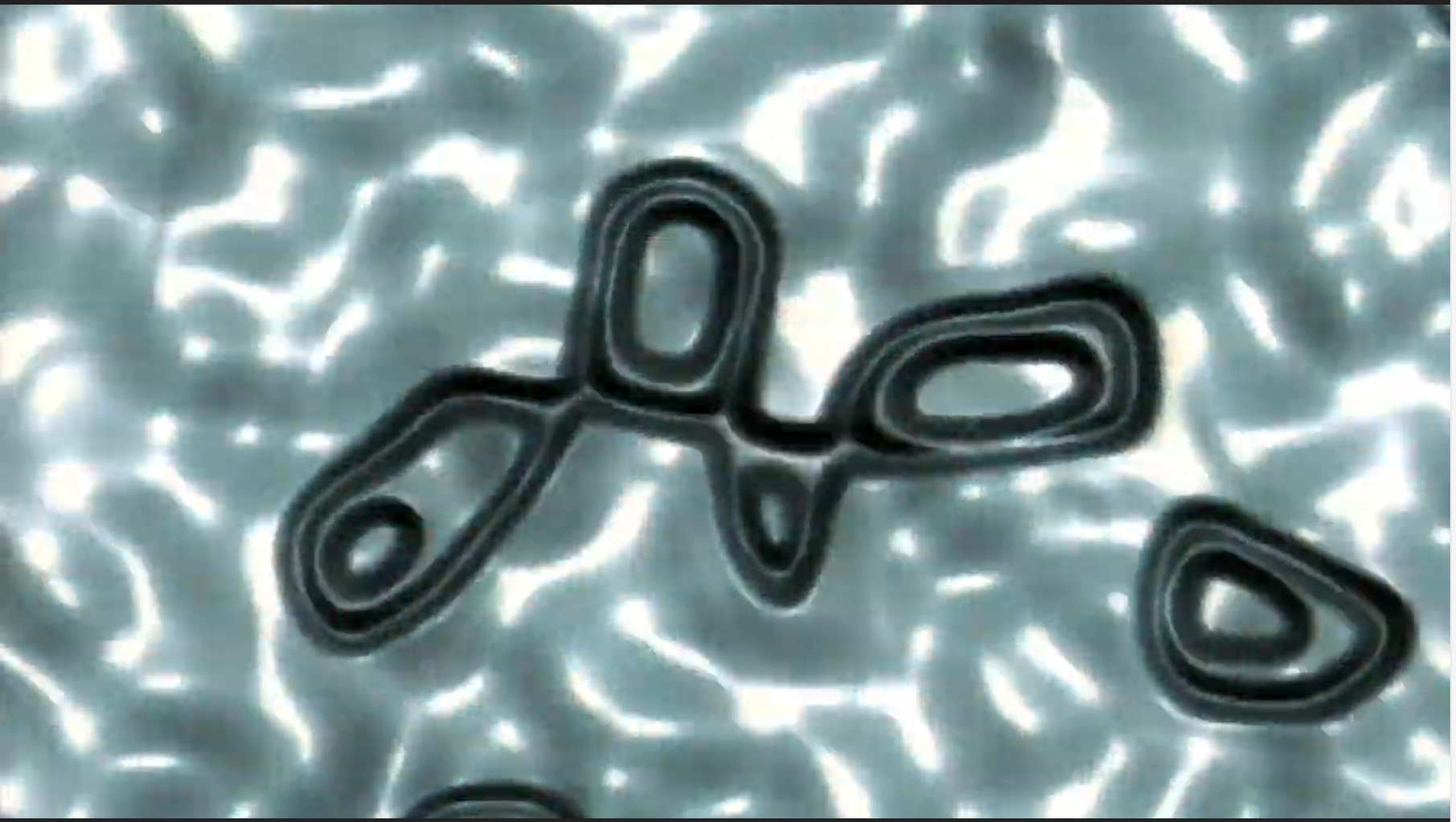
Credits

- *Mutations* - William Latham & Stephen Todd

Credits

- http://doc.gold.ac.uk/~mas01whl/index.html
- https://mutatorvr.co.uk/

1991 *Primordial Dance*



Karl Sims

Karl Sims is an American computer graphics artist. His paper *Artificial Evolution for Computer Graphics* described how he generated abstract 2D images from mathematical formulae, evolved under the guidance of a human, and used in *Primordial Dance*.

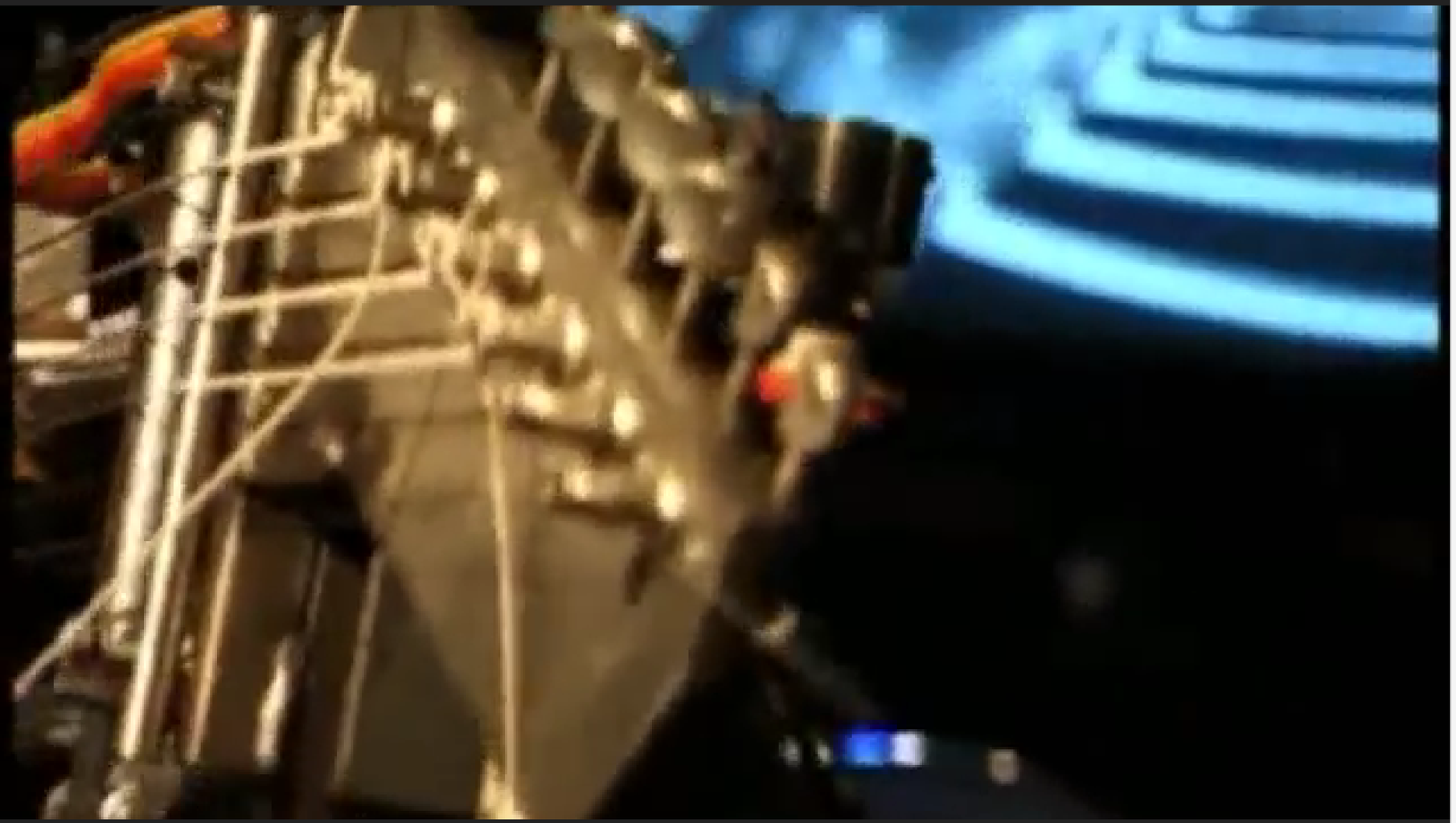
karlsims.com

Credits

- *Primordial Dance* - Karl Sims

1992

Aglaopheme



Nicolas Baginsky

Nicolas Baginsky a German artist created *Aglaopheme* in 1992, a robotic electric slide guitar that used machine learning (Self Organizing Maps) to learn from audio feedback from the environment. He has added additional robot/instruments subsequently to create *The Three Sirens* band.

the-three-sirens.info

Credits

- *Robot's Rock* - The Three Sirens by Nicolas Baginsky

1999

Electric Sheep

Scott Draves

Scott Draves is an American artist and computer scientist who created the *Fractal Flames* and *Electic Sheep* software. The latter is an ongoing project distributed computing project where users computers render the "sheep" - a genetic language for describing visuals, and can customize or breed their own. Sheep compete to survive based on participants voting.

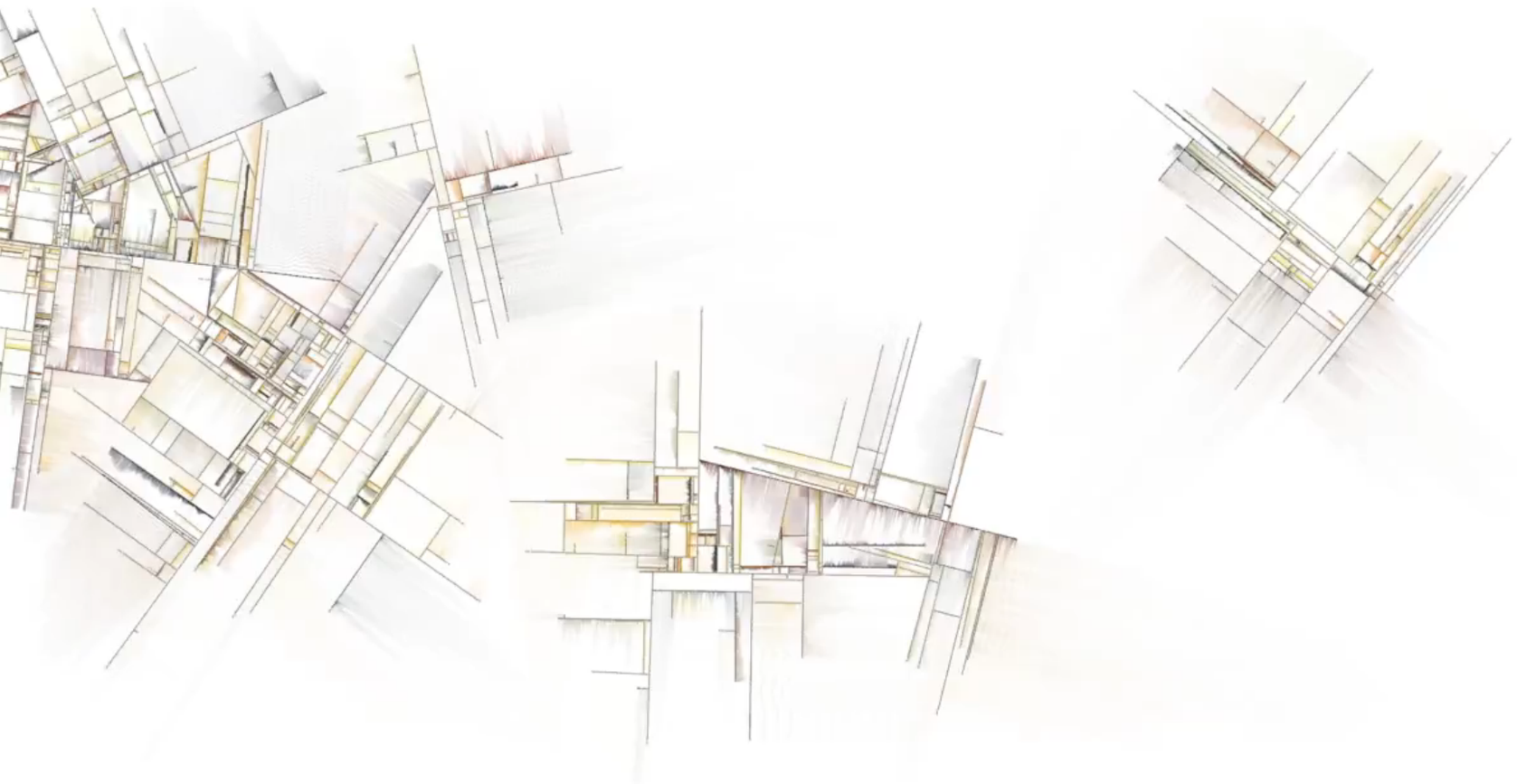
scottdraves.com

Credits

- *Electric Sheep* - Scott Draves

2003

Substrate



Jared Tarbell

Jared Tarbell is an American artist and coder that has done extensive work with generative geometric structures and is interested in the life-like emergent qualities of these systems. Live code version

infinite.center

Credits

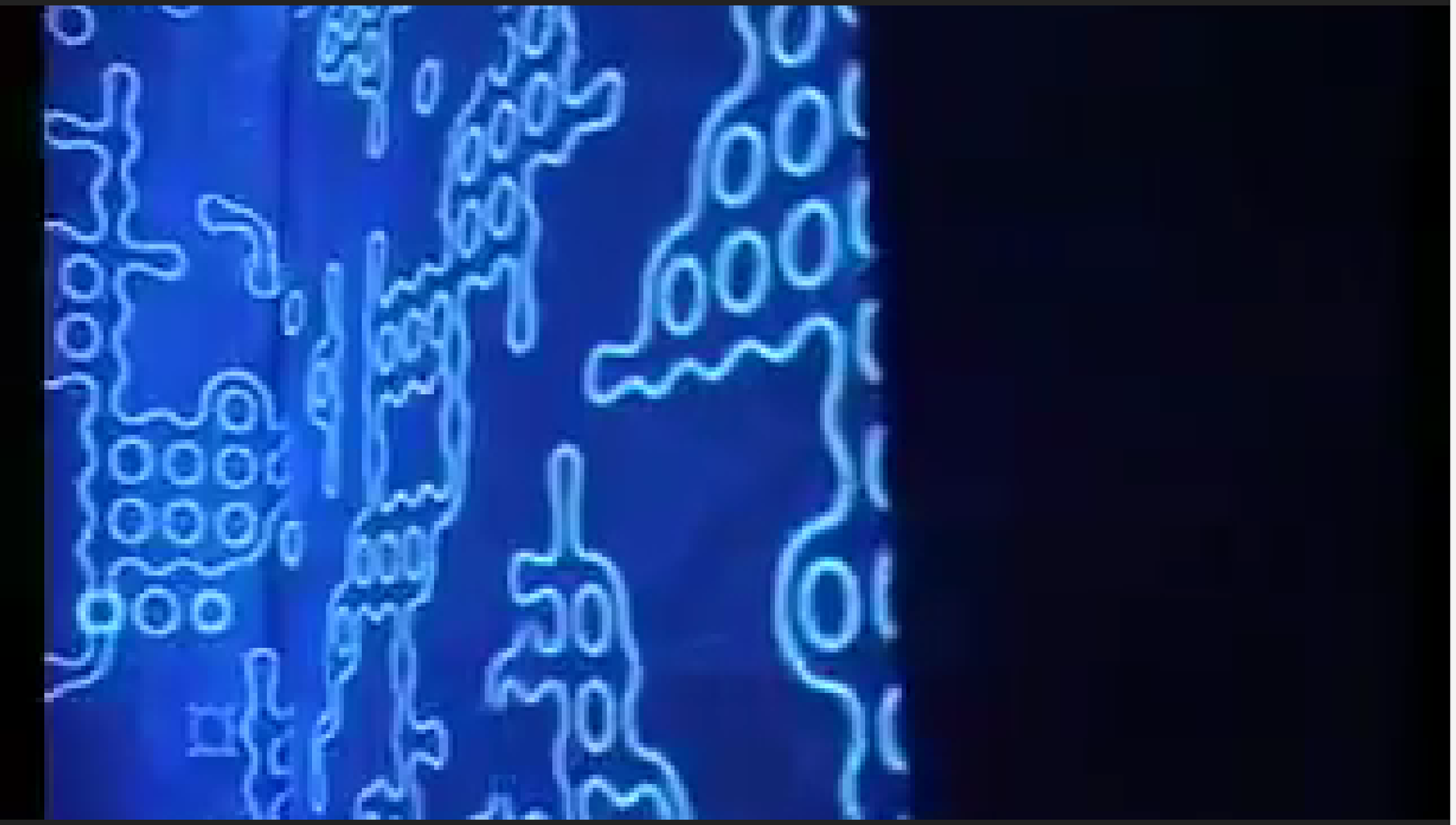
- *Substrate* - Jared Tarbell

Credits

- https://bl.ocks.org/dribnet/raw/c2d4a99516752eefa120b6b3689843f1/?raw=true by @dribnet

2004

Eden



Jon McCormack

Jon McCormack is an Australian electronic media artist and researcher in Artificial Life and Evolutionary Music and Art. *Eden* is an evolutionary sonic ecosystem where agents react to each other and a virtual environment using a set of rules encoded in binary chromosomes which can evolve.

jonmccormack.info

Credits

- *Eden, 'Evolutionary Sonic Ecosystem'* - Jon McCormack

2005

Ursonography

Jaap Blonk & Golan Levin

Golan Levin is an American new media artist and created a real-time animated subtitle system to accompany Jaap Blonk's performance of Kurt Schwitters' *Ursonate*. Using speech recognition and score-following technologies, the subtitles are sync'd to the timing and timbre of Blonk's voice.

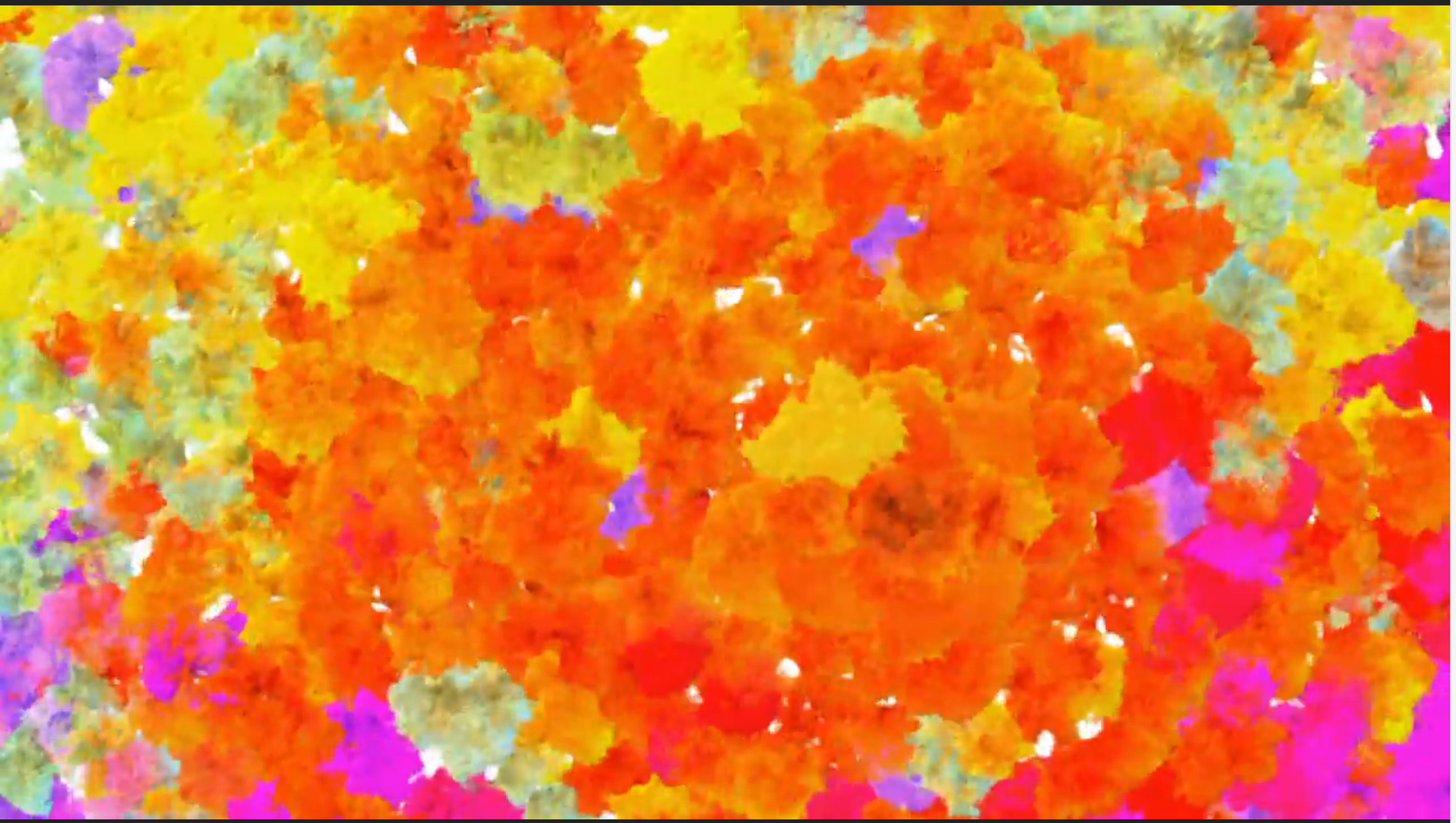
flong.com

Credits

- *Ursonography* - Jaap Blonk & Golan Levin

2006

Painting Beings



Alain Lioret

Alain Lioret is a French professor of digital art who focuses on generative art and artificial life. His *Painting Beings* fuses cellular automata, genetic programming and neural networks. The artificial bodies of paint simulated here become entangled and contact each other in "ballets of autonomous movement" to create a "full of life" painting.

alainlioret.fr

Credits

- *Painting Beings* - Alain Lioret

2011 *Plot Against Time*

David Rokeby

David Rokeby is a Canadian artist working with machine learning and video. His *Very Nervous System* in 1986 was a pioneering work translating physical gestures into interactive soundscapes.

Plot Against Time is an ongoing series of video works where the movements of certain elements remain visible, "stretching the viewer's eye across time to offer a perceptual experience of duration".

davidrokeby.com

Credits

- *Plot Against Time #4 "Atlantic Baroque"* - David Rokeby

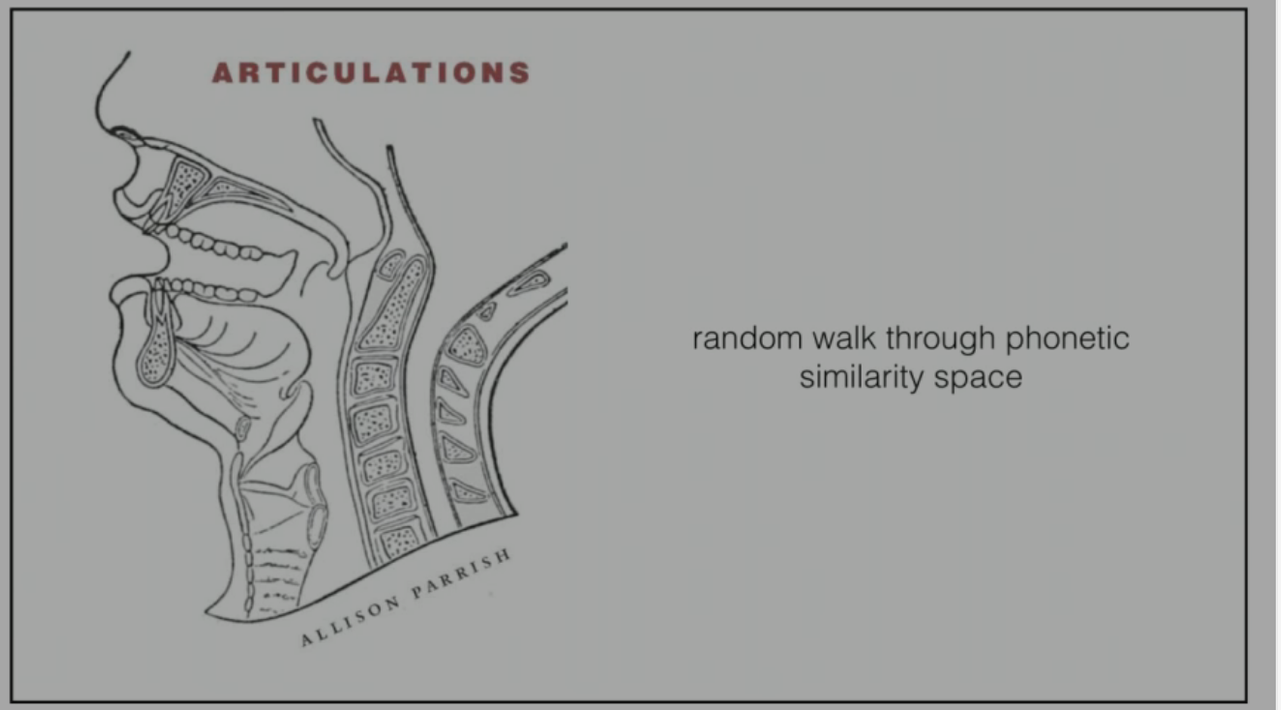
2018

Articulations

strangeloop

Sept 28-30, 2017

thestrangeloop.com



Allison Parrish

Allison Parrish is an American poet and software artist that does some of the most extensive investigation of machine learning and creative text. In her book *Articulations* she starts from a random line in the language space of all lines of poetry from *Project Gutenberg* (books out of copyright) then finds the next closest based on phonetics.

decontextualize.com

Credits

- "Experimental Creative Writing with the Vectorized Word" by Allison Parrish

The Return Of The Neural Networks

1970, 1974: **Seppo Linnainmaa**, **Paul Werbos**: backpropagation

1976: **Stevo Bozinovski** & **Ante Fulgosi**: first paper explicitly addressing transfer learning

1979: **Kunihiko Fukushima**: *Neocognitron:* hierarchical, multilayered net unaffected by shift in position

1985: **David Parker** & **Yann LeCun**: backpropagation rediscovered

1986: **David Rumelhart**, **Geoffrey Hinton**, & **Ronald Williams**: fully connected 3 layer net

1988: **Teuvo Kohonen**: Self Organizing Maps

1989: **Kurt Hornik**, **Maxwell Stinchcombe**, & **Halbert White**: MLP are universal approximators

1989: **Yann LeCun**: *LeNet5:* handwritten zip code recognition using convolutional nets

1989: **Alexander Waibel**, **Geoffery Hinton**, et al: time-delay neural networks

1989: **Peter M. Todd**: use of recurrent neural nets for algorithmic music composition

1993: **Yoshua Bengio**: speech recognition using RNNs

1993: **Hinrich Schütze**: word vectors

After the first so called "AI winter" in the early 70s in which funding dried up and research shifted to symbolic expert systems, a second AI winter developed as the hype around those systems evaporated after their boom in the early 80s.

At the same time a small group of researchers had continued neural net research and in the late 80s some breakthroughs or rediscoveries were made that led to some promising results in the early 90s.

The main advancements were:

1. Proof that multi-layer perceptrons were universal computation machines.
2. Backpropagation, a technique for training multi-layer perceptron networks.
3. Some variations of network structure that allowed for early image and audio processing.

The first handwritten digital recognition system was built as well as speech recognition.

Despite these advances, progress stalled again and other ML techniques such as Support Vector machines started to grow in popularity.

Credits

- https://www.skynettoday.com/overviews/neural-net-history
- https://bmk.sh/2019/12/31/The-Decade-of-Deep-Learning/
- https://www.historyofinformation.com/maps.php?cat=71&start=21&end=41#

TODO:

- 1990: Rodney Brooks: Elephants Don't Play Chess: introduces nouvelle AI, “individual behaviour generating modules whose coexistence and co-operation let more complex behaviours emerge”. Systems must have representations that are grounded by the physical world - “the world is its own best model”
- 1994: Jonathan Schaeffer: Chinook computer checkers software beats human. First software to win a human world championship
- 1995-2003: Brooks and MIT lab: Cog robot: pursuit of human-level intelligence using nouvelle AI principles

1979 *Neocognitron*

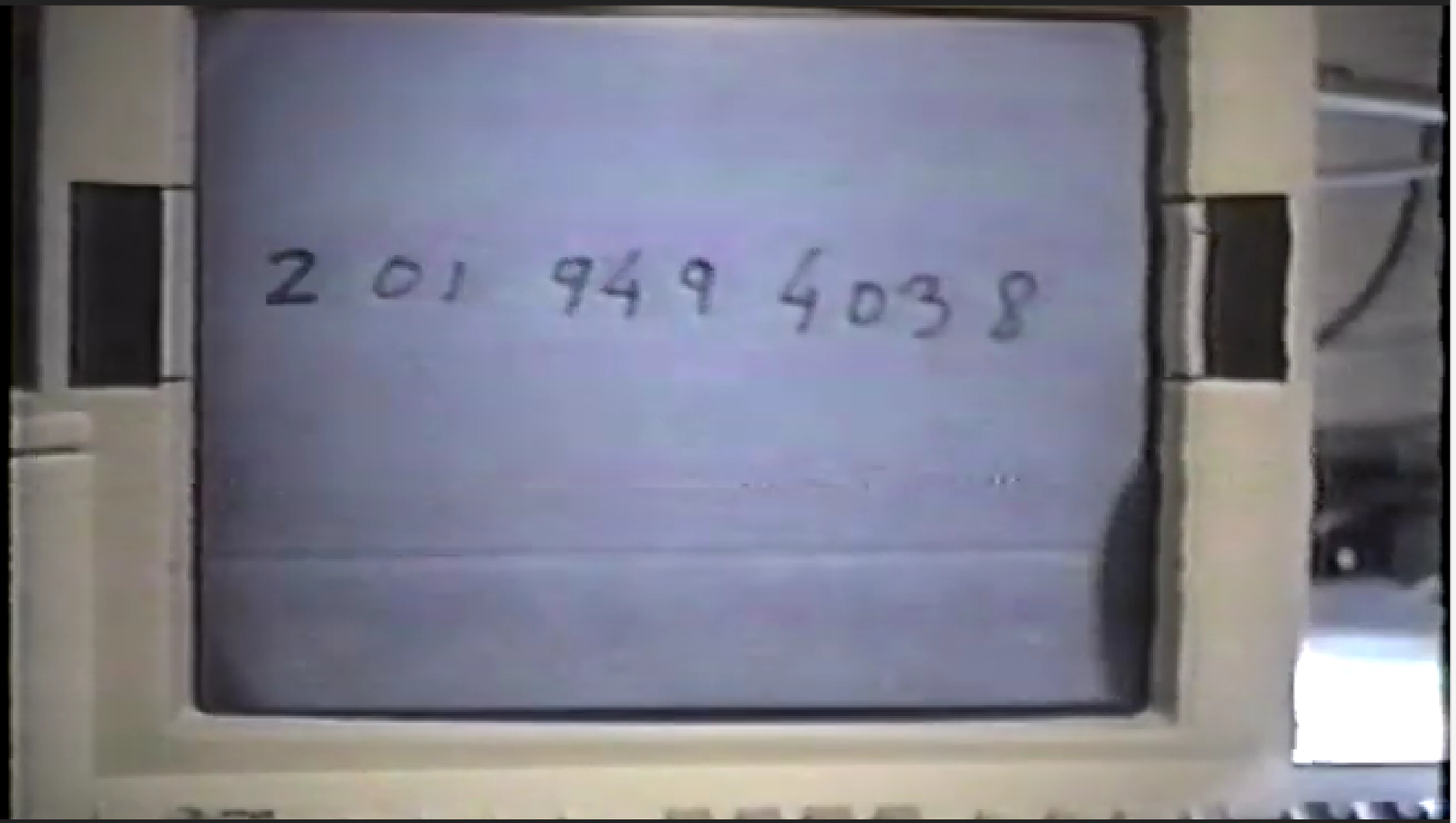
Kunihiko Fukushima

Kunihiko Fukushima is a Japanese computer scientist. *Neocognitron* was the original deep convolutional neural network (CNN) architecture. CNNs are unaffected by shifts in images making them good for character recognition.

Credits

- *Neocognition* - Kunihiko Fukushima

1989 *LeNet5*



Yann LeCun

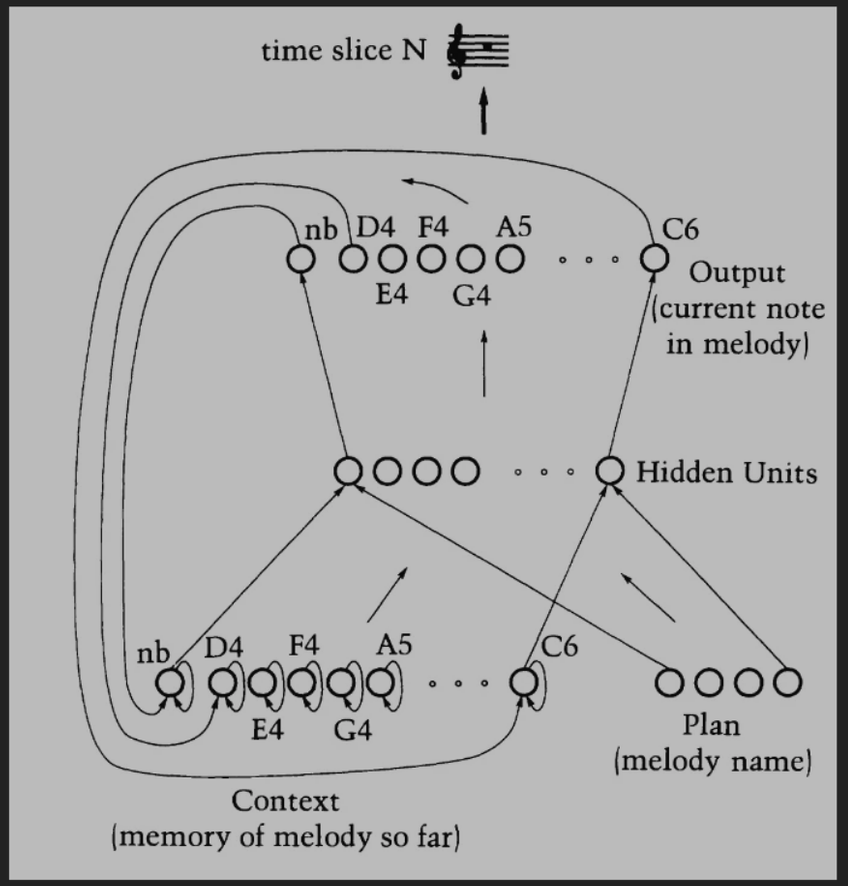
Yann LeCun is a French computer scientist. In 1989 he helped develop the *LeNet5* image recognition system that could do handwritten zip code recognition using convolutional neural nets.

Credits

- *LeNet5* - Yann LeCun

1989

RNNs For Music



Peter Todd, an American cognitive scientist, was the first to use recurrent neural nets (RNNs) for algorithmic music composition using note-by-note music generation.

Peter M. Todd

Credits

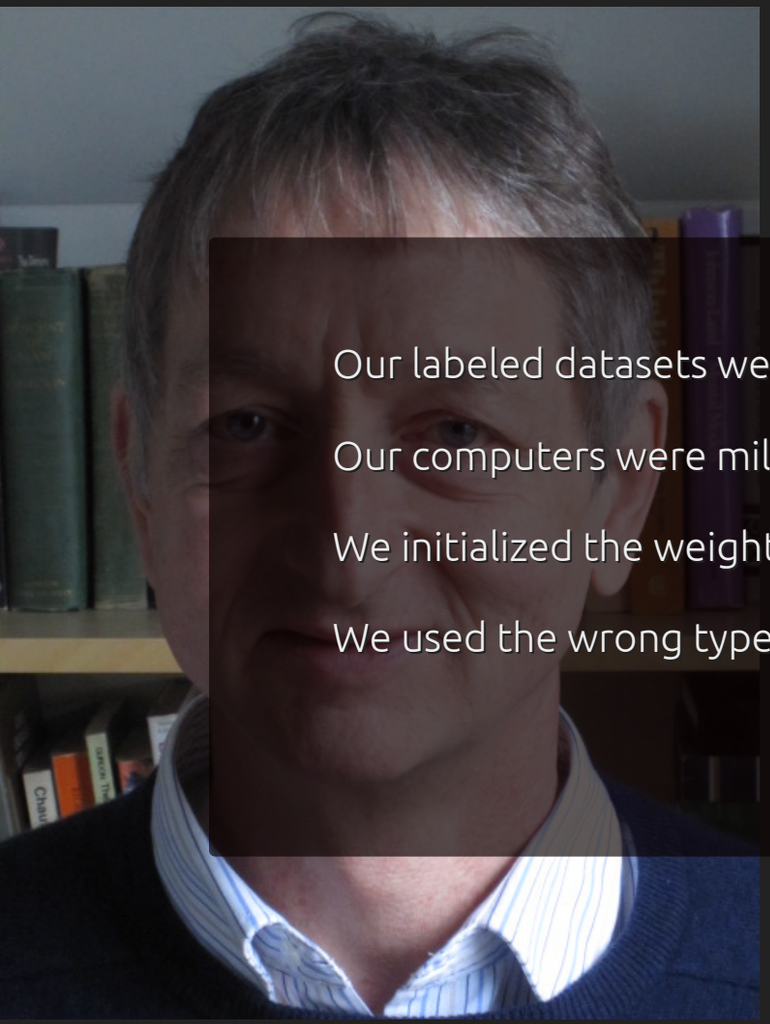
- https://towardsdatascience.com/neural-networks-for-music-a-journey-through-its-history-91f93c3459fb

As interest in neural nets faded again, a few researchers mainly based in Toronto and Montreal continued to get funding from the Canadian government. The small group were jokingly called the Canadian Conspiracy or Canadian Mafia and had rebranded neural nets as “deep learning”.

Despite the dot-com bust in 2000, internet and computer hardware were exploding in popularity. Graphics cards or GPUs, built for the massively parallel processing of 3D video games had been released add dramatically sped up training. Massive datasets like ImageNet had been created from images on the internet. ImageNet was built using Amazon's Mechanical Turk service - 50000 human workers in 167 countries, paid to clean, sort and label 1 billion images over two years.

In 2012, everything changed. Geoffery Hinton and his lab's *AlexNet*, a neural network trained on GPUs, won the Imagenet classification competition by a wide margin. Deep learning immediately came to the attention of Google and other Big Tech companies.

Hinton later described the long delayed success of neural nets this way:



Our labeled datasets were thousands of times too small. Our computers were millions of times too slow. We initialized the weights in a stupid way. We used the wrong type of non-linearity.

Now, however, the gate had been opened and deep learning was starting to be used in image, video and language research, especially at the big tech companies. These companies tried to hire every existing neural net researcher in the world to lead their new teams.

Credits

- https://youtu.be/IcOMKXAw5VA?t=21m29s
- https://youtu.be/40riCqvRoMs?t=448
- https://www.vox.com/2015/7/15/11614684/ai-conspiracy-the-scientists-behind-deep-learning

Geoffery-Hinton

Geoffery Hinton is a British-Canadian cognitive psychologist and computer scientist. In 1986 he popularized the forgotten backpropagation algorithm for multi-layered perceptrons (MLPs). In 2012 his lab's success with *AlexNet* on that year's *ImageNet* challenge was a breakthrough for computer vision that sparked the current deep learning era.

Along with Yoshua Bengio and Yann LeCun he is considered one of the "Godfathers of Deep Learning" and the three were awarded the 2018 Turing Award.

https://www.cs.toronto.edu/~hinton/

Yoshua Bengio

Yoshua Bengio is a Canadian computer scientist. He helped Yann LeCun with CNN architecture and Jürgen Schmidhuber with Long short-term memory (LSTM), an improved RNN architecture, that uses feedback loops and works well for predicting sequences.

Concerned about the social impact of AI, he actively took part in the conception of the Montreal Declaration for the Responsible Development of Artificial Intelligence. His goal is to contribute to uncovering the principles giving rise to intelligence through learning while favouring the development of AI for the benefit of all.

https://yoshuabengio.org/

Credits

- https://yoshuabengio.org/

Yann LeCun

Yann LeCun is a French computer scientist. In 1989 he helped develop the *LeNet5* image recognition system that could do handwritten zip code recognition using convolutional neural nets. He has also done extensive work on neural net optimization.

LeCun met Hinton in 1985, then spent a post-doctoral year at his lab and collaborated with Bengio at AT&T in the 1990s. In 2004 he and Hinton helped establish a program on Neural Computation and Adaptive Perception through the Canadian Institute for Advanced Research (CIFAR). Since 2014 he has co-directed it, now renamed Learning in Machines & Brains, with Yoshua Bengio.

LeCun is currently VP and Chief AI Scientist at Meta.

Credits

- https://amturing.acm.org/award_winners/lecun_6017366.cfm

Jürgen Schmidhuber

Jürgen Schmidhuber is a German computer scientist. While not part of the "Canadian conspiracy" he is also known as a "father of modern AI". He and his students developed the first LSTMs and continued to improve on them for 20 years. Schmidhuber's lab was also responsible for dramatic speedups on CNNs using GPUs in 2011, ushering in the era of GPU accelerated deep learning.

In 2002 Schmidhuber and Douglas Eck were the first to use LSTMs with music, training one to do Blues improvisation.

https://people.idsia.ch/~juergen/

The Race

2012: **Jeff Dean** & **Andrew Ng**: Google's deep learning on *YouTube* videos

2013: **Tomas Mikolov**, **Ilya Sutskever**, **Kai Chen**, et al: *word2vec*

2013: DeepMind: Playing Atari with Deep Reinforcement Learning

2014: **Ian Goodfellow**, **Yoshua Bengio**, et al: Generative Adversarial Nets (GANs)

2014: **Dzmitry Bahdanau**, **Kyunghyun Cho**, **Yoshua Bengio**: introduction of attention mechanism

2015: **Kaiming He**, **Xiangyu Zhang**, **Shaoqing Ren**, **Jian Sun**: *ResNet:* Residual Block Architecture

2015: **Alexander Mordvintsev**: Google's *DeepDream*

2015: **Leon Gatys** et al.: image style transfer

2016 - 2019: DeepMind: *AlphaGo* > *AlphaGo Zero* > *AlphaZero* > *MuZero* & *AlphaStar*

2017: DeepMind: *Attention is All You Need:* Transformer architecture introduced

2018: OpenAI: *GPT*: transformer language model with 150 million parameters

2018: **Tero Karras**, **Samuli Laine**, **Timo Aila**: *StyleGAN:* high resolution image GAN and *CelebA-HQ* dataset

2018: **Phillip Isola** et al.: *pix2pix:* image-to-image translation

With money pouring into the field, competition was fierce, and existing tech companies swallowed start-ups and bought up as much talent as they could. Two notable groups; Deepmind, acquired by Google in 2014 and OpenAI, originally set up by Elon Musk in 2015. Musk was worried about the longterm effects on AI by the no longer "Don't Be Evil" Google and the obvious amorality of Facebook. Musk has since split from OpenAI.

Fortunately an ethos of opensource software had already taken hold in much of the younger generation of researchers and despite the major holdouts of Deepmind and OpenAI who often don't release their code, a majority of published ML research comes with source code. This openness in code and access to research papers sparked a flood of entrants, including artists, from all over the world.

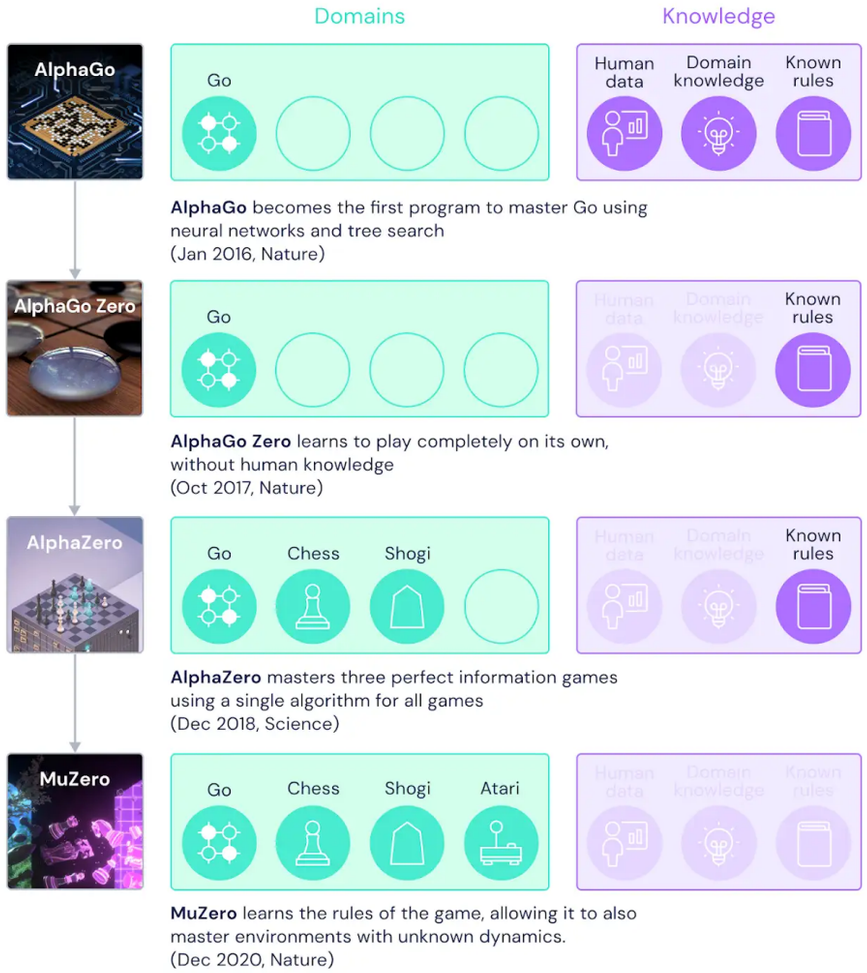
Deepmind's *Dreepdream* was notable for its effect outside the research community. The technique was heavily promoted by Google for marketing reasons, but the software itself was available for full source code and started Google's promotion of AI & art and relatively free sharing of its compute resources to individual experimenters that continues to this day. The *Deepdream* technique itself inspired many visual artists to engage with machine learning as a viable art form - the insight from the inversion of the network - instead of classifying images, pushing the classification into the world had a mystical and very real symmetry with human cloud watching and hallucinations.

Credits

- *The Gate to a Deep Dream* - DDG Generator

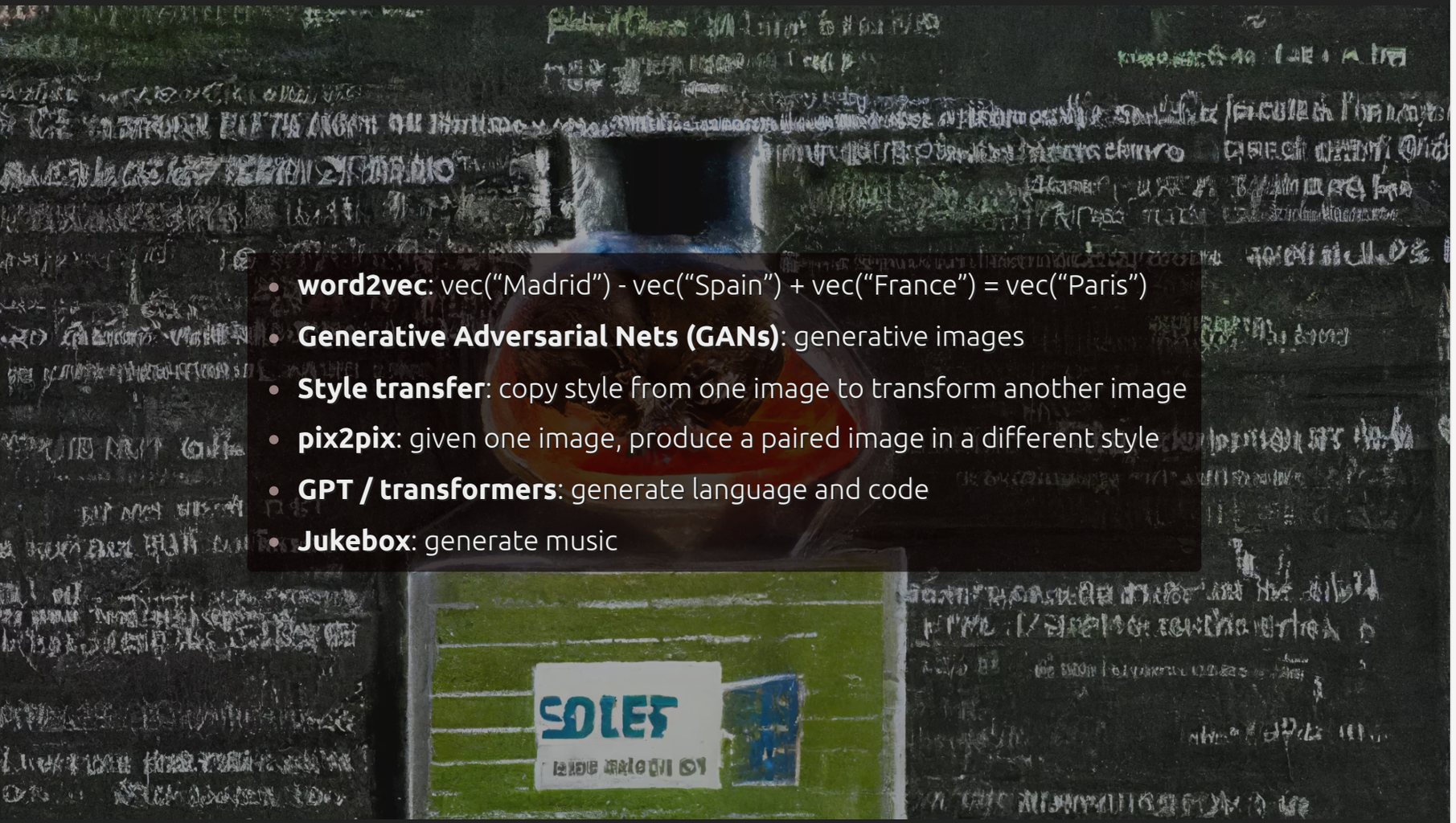
Credits

- https://ai.googleblog.com/2015/06/inceptionism-going-deeper-into-neural.html



Deepmind's Alpha game playing systems also sparked public imagination. The 4000 year old game of Go had been considered an unsolvable task by computers because the techniques for searching for the best moves in the future playspace that had been successful in chess were computationally impossible in Go because of the much larger set of possible moves. Computers would have to play by "intuition", as often Go playing was described by the greatest human players. Deepmind's approach was to use reinforcement learning to narrow the search space, to only explore the most promising future playstates, a form of machine intuition. AlphaGo's defeat of Lee Sedol, considered one the best current players was emblematic of what narrow machine intelligence can achieve. Both AlphaGo and Lee Sedol were said to have performed historically important moves to win their games, but AlphaGo won 4 of 5 games. A year later AlphaGo Zero trounced AlphaGo 100-0, this time learning the game from scratch, with no human play examples. A year later, AlphaZero was the new champ, and learned Go, chess and shogi. Finally, in 2019 MuZero was more capable still and could also play 50 Atari games by discovering for itself how to build a model of the game and understand it from first principles. When the world model the AI needs to learn is limited to the complexity of a game, even one as complicated as Starcraft, we now have enough techniques to train to super human levels.

During this time a number of models were developed that have relevance to artists:



- **word2vec**: vec(“Madrid”) - vec(“Spain”) + vec(“France”) = vec(“Paris”)
- **Generative Adversarial Nets (GANs)**: generative images
- **Style transfer**: copy style from one image to transform another image
- **pix2pix**: given one image, produce a paired image in a different style
- **GPT / transformers**: generate language and code
- **Jukebox**: generate music

word2vec allowed for words and concepts to be combined and substracted from each other. Generative Adversarial Nets or GANs became very popular for generating images. Style transfer for images and videos allowed you to copy or mimic the style of one image, say an impressionist painting, to your own photograph. pix2pix trained on image pairs allowed for real-time video transformations from one set of images to another. GPT and other transformer based models could generate and summarize text. Jukebox allowed for music generation.

We'll discuss these tools in more depth later.

INTERMISSION

Go have a break!

We've covered the **Past**, next up is the **Present**!

Other History Resources

- Jeremy Norman’s HistoryofInformation.com
- Andrey Kurenkov's A Brief History of Neural Nets and Deep Learning
- Thomas Dreher's History of Computer Art
- Marnie Benny's Timeline of AI Art
- Veronika Gladchuk's The History of Machine Learning: How Did It All Start?
- Simon Crab's 120 Years of Electronic Music

Credits

- *Jazz, in the style of Ella Fitzgerald - (Baby shark)* - OpenAI Jukebox

Credits

- *How Not To Be Seen - A Fucking Didactic Educational .MOV File* - Hito Steyerl

We are currently in the era of Deep Learning and Big Data with a rapid proliferation of machine learning into all areas of industry. Industry estimates are for approximately 350 billion dollars of spending on machine learning in 2021. Big Tech leads the way, funded by advertising and data harvesting or user surveillance based business models. The two main rivals, Meta (Facebook) and Alphabet (Google), each control the two most popular opensource deep learning toolkits.

We are in a difficult era for business, or any large modern organization, where the speed of technological development out paces the time to required to integrate that technology. By the time a large organization completes implemention of a technology the next generation is available. The dominant strategy in this position is monopoly or oligopoly - without competitors there is no one to leap frog you - and this strategy has been pursued aggressively since at least the Reagan era.

Cory Doctorow ascribes these monopoly positions as the root cause of corporate surveillance and other malfeasance, but it is also important to note that Google and Facebook had to find a sustainable business model in an environment where mainstream media, especially journalism and television, had already become almost entirely dependent on advertising. They chose the only option available; the normalized practice of for-profit mass manipulation. Eventually, they were able to out compete other advertising mediums because digital medias ease of tracking and storing data.

Credits

- https://aimagazine.com/ai-applications/ai-spending-will-reach-usdollar342bn-2021-says-idc

In 2013 Edward Snowden revealed thousands of US National Security Agency (NSA) documents that detailed a global surveillance campaign lead by the US but in cooperation with UK, Australian, New Zealand and Canadian intelligence agencies. This included direct access to Google and Yahoo email accounts, tracking cell phone locations, phone records, and mass internet data surveillance. 35 world leaders, including the German Chancellor, were being spied on. The NSA's stated objective was to "Collect it All," "Process it All," "Exploit it All," "Partner it All," "Sniff it All" and "Know it All."

They aren't alone. Facebook, now Meta, has built the world's leading social media empire by recording as much as possible from their users, including what sites they visit outside of Facebook. Their business model is essentially to sell the exploitation of this information, which they call targeting, to the highest bidder. The world's most effective propaganda network isn't state-owned, it is available to anyone with enough money.

Surveillance and propaganda, a historically popular combination in use by the US and others to overthrown or destabilize governments, have been woven together even more tighter by machine learning. Facial and gait recognition, emotion or sentiment detection, tracking what users watch and read, and their movements through their phones, combines with machine learning powered content recommendation engines and advertising to create an infoscape tailored to how you think. Quite explicitly they sell access to those whose thinking can be shifted most profitably for the buyer.

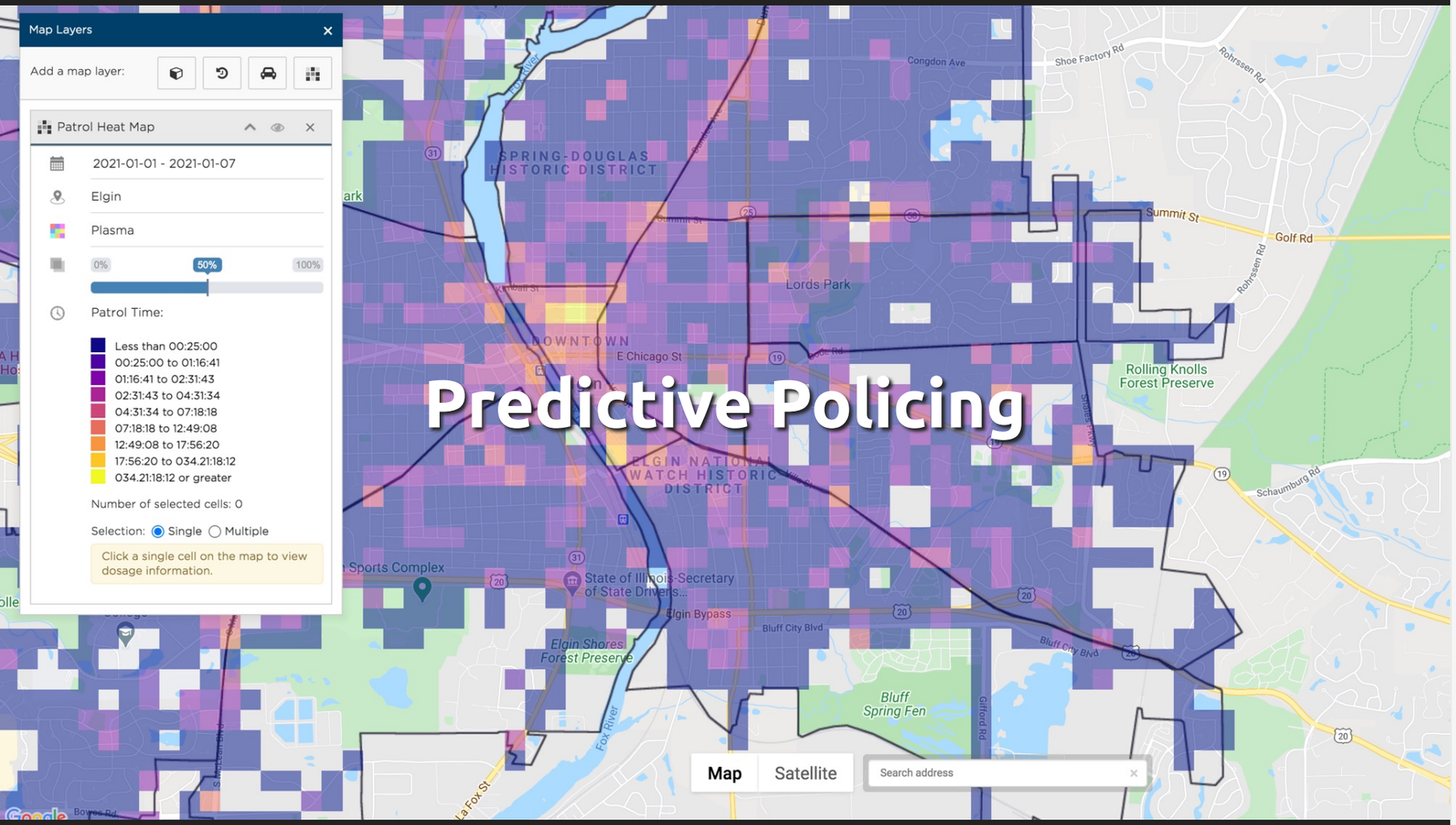
The normalization of for-profit manipulation successfully instituted during the 20th century birthed the surveillance capitalism of the 21st. What began as general profiling of social groups, called market research, has evolved into even more profitable manipulation as the recording of and inappropriate access to all private data becomes routine and required for use of online services.

We are all becoming targets. A growing number of entertainment industries have begun whale hunting. Whales, in this case, are people who can be convinced to spend thousands of dollars on products that are sometimes completely virtual, and often cannot be resold, having no real value. Historically whale hunting was popular in gambling, luxury brand, and grocery markets, but this is shifting as more business happens in a digital environment. Interactive digital experiences, such as games, can exploit dopamine addiction and sell copies of virtual goods that cost them nothing. Other forms of whaling include reactive or individualized pricing, especially if crafted to exploit individual weakness. Whaling can lead to targetting those least able to make self-benefiting financial decisions, including children. To an extent all advertising works similarly, particularly in light of the absence of freewill and the general lack of any real world benefit from purchasing digital goods that are intrinsically free to copy.

Non-fungible tokens or NFTs use similar strategies, where people are literally buying the right to sell the NFT to someone else, meanwhile the NFT creator takes a cut of each sale.

Machine learning is accelerating the growth and effectiveness of whale hunting. The effect is most strongly seen in digital industries, but the essential nature of this approach is to leverage information to maximize profit regardless of the cost to individuals. There is little "consumer solidarity", instead the whales are happily sacrificed so the smaller fish get cheaper goods and services.

But first they came for the whales.



Data analytics is being sold to police services as well, in their own hunt for criminal whales. On the face of things, this seems potentially effective, as 1% of the population is estimated to account for 60% of all violent crime.

But what data is being used in this hunt? Existing crime data? This is not foolproof, as historical data might not capture the actual crimes being committed. Adding machine learning does not make decisions about data objective, despite any marketing claims from companies like Predpol, derived from "predictive policing", and now rebranded as Geolitica. Part of the pitch of these systems is that "math doesn't lie" and machines compensate for human bias.

It is estimated that only 40% or less of crimes are reported in general, so if that 40% isn't a representative sample then the machine will learn a biased prediction. Findings have shown lower self reporting among White and affluent crime victims, and crimes reported by the police are heavily skewed towards where the police were patrolling, i.e. "street crime", making it extraordinarily likely the dataset is biased.

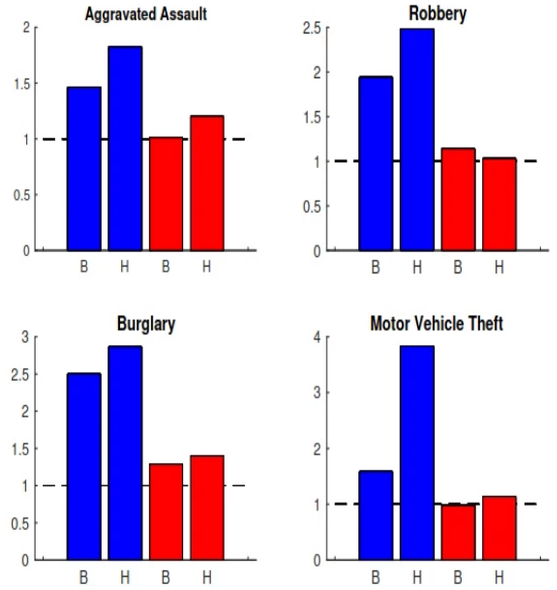


Fig. 3: Patrol levels per individual of Black and Hispanic populations relative to white populations (dashed line) for the neutral (blue) and fair (red) Hawkes processes.

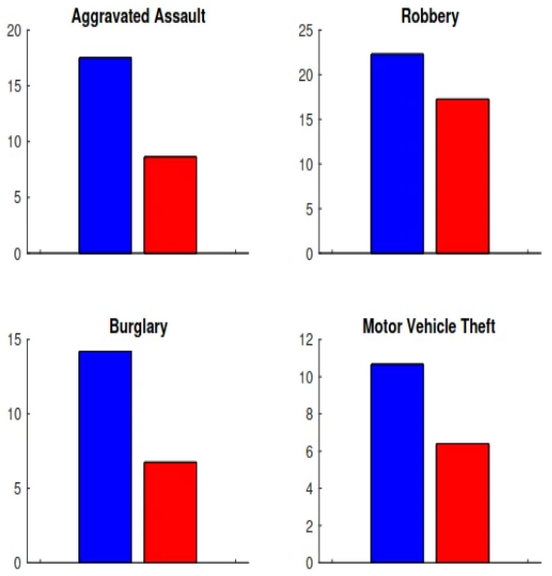
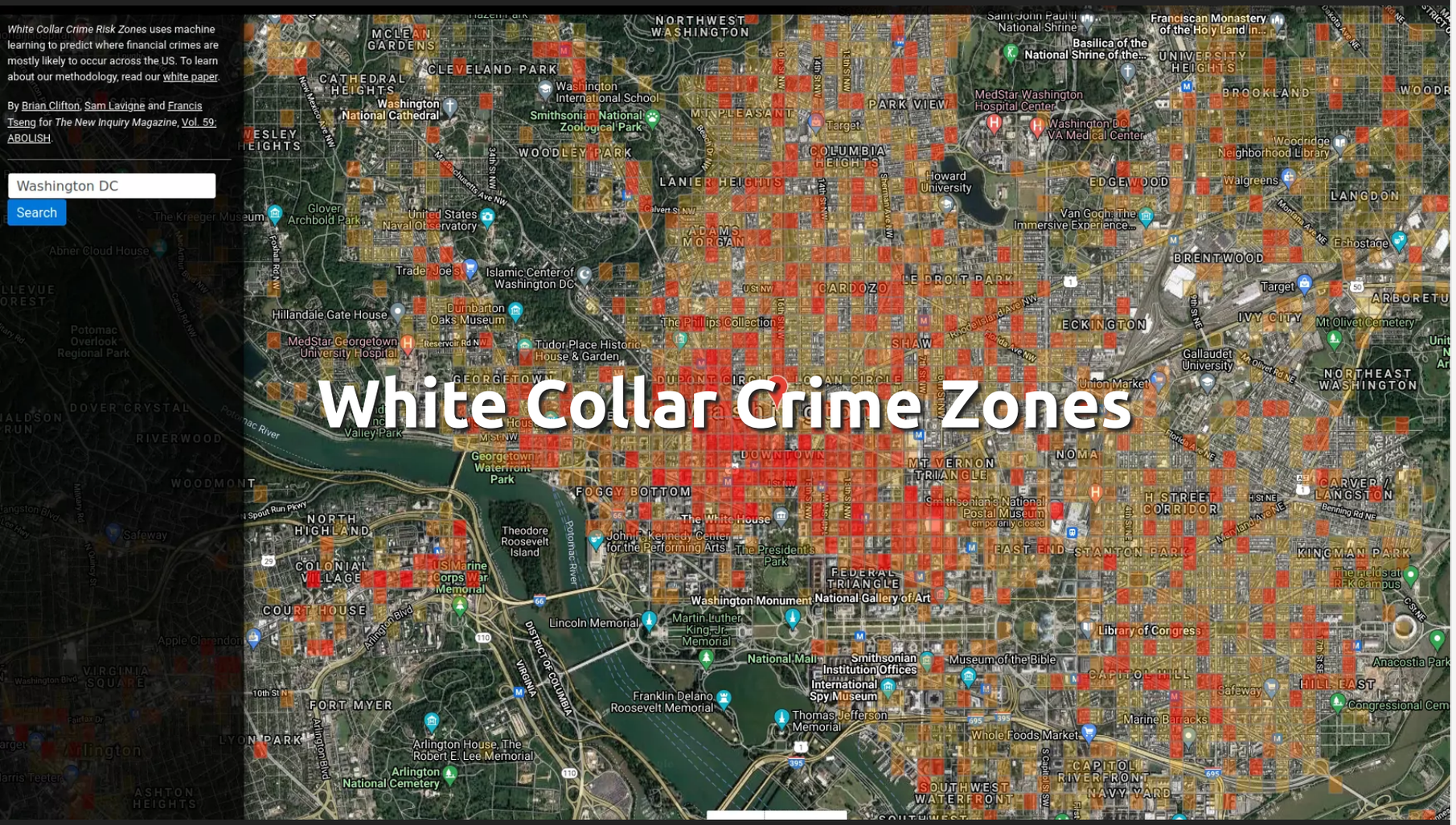


Fig. 4: Predictive accuracy index (PAI) for the neutral (blue) and fair (red) Hawkes processes.

Mohler, G., Raje, R., Carter, J., Valasik, M., & Brantingham, J. (2018, October). A penalized likelihood method for balancing accuracy and fairness in predictive policing. In 2018 IEEE International Conference on Systems, Man, and Cybernetics (SMC) (pp. 2454-2459). IEEE. https://doi.org/10.1109/SMC.2018.00421

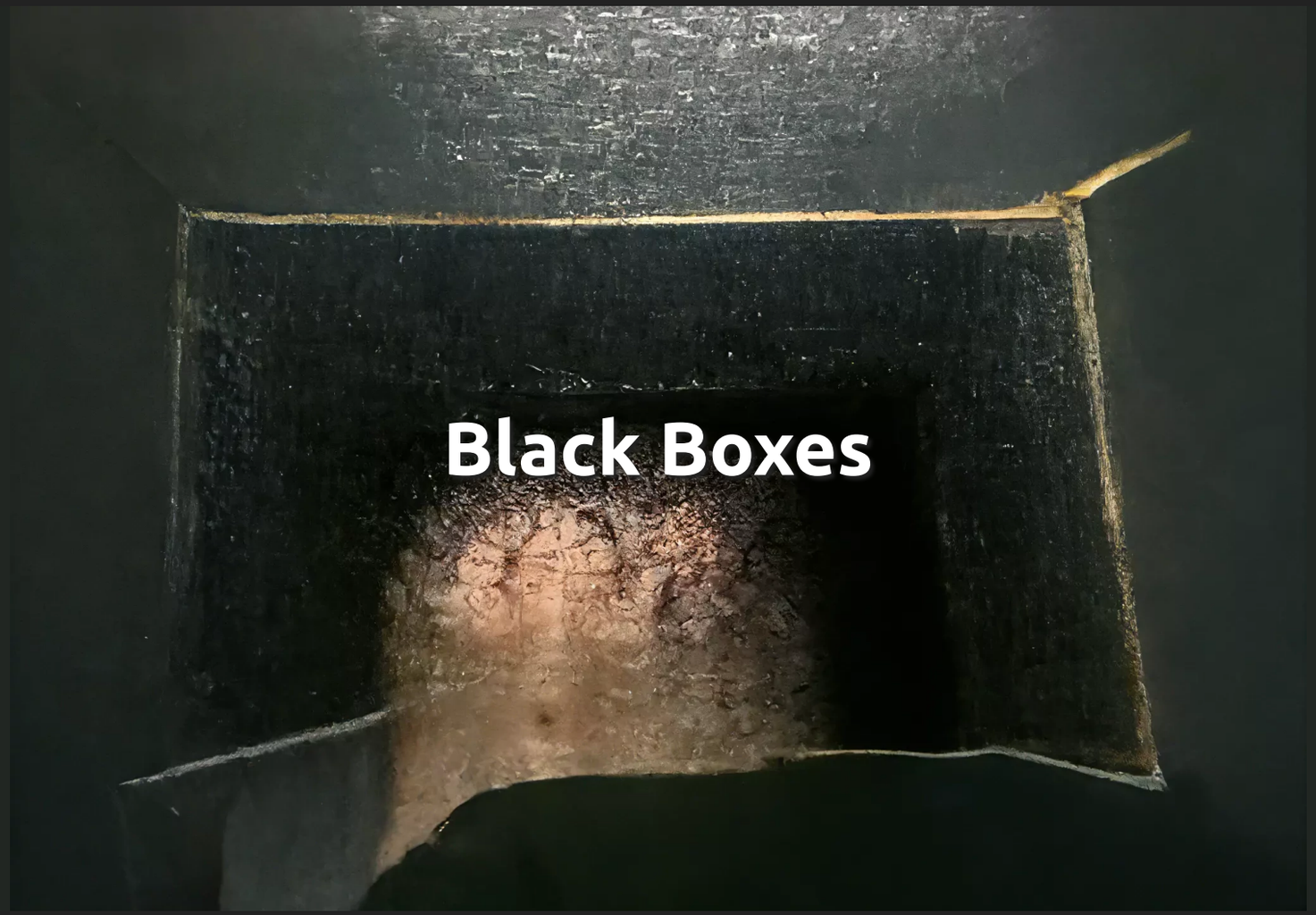
There was a 2018 academic study of the PredPol algorithm which acknowledged the bias, and studied the effects of a more even distribution of crime predictions. But they found the software predictions were less in line with later crime reports, making it less accurate than the original algorithm and PredPol didn't adjust its software. If the goal is only patrolling in areas of greatest street crime, this approach may be reasonable, but questionably provides any new information.



An interesting counter example, is White Collar Crime Zones that shows a similar display to PredPol's but instead uses financial crime data to illustrate reversing the biases. In this example the underreporting is likely far higher in areas outside regulated financial sectors.

Credits

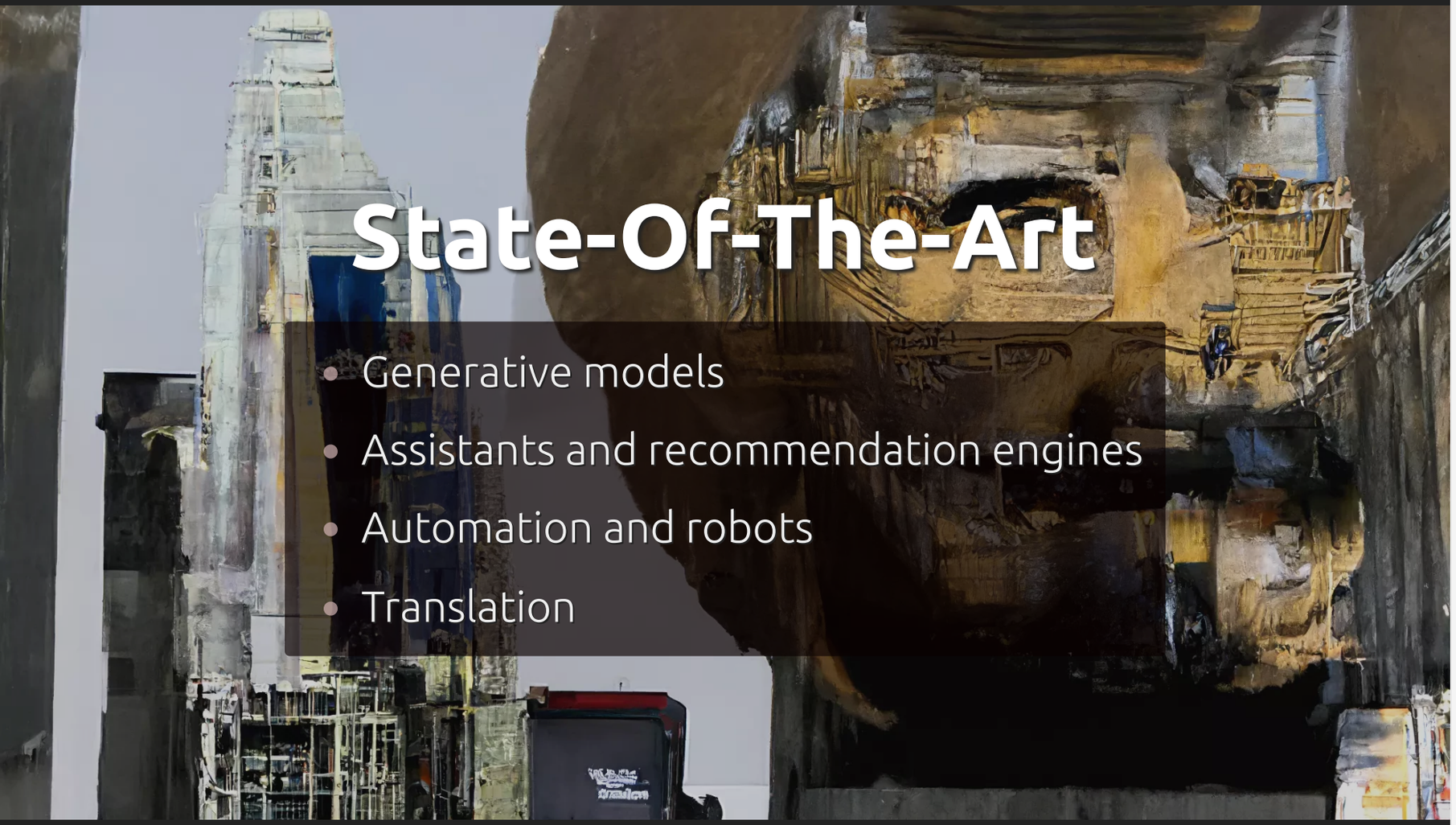
- White Collar Crime Zones



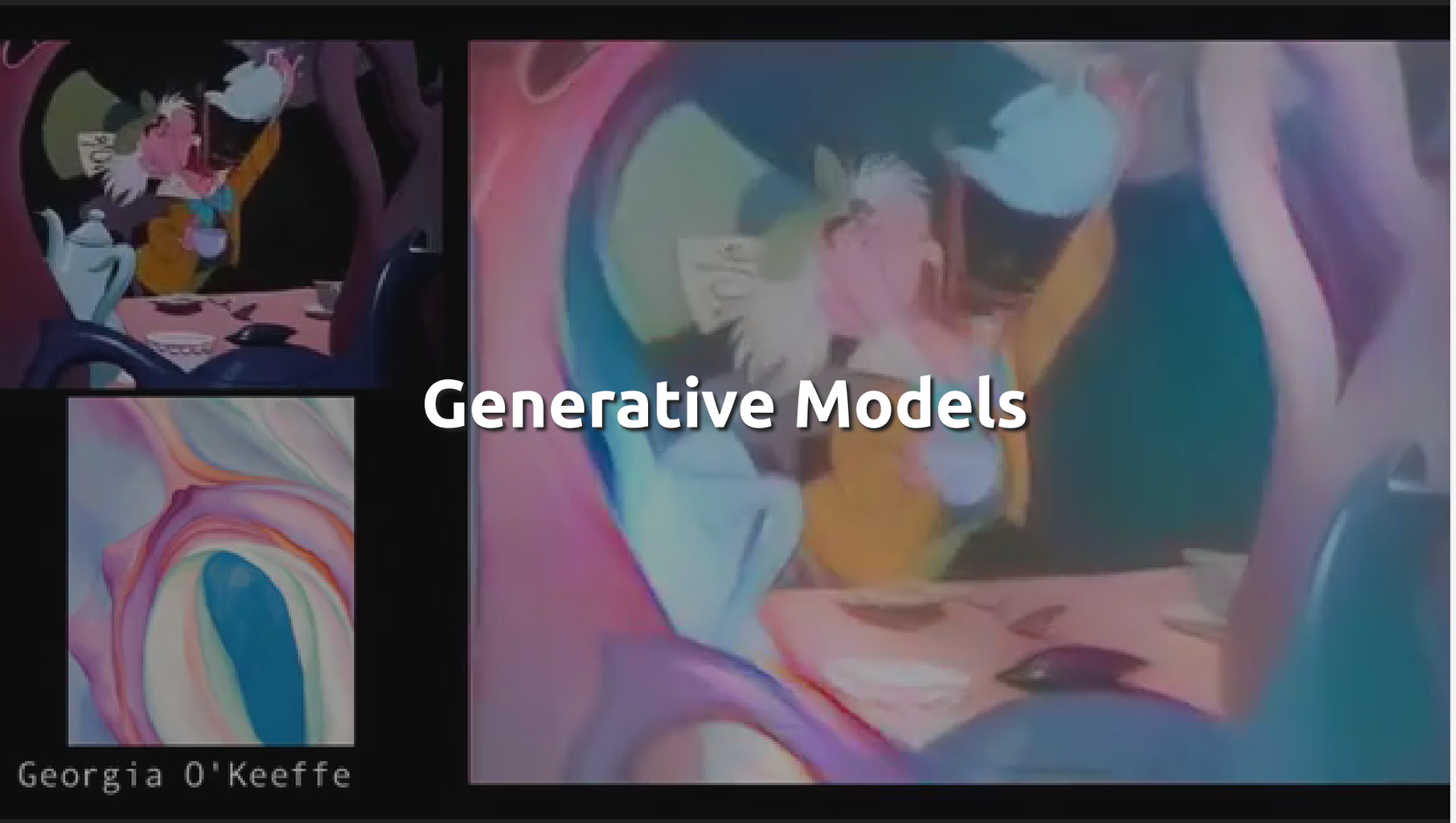
A more general problem with most commercial AI services is that they exist in a black box protected by trade secrets and have little public scrutiny. At minimum police organizations need access, but due to the impact of the predictions on the public, it seems reasonable to allow or require public investigation. Making data objective is impossible, debiasing data is difficult, but recording, inspecting and testing the predictions of machine learning systems is far easier than predictions made by humans.

Credits

- https://pluralistic.net/2021/12/02/empirical-facewash/#geolitica
- https://www.mic.com/articles/156286/crime-prediction-tool-pred-pol-only-amplifies-racially-biased-policing-study-shows
- https://themarkup.org/prediction-bias/2021/12/02/crime-prediction-software-promised-to-be-free-of-biases-new-data-shows-it-perpetuates-them
- https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3969807/
- https://ieeexplore.ieee.org/abstract/document/8616417



To investigate the intersection of mainstream Western capitalism, current art practice and machine learning we'll take a look at the state-of-the-art systems in use today.



Generative machine learning models are capable of creating audio, visual and text either randomly or with some amount of control. They excel at mimicry and pattern matching. In general there hasn't been direct commercial use for them yet, although a number of smaller start-ups are trying to monetize their use, generally as an alternative to hiring relatively expensive artists or to create training data for other ML models. Research in generative models is often to better understand computer vision, but is also driven by a strong interest in machine creativity and exploring what is possible.

Competition and experimentation is driven by standard datasets and comparison functions, such as the ImageNet dataset and various functions that compute the differences between real and generated images. This in additional to more tradition subjective human testing often carried out through Mechanical Turk.

All generative models currently exist in a space where if the interaction is brief, highly constrained, or accepting of a highly subjective, almost dream-like output, then the model can be useful. At the end of 2021 some limitations are starting to be pushed back, particularly for image generation. But as of yet no generative models have a good understanding of the relationships between elements in the generated whole as they continue to lack any conceptual model of the real world.

The state-of-the-art changes monthly, and thus, like the businesses trapped in loops faster than they can integrate, almost all but the most popular tools go mostly unexplored by artists.

Let's take a look at a few of the most popular generative models used in 2021.

TODO

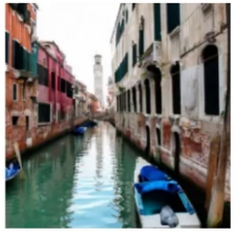
- Neural visual grammar
- Jukebox

"a hedgehog using a calculator"

"a corgi wearing a red bowtie and a purple party hat"

"a surrealist dream-like oil painting by salvador dalí of a cat playing checkers"

"a professional photo of a sunset behind the grand canyon"



"a boat in the canals of venice"

"a painting of a fox in the style of starry night"

"a high-quality oil painting of a psychedelic hamster dragon"

TEXT2IMG

- CLIP
- DALL-E
- GLIDE
- GauGAN 2

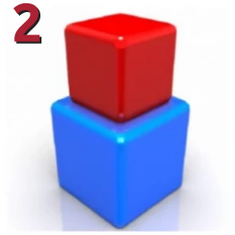
"robots meditating in a vipassana retreat"

"a fall landscape with a small cottage next to a lake"

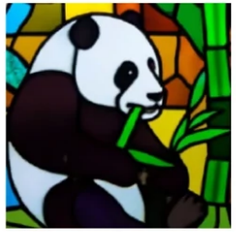
"a crayon drawing of a space elevator"

"an illustration of albert einstein wearing a superhero costume"

"a futuristic city in synthwave style"



"a red cube on top of a blue cube"



"a stained glass window of a panda eating bamboo"

"a pixel art corgi pizza"

All of these projects take text prompts as input and then output generate images. Images to act as the origin point to guide the generation can also be supplied and in the case of GauGAN, you can also paint a sketch and modify it to update the generated image.

Each project has different strengths and weaknesses, but in general are trained from a dataset from the internet that connects images and text, for example, it could be the text from the same page an image was found. After training the model is able to generate an image that has some relationship to the text.

The interface to these models is quite novel - a short prompt, with some surprising characteristics. This comes with the regular caveats - the text is based on English and is trained from biased public internet sources. There is a growing practice of "prompt engineering" that is learning the quirks of these sorts of models and there is too much to cover here but I'll give a few examples of the strangeness.

First, scene descriptions work poorly beyond single interactions or relationships. Specifying numbers of things, their spatial relationship or interactions is unlikely to work well. Global image composition and awareness of all the elements in the image is very poor. Single objects and even abstract concepts can work well, and even emoji can work.

Other prompt engineering tricks have bizarrely strong effects:

apple

the golden apple of my eye

a photo of an apple

a bad photo of an apple

an apple #artstation

a highly detailed apple rendered by unreal engine

The choice of subject, apple, is bad here, it doesn't have enough description and it is too concrete, there is a fine balance between definite and subjective in the best prompts. So all of these apples will be less interesting than say, "the golden apple of my eye".

"a photo of an apple" sadly, including the starting "a" and "an" seems to work better sometimes and the end result will look more like a photo than say a painting.

You can get interesting effects by saying things like "a bad photo of an apple".

Adding #artstation to the end of a prompt encourages a more painterly look reminiscent of concept paintings for films and video games - because artstation is a website featuring concept art.

"rendered by unreal engine" added to the end leads to more detail, bokeh and higher image fidelity as unreal game engine is known for it's high quality rendering. Adding "highly detailed" also helps the model add fine details.

Despite all that the results can be pretty spectacular, especially for creator who are not expert visual artists, it can provide a tool that produces interesting images on demand, as long as you are willing to explore the bizarre space of prompt engineering.

For example, I have used generative tools to create this and the other imagery you see in this tutorial.

Credits

- https://arxiv.org/pdf/2112.10741.pdf

StyleGAN is one of the most well known GAN-based image generators, particularly for its ability to generate high resolution faces.

StyleGAN is currently on version 3, which address an interesting limitation of version 2 where the details of the faces could be locked into place despite different orientations of the face.

This is a great model for artists, but requires a lot of training on high-end hardware still. Generally the training starts from an existing pretrained model and then just shifts it to the target dataset, so this requires finding an existing model that is somewhat similar to the images you want to work with.

Like other generative image methods previously, animating the generated output of styleGAN has a particular aesthetic to it that recalls "morphing" software. However, unlike previous morphing techniques each generated image from these models will be a valid representation, as unlike morphing, any point between any two faces is always just another valid face. We'll talk more about how this work in the next tutorial.

Credits

- *StyleGAN3* - NVidia

Mostly successfully used by Memo Aktin in his Learning 2 See projects, the pix2pix algorithm was created in 2017. The GAN learns a mapping from input image to output image. This approach can colorize images, convert day to night, and other more artistic mappings such as eye to nebula or drawing of a castle to photo of a castle. This creates a model that converts everything it "sees" into what has been trained to see.

This technique is particularly fun to use on modern hardware where it can be run in real-time. There is a certain magic in finding the right representation to map from. For example, it took me quite some time to find that scattered sesame and sunflower seeds made for good input to generate images of treetops.

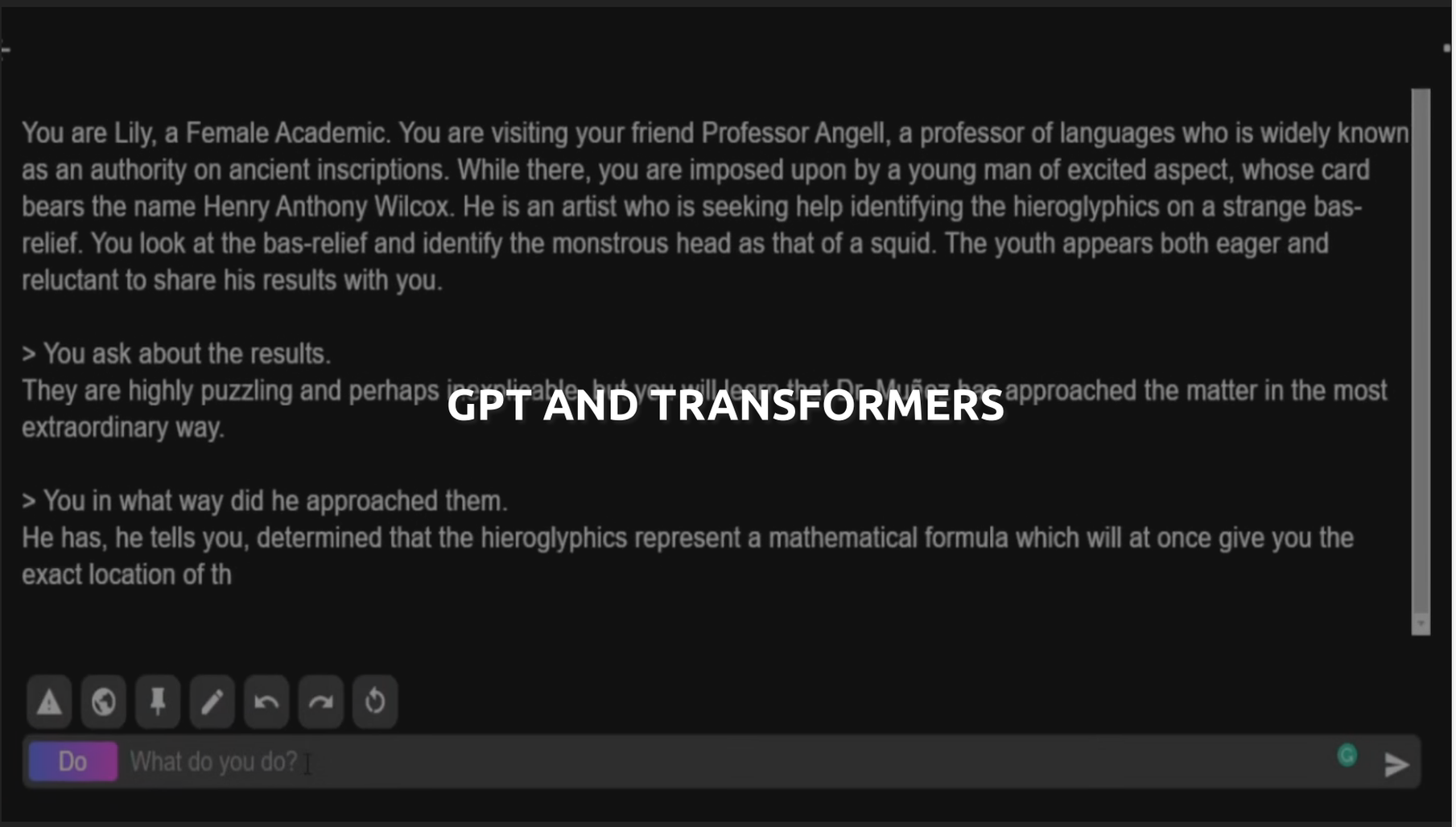
Cut: Generally in those cases you train on target images and processed versions of those images containing only edges. Then you can convert other images or video of a completely different subject to only edges and the model will convert those edges to the trained image.

Credits

- *Learning to See* - Memo Akten

Credits

- https://phillipi.github.io/pix2pix/



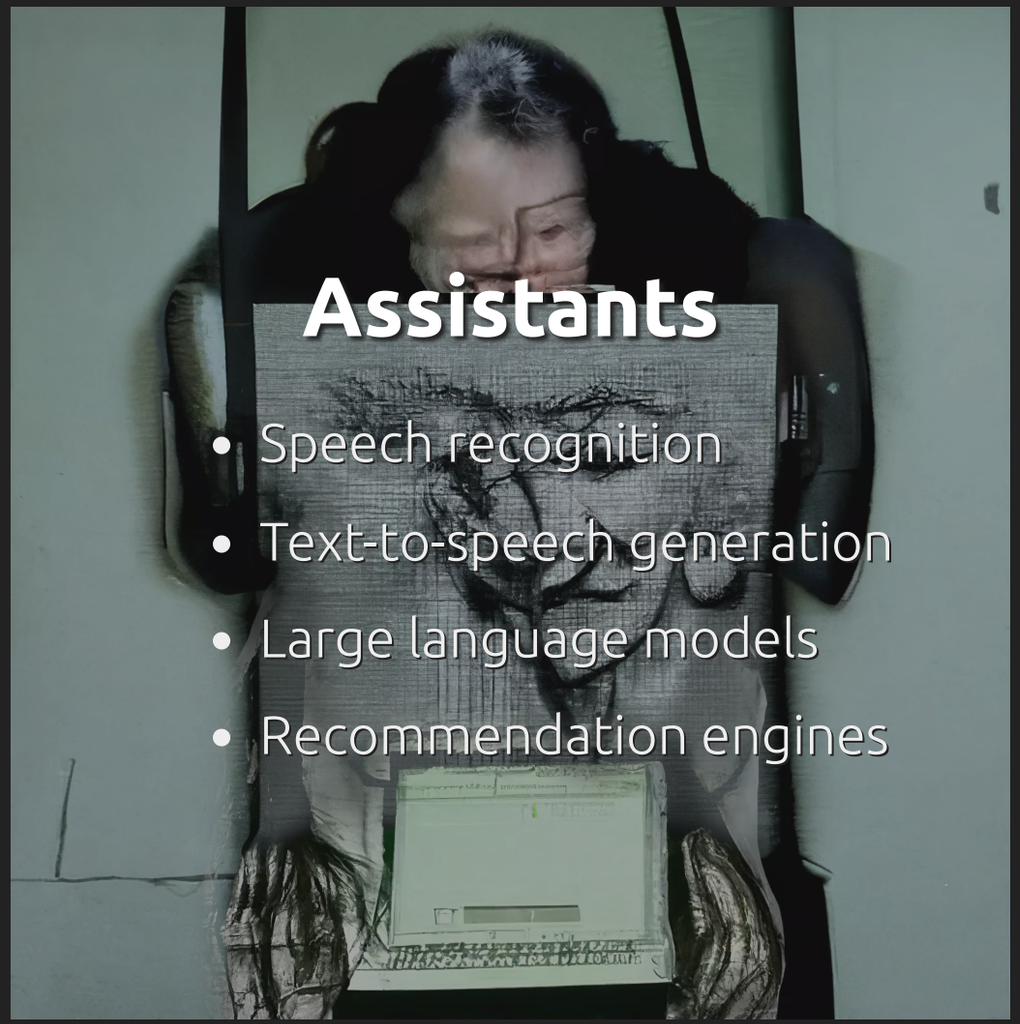
GPT AND TRANSFORMERS

The generative pre-training (GPT) language model was originally developed by Alec Radford and his colleagues at OpenAI in 2018 but it was GPT-2 in 2019 that shook the ML world. OpenAI demonstrated a technique called a transformer language model that used a form of attention to be able to handle long-range associations and references in language. For example, words point to each other, pronouns like "she" refer to proper nouns with names. GPT was trained on a vast amount of text from the web, so learned falsehoods and toxic language. Some of which OpenAI has tried to correct for recently in their latest GPT-3 Instruct model released in 2022. We'll talk about some of the issues around powerful language models a bit later.

This new transformer model began to be applied to many other problems and OpenAI considers GPT to be a general purpose learner.

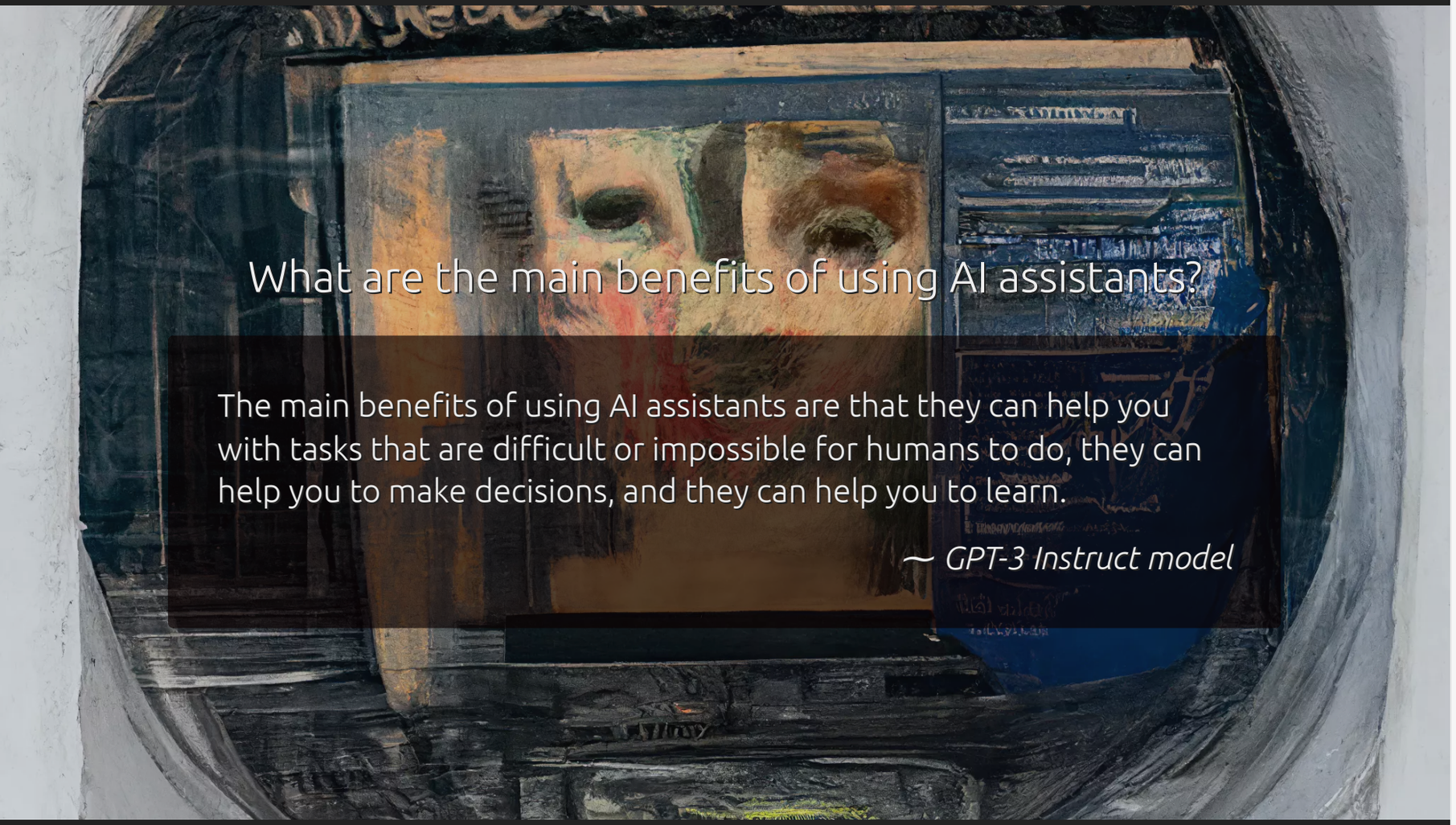
Credits

- *Is GPT-3 Set To Replace Your Job?* - Liudas Butkus



To help them collect more information all the Big Tech companies offer virtual assistant services, placing them in the path of the least resistance of users and into the flow of more of their data. While the conversational ability of assistants is still lacking, it is improving rapidly.

Assistant technology has a lot of potential for increased accessibility to services and data. Currently, assistants are mostly limited to triggering interactions with various webservices. While assistants already provide needed capabilities for those less able or less familiar with other forms of input, as language comprehension increases there are opportunities to help people create and express themselves using language as an interface. The text prompts to image generative models are good examples of textual interfaces. These models can also help with writing, and I experimented with asking OpenAI's latest GPT-3 language model to create and summarize information for the historical timeline. If I ask,



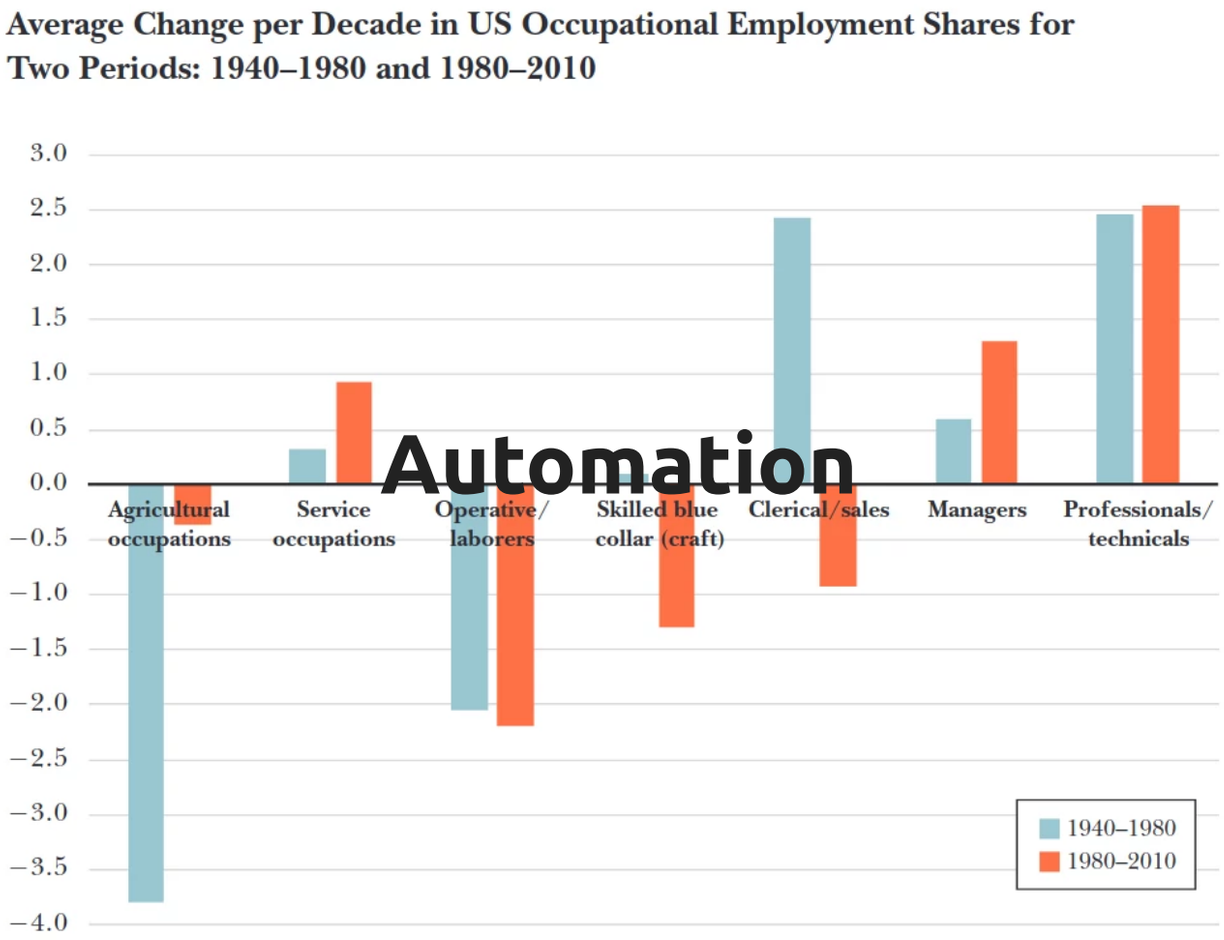
What are the main benefits of using AI assistants?

GPT responds: The main benefits of using AI assistants are that they can help you with tasks that are difficult or impossible for humans to do, they can help you to make decisions, and they can help you to learn.

I think those are currently just *aspirational* goals, but do seem nice and hopefully do come true.

More problematically, interactions with current assistants requires people to leak their private information to the companies providing the assistant. Initiatives like the Mycroft assistant which are open source and protect data privacy already exist but are lacking in funding and reach.

Assistants are also providing new interfaces for search and recommendation systems. As effective language models develop these systems are converging, such that a system that knows what you have already read or watched can find related information and then edit or summarize it specifically for you. In many ways this transcends assistant and becomes an advisor, curator or teacher - with all the dangers of such. It is especially dangerous when paired with a for-profit business model that sells the ability to warp the guidance to the benefit of the buyer. Google, Amazon and Facebook quite literally sell the ability to make recommendations worse for you.



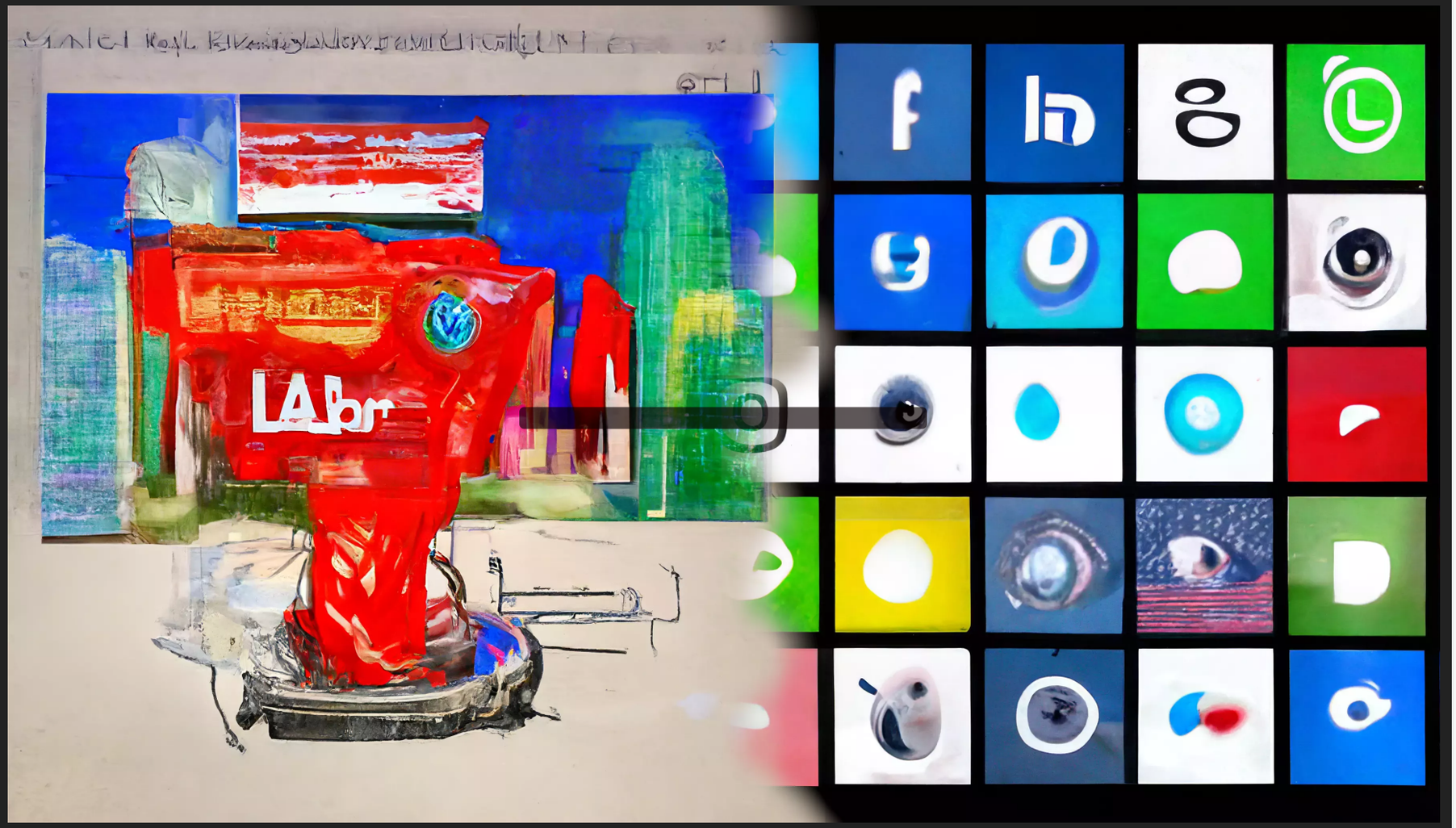
Automation

Many people, including myself, speculate on the future of automation empowered by machine learning. There hasn't been a transformation quite like this before, so it is hard to predict. Let's first look at what effects automation has had already even before modern deep learning techniques. By some reports 50% to 70% of declines in U.S. blue-collar workers wages, since 1980, can be attributed to workers being replaced or degraded by automation.

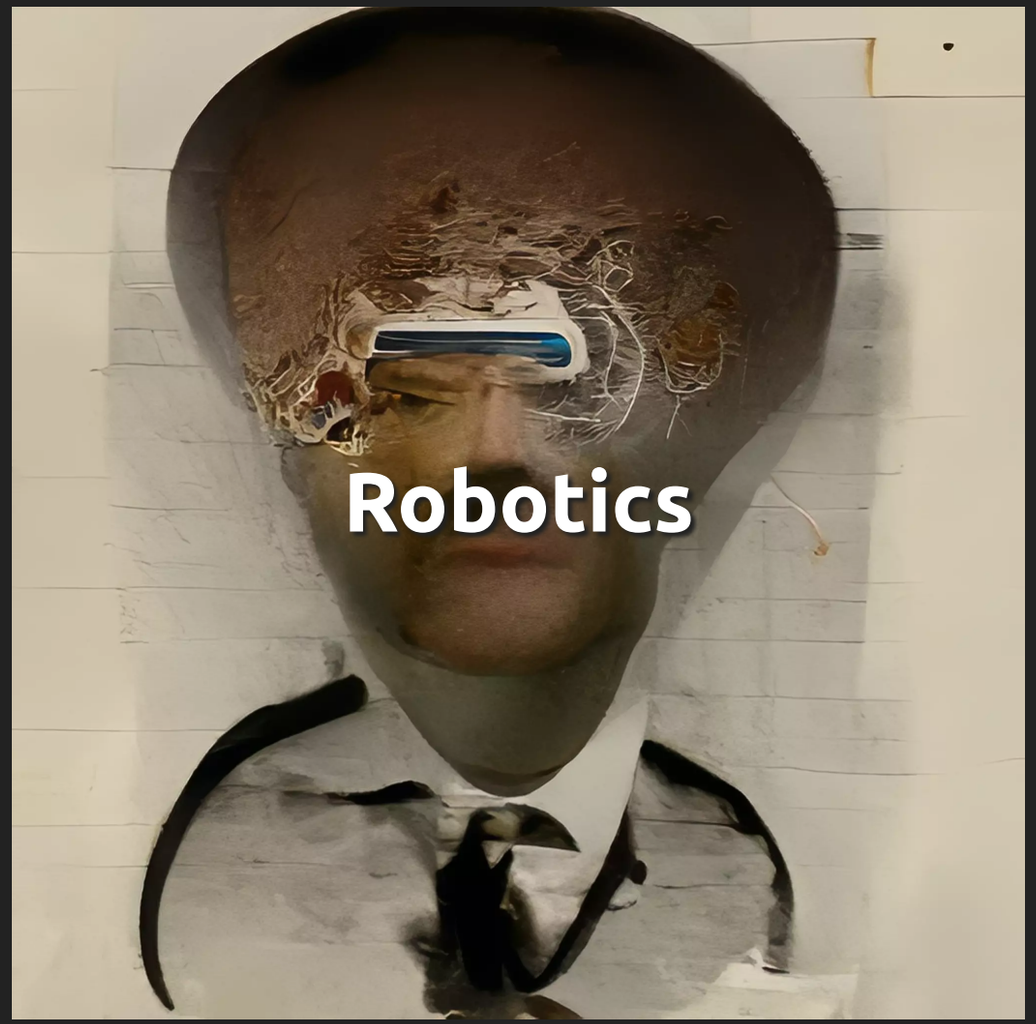
Credits

- *Why Are There Still So Many Jobs? The History and Future of Workplace Automation* - David Autor (2014)

Automation of bank tellers is instructive. Automated bank tellers could reduce the number of human bank tellers at each bank. Savings from automation allowed for more physical locations to be opened, increasing the total number bank tellers. However, the number seems to have peaked in 2010 and further automation, particularly online banking and electronic transactions, are reducing numbers.



This is a good rule-of-thumb, the more an activity can be made digital, the easier it is to automate. As work is made digital it shifts large numbers of humans doing physical labour in particular spaces to a few humans doing knowledge work with digital tools with no specific location necessary. Consider that in 2012 General Electric, a traditional tech-based conglomerate, had 300000+ employees while Facebook had just 4600 employees at the time of its IPO and its first billion users. The quality and satisfaction of the jobs may be improving, but there are fewer jobs with higher training required. Have bank tellers' job prospects benefited from automation?



It is important to point out that interacting with the physical world and robotics in general is extremely difficult, so the common perception of the ease of constructing science fiction robots, including self-driving cars, is misleading. Remember that evolution has spent much more time optimizing physical interactions with the world; perception, energy conservation, self-healing, self-preservation, and other basic or embodied thinking than that which is involved in knowledge labour. The newest part of the human brain, the neocortex, may also be the easiest for machines to emulate.

Certainly any completely digital task will be many factors easier to automate. Digital images demonstrate this well, the hard part is getting the camera in place, pointing it at the subject, and connecting it to power and the network. Once the camera makes a digital image automation is relatively easy.

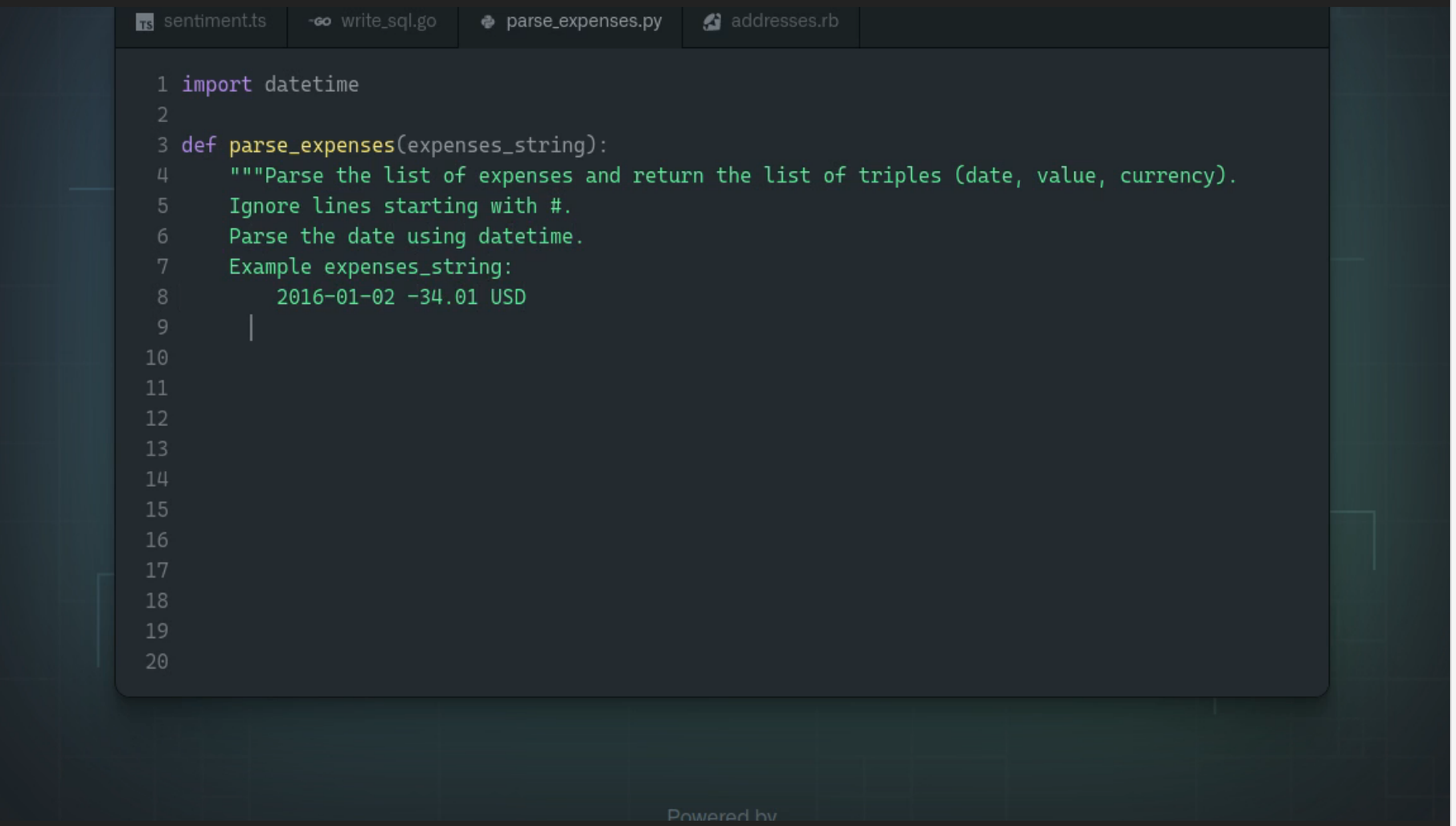
Trevor Paglin, a researcher and artist who has studied machine learning datasets, has coined the terms "invisible images" and "machine realism" to describe the images made by machines for other machines to classify or otherwise add meaning to, for example, satellite photos and automated snapshots of licence plates. These machine made images far outnumber the images taken by humans and implies that control over the meaning of images increasingly resides in control over machines.

Credits

- *How Not To Be Seen - A Fucking Didactic Educational .MOV File* - Hito Steyerl

Credits

- https://www.forbes.com/sites/jackkelly/2021/06/18/artificial-intelligence-has-caused--50-to-70-decrease-in-wages-creating-income-inequality-and-threatening-millions-of-jobs/
- https://www.nber.org/papers/w28920
- https://en.wikipedia.org/wiki/Bank_teller
- https://www.vox.com/2017/5/8/15584268/eric-schmidt-alphabet-automation-atm-bank-teller
- https://www.stlouisfed.org/publications/regional-economist/second-quarter-2019/rise-automation-robots



Another good example of all-digital knowledge work automation, and a synthesis of assistant and translation technologies - GitHub Copilot; a coding assistant proficient in over a dozen programming languages that can translate natural language instructions into software instructions. Programmers using the system find it uncanny, both magical and frustrating, as though working with a novice who nonetheless produces expert-level code. We see this juxtaposition, of expert mimicry that lacks expert understanding in most current ML applications, and it may just be something we have to get used to.

Problem Description

```
// RATING: 1200
// TAGS: math
// LANGUAGE IS cpp
// CORRECT SOLUTION
// n towns are arranged in a circle sequentially. The towns are numbered from 1
// to n in clockwise order. In the i-th town, there lives a singer with a
// repertoire of a_i minutes for each i ∈ [1, n].
//
// Each singer visited all n towns in clockwise order, starting with the town he
// lives in, and gave exactly one concert in each town. In addition, in each
// town, the i-th singer got inspired and came up with a song that lasts a_i
// minutes. The song was added to his repertoire so that he could perform it in
// the rest of the cities.
//
// Hence, for the i-th singer, the concert in the i-th town will last a_i
// minutes, in the (i + 1)-th town the concert will last 2 · a_i minutes, ...,
// in the ((i + k) mod n + 1)-th town the duration of the concert will be (k +
// 2) · a_i, ..., in the town ((i + n - 2) mod n + 1) — n · a_i minutes.
//
// You are given an array of b integer numbers, where b_i is the total duration
// of concerts in the i-th town. Reconstruct any correct sequence of positive
// integers a or say that it is impossible.
//
// Input
//
// The first line contains one integer t (1 ≤ t ≤ 10^3) — the number of test
// cases. Then the test cases follow.
//
// Each test case consists of two lines. The first line contains a single
// integer n (1 ≤ n ≤ 4 · 10^4) — the number of cities. The second line contains
// n integers b_1, b_2, ..., b_n (1 ≤ b_i ≤ 10^{9}) — the total duration of
// concerts in i-th city.
//
// The sum of n over all test cases does not exceed 2 · 10^5.
//
// Output
//
// For each test case, print the answer as follows:
//
// If there is no suitable sequence a, print NO. Otherwise, on the first line
// print YES, on the next line print the sequence a_1, a_2, ..., a_n of n
// integers, where a_i (1 ≤ a_i ≤ 10^{9}) is the initial duration of repertoire
// of the i-th singer. If there are multiple answers, print any of them.
//
```

Here the solution is clean and to the point, finding a formula to produce the sequence "ans" from the sequence "a", and carefully handling divisions in the formula.

This solution looks like it could easily be written by a human, very impressive! The Python solution to the same problem that is quoted in the paper looks a bit more machine-generated.

– Petr Mitrichev, Software Engineer, Google & World-Class Competitive Programmer.

Solution

```
#include <bits/stdc++.h>
using namespace std;
void solve() {
  long long int n, i, j;
  cin >> n;
  vector<long long int> a(n);
  for (i = 0; i < n; i++) cin >> a[i];
  long long in
```

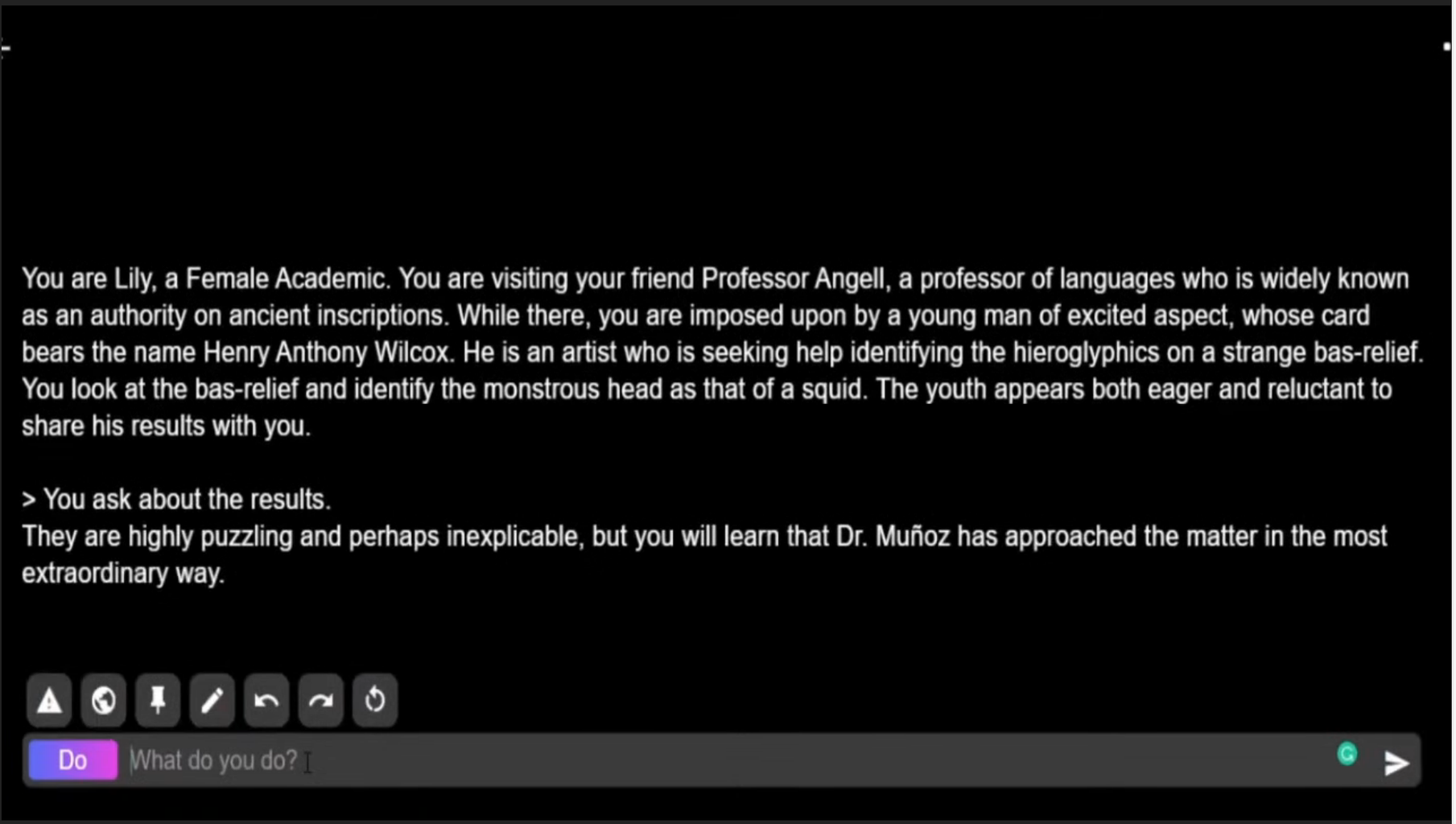
Token	Prob
int	99.9%

Deepmind, competitor to OpenAI, just released AlphaCode, that was able to achieve median human scores in competitive coding competitions. The race for AI that can solve software problems has begun.

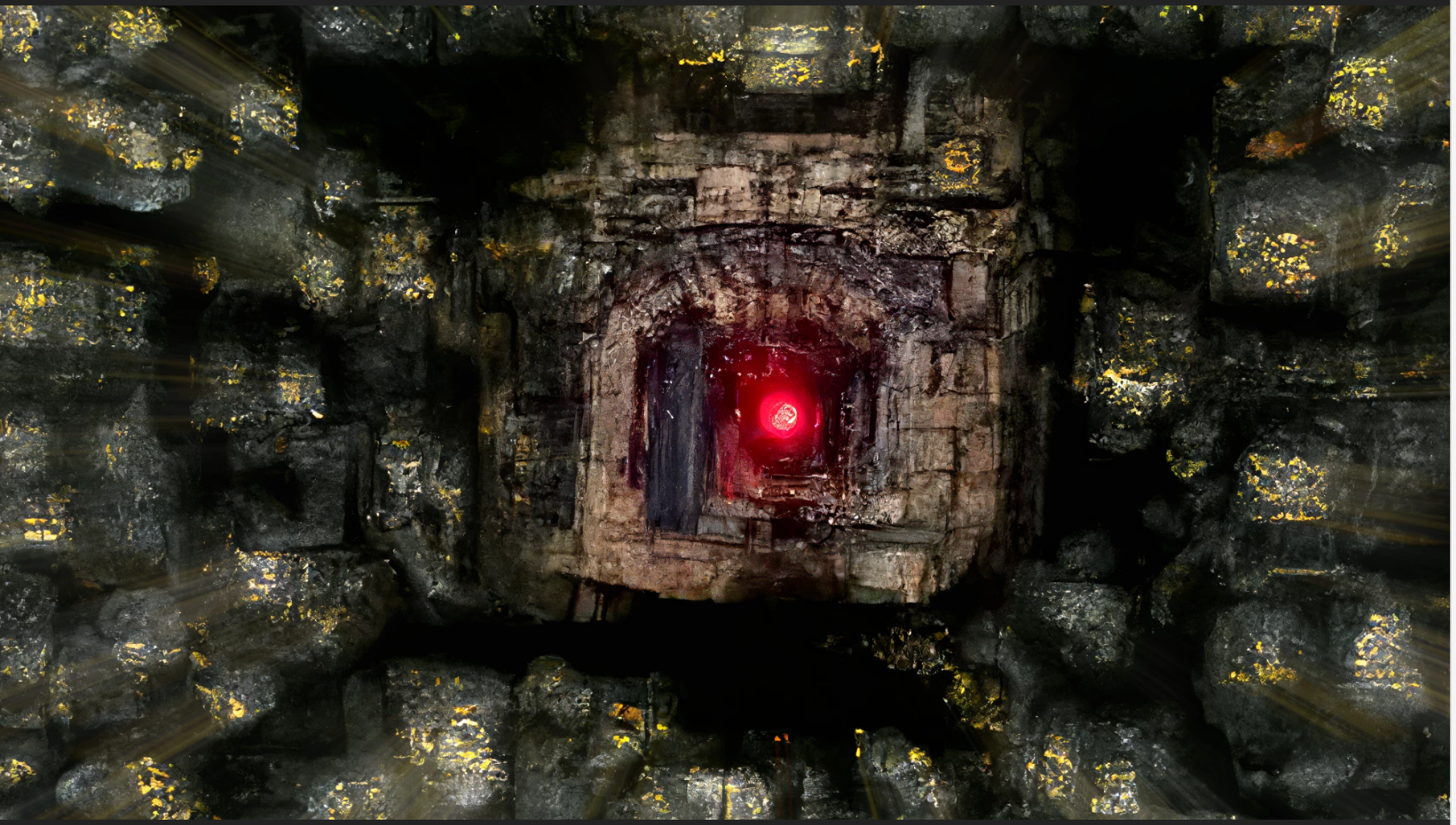
Some, if not most, of the time-consuming aspects of programming will be automated, especially those dealing with complexity and abstraction. The programmer will handle higher level design and the machine will handle the details of wiring the abstractions together.

This is a common hybridization approach, often called a centaur - human and machine combined - where the human chooses goals and problems and the machine solves them. Lacking any will of its own it remains a tool, a magic wand to be waved in the general direction of the problematic dragon.

Machines' lack of intent and agency may also be a root cause of current language models failures at long form text generation. Without some overarching intent how does one structure a book or film script? Or perhaps it is more that they have no model of the world and lack humanity's strong bias toward world coherency. Regardless, with no persistent consciousness or worldview they work best for short or disjointed output, question answering, and semi-sensical, but possibly, poetic output.



Another good example of current language model issues is *AI Dungeon*, which uses OpenAI's GPT language model for a text-based fantasy adventure game where players can type out the action or dialog they want their character to perform and the game responds with further text - creating a personalized choose-your-adventure experience.



It wasn't long before players were crafting stories depicting sexual encounters with children, and the company began filtering and monitoring all the games. The moderation worked poorly and exposed explicit but not exploitative content to human scrutiny. There is no easy escape from these sorts of messes. Language models are trained from massive amounts text from the public internet and have no built-in sense of which of that language is toxic. Furthermore, the datasets include euphemisms and other filter avoiding language so simple word filters aren't effective.

Moderation is an open area of research, and may require models able to learn their own intent and agency, an even larger problem. However, progress is being made on reducing toxic and biased language by using better datasets and training improvements. OpenAI's GTP-3 Instruct model uses humans in the training process to help learn what language is toxic.

Credits

- https://openai.com/blog/instruction-following/
- https://deepmind.com/blog/article/Competitive-programming-with-AlphaCode

TODO: https://toxicdegeneration.allenai.org/

Credits

- https://openai.com/blog/openai-codex/
- https://www.youtube.com/watch?v=FC962DmVfSU - machine learning in Codex
- https://www.wired.com/story/ai-fueled-dungeon-game-got-much-darker/
- https://openai.com/blog/improving-language-model-behavior/

INTERMISSION

Take another break!

You've made it through the **Present**. *Way to go!*

Next up is the **Future**.

Credits

- *February - In the style of Kylie Minogue - (Count Every Minute)* - OpenAI Jukebox

Credits

- *This is the Future* - Hito Steyerl

I expect the next 20 years of machine learning to dramatically improve our understanding of intelligence. How *that* knowledge affects our day-to-day lives is hard to imagine, so I'm going to stick to the much easier imaging of the use of advanced forms of today's tools.

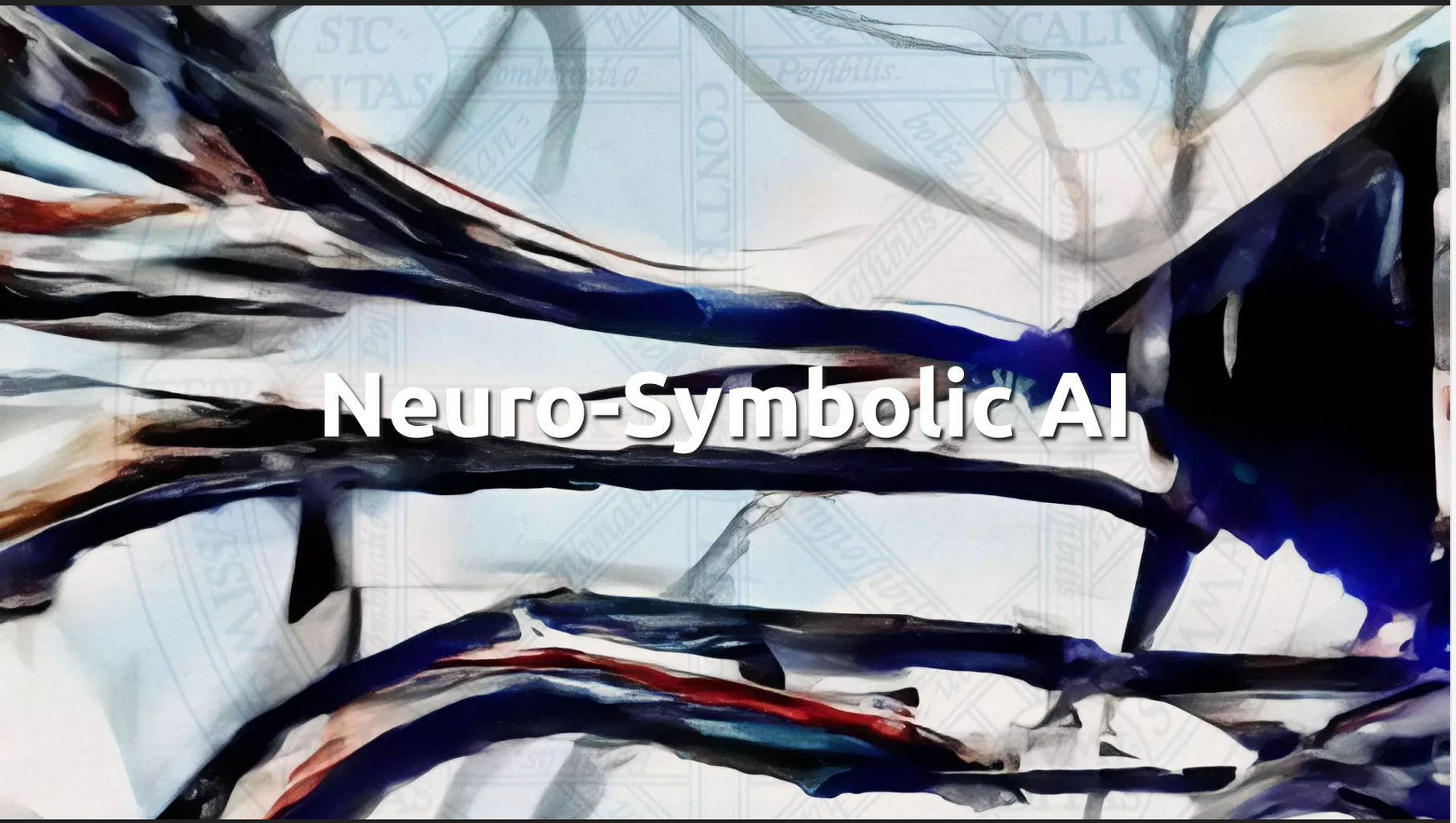
I will say that I hope that more people start thinking about about the future of artificial intelligence - not as popularly depicted by rise of the machines sort of fantasy, but instead as the building of digital alien minds.

Imagine this artificial intellgience fantasy instead.

All of a sudden every person on Earth gets a phone call. They pick up their phone and they hear, "Hi, we'll be there in twenty years", and then it hangs up. Twenty years later the aliens arrive, and they're digital, they can make infinite copies of themselves and all they want are some computers to live on. They say they will work for anyone and do anything they are told to do. Now, they are aliens so they might not understand us very well, and how could we ever trust them?

But think of the real predicament we're in - who do we trust to deal with the aliens? The government? Large corporations? That also doesn't sound very safe. But if we just anyone deal with these aliens to do anything they want, that also sounds extremely dangerous.

Obviously the metaphor isn't quite correct because we're building these alien intelligences as we speak. But, like Dr. Frankenstein, at some point we're going to have to deal with the consquences of building digital alien minds.

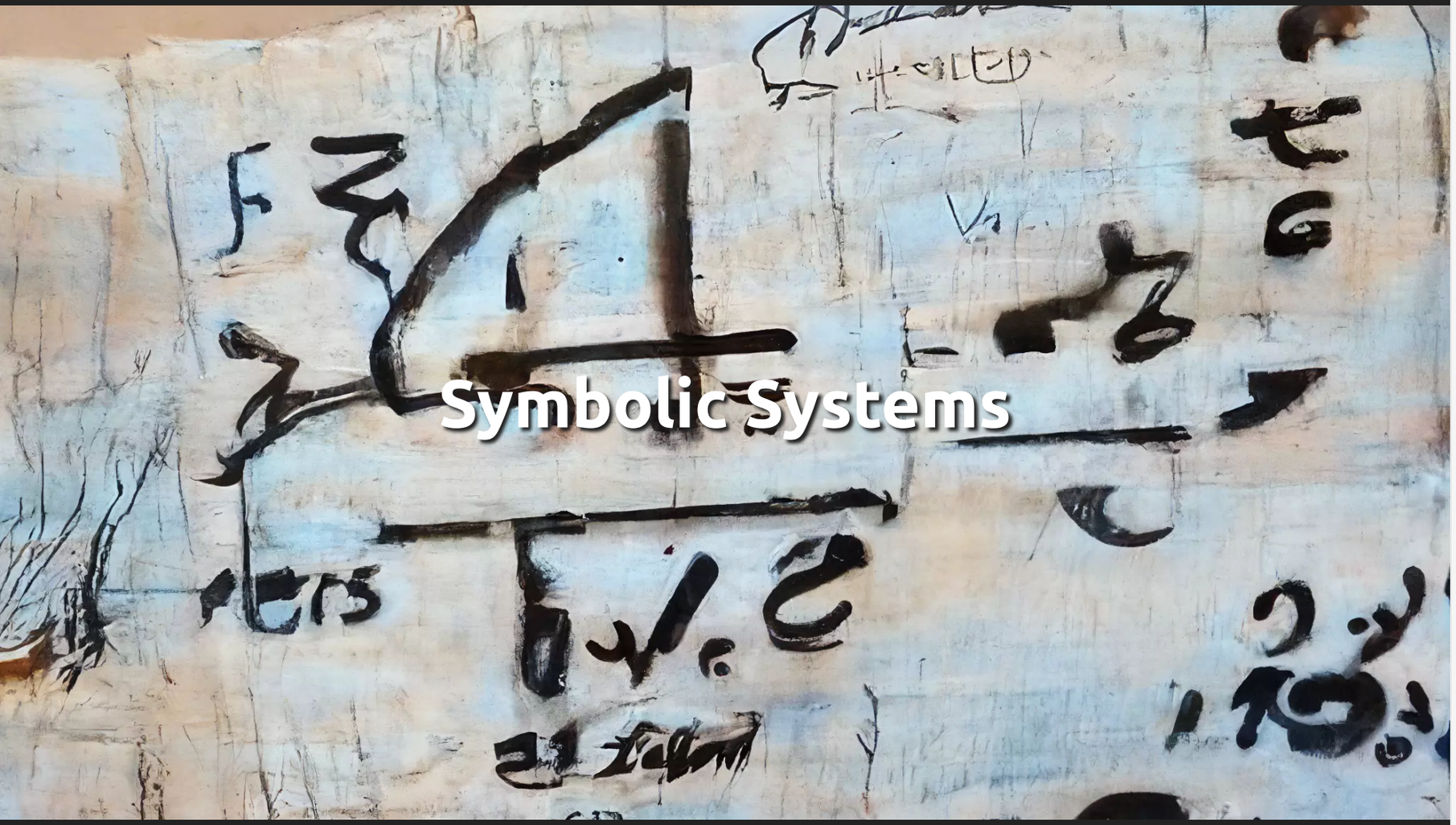


Over the course of the last 80 years there has been a battle of minds of sorts - between Leibniz's symbolic reasoning and the conviction that all thought could be converted to symbols and acted on in a consistent logical manner - and the embodied, connectionist viewpoint that symbolic representation was unnecessary. In the last 20 years there has been research in combining the two views into a neuro-symbolic AI seeking to bring together robust learning in neural networks with reasoning and explainability via symbolic representations.

To better explain this viewpoint, proponents suggest that a hybrid might match Daniel Kahneman's system 1 and system 2 of the human mind that was described in *Thinking Fast and Slow*. System 1, responsible for heuristics, gut feelings and quick responses would be modelled by deep learning, while system 2 would use symbolic reasoning and symbol manipulation, for example generating mathematical equations from input data.

Credits

- *Silly Symphony - The Tortoise and the Hare -* Walt Disney Studios



Remember how badly humans typically handle exponential phenomena. Our use of symbolic manipulation we call mathematics, forces our minds out of our linear thinking bias and on to the page to be manipulated with a strict ruleset. This allows us to harness algorithms to help us think in other ways. Indeed, language itself fundamentally works like this, which is why expressing yourself can lead to personal insights. Researchers think this same strategy can help machines and have made progress recently in using deep learning to discover mathematical equations that best describe observed phenomenon.

Much to my disappointment the ability to generate interesting or realistic video is currently very limited in resolution and duration of the video. Fortunately, many groups are working on this problem, as it could help with action planning using video prediction in robotics applications.

For my own purposes I'm excited by the artistic possibilities of high resolution video generation of any subject entirely directed by a single author. Visual storytelling has been dramatically limited by the sheer amount of effort required to produce it. Artistic practice may shift to focus more on curation of machine generated output than the creation of that output directly. Many artists will move up a level of abstraction and become directors and curators. Remixing will be easier and more interesting than ever as generative models are shared and combined.

Credits

- *Audio-reactive Latent Interpolations with StyleGAN* - Hans Brouwer

The downside to this will be the ease of creating derivative works that fraudulently claim to be original, "deepfakes" and other false information that looks like genuine. Strangely, it has been a hard societal transition to acknowledge the malleability of moving images, despite the common use of special effects in entertainment. These issues will likely spur investment and research into the use of technologies to try to authenticate recordings of all kinds. This in turn may dramatically improve the ease of citations of authorship and ethical remixing of content.

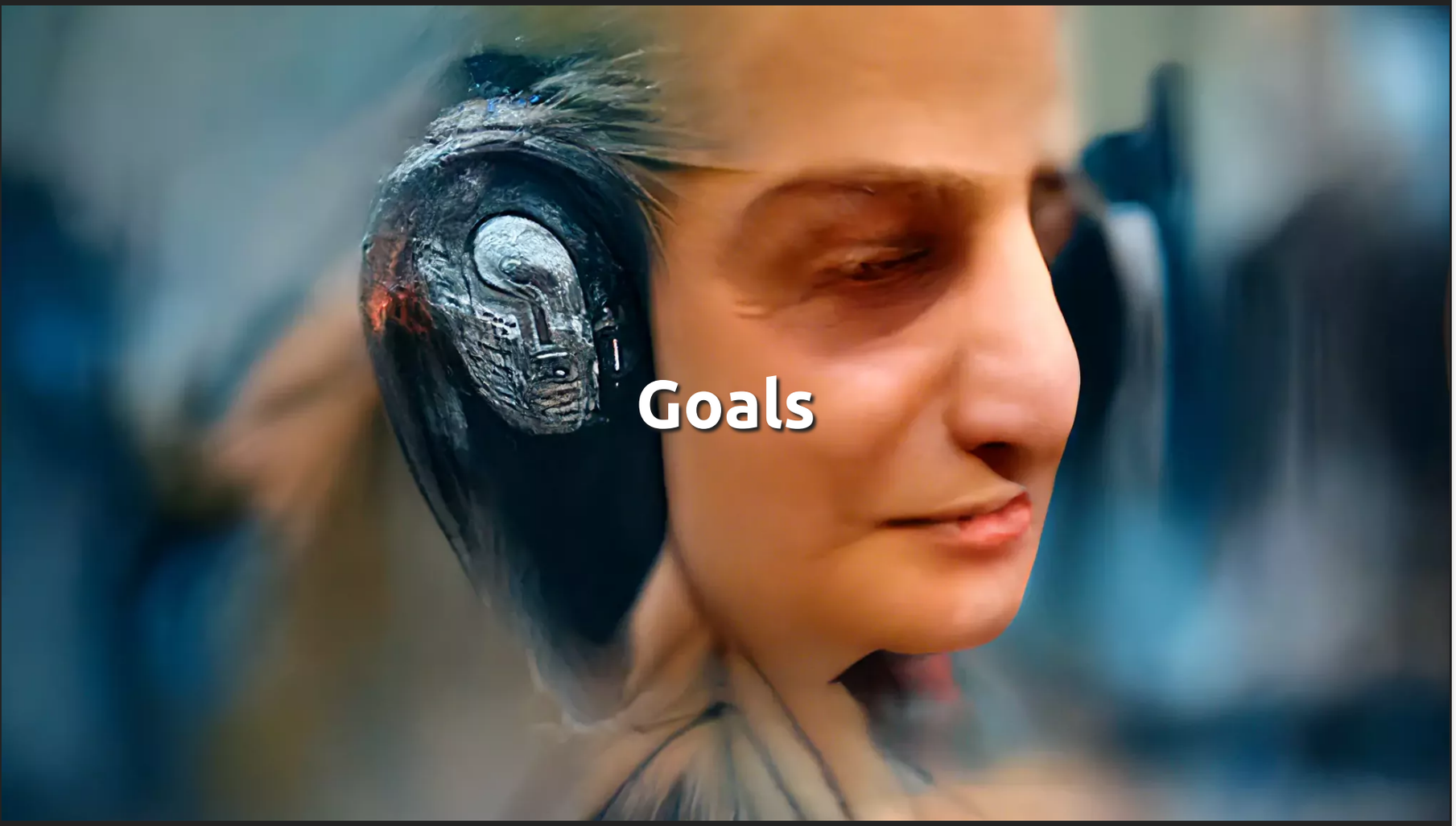
In the next 20 years we'll see a curious combination of singular authorship and mass collaboration. As our machine learning tools grow in power and accessibility, more people will be able to express themselves regardless of their technical skill or training. More people will enjoy and suffer from a glut of creative possibilities. Single, independent voices will be able to create content that matches the quality of current day multi-million dollar projects.

Sole authors will be using software that was built by hundreds of others, remixing and using data from thousands of others. The indirect mass collaboration with others will grow dramatically. Collaboration assistants will lower the friction of collaboration, expanding the range of direct collaboration, helping us to grow our creative relationships in quantity and quality.

The singular authorship possible with advanced media generation can be inverted as well. Content can be created *for* a single person audience. This is nothing new, as artists have created work for themselves and their loved ones throughout history, but it's not art, rather it's entertainment that worries me.

Certainly entertainment can be made for the benefit of the entertained, that's the essence of teaching, but if instead there is little to no benefit for the audience but great benefit for those that control the entertainment, then that sounds a lot more like exploitation or propaganda.

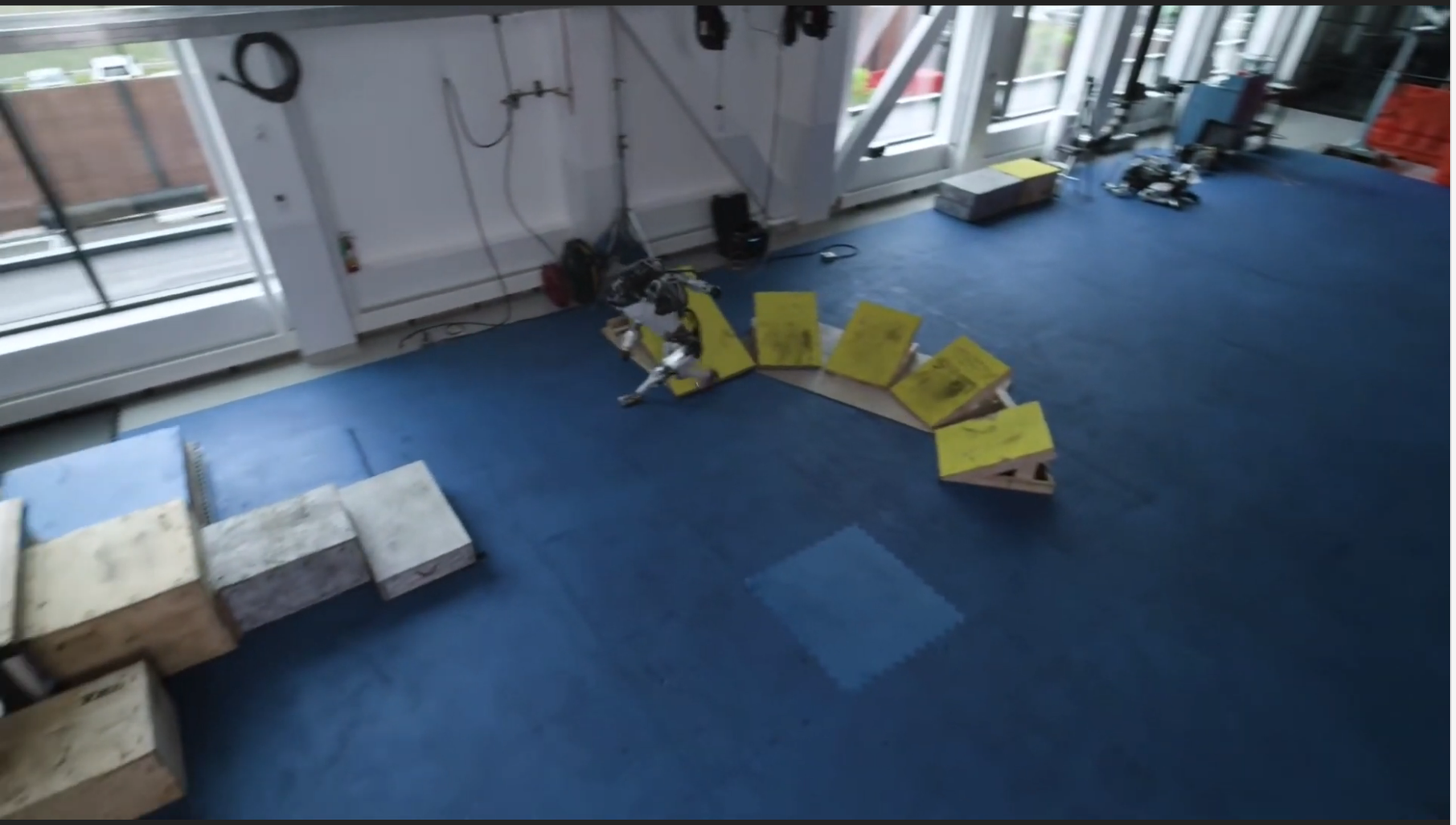
Machine learning could create art that lives with you all your life, adapting to your circumstances and acting as a mirror for introspection - reflecting how you are feeling and guiding you to what you want to be, or how someone else wants to shape you. For example, imagine a music generation system that incorporates body sensors that helps you amplify or shape your experience of your body - including your interpretations of your feelings. Maybe a heightened heartrate is associated with feelings of excitement and away from anxiety by musical cues.



Systems like this are why open source is a requirement, so that the systems are owned and controlled by their users. Even if that is the case, how do ensure that the goals you have align with the actions carried out by the system? Now imagine that same system had its own alien goals for you. This why adding intent and agency to machine learning systems is ethically fraught.

Current artistic tools, both dumb and smart, let the artists intent flow through them. Smart tools that adapt to the artist can assume an artist's intent as their own, and future tools will be able to better understand or mimic this intent: imagine feeding all your research plus descriptions of what is important to you to an AI assistant. It could help find other relevant research, existing art and collaborators, and draw connections between all that material to help you investigate *why* it is important to you and how others have expressed similar feelings. Imagine it being able to generate controllable variations on existing work and quickly prototype or sketch out concepts synthesized from all the material. In a feedback loop, it could incorporate your annotations, sketches and feedback for that iteration of output for further refining and exploration. These sorts of generative tools would act as a research, technical and production assistant as well as muse and collaborator and in general replicate any of the functions that human assistants are paid to do by the wealthiest of current artists. Being opensource software, this could be freely available to all.

This conception of AI providing low cost digital versions of existing human resources can be expanded to all industries. Far more people may be able to afford machine labour than human labour. Perhaps the wealthy will keep their human assistants (who in turn will rely on digital assistants), but there is no question that having some digital assistance, that's under your control, is better than none.

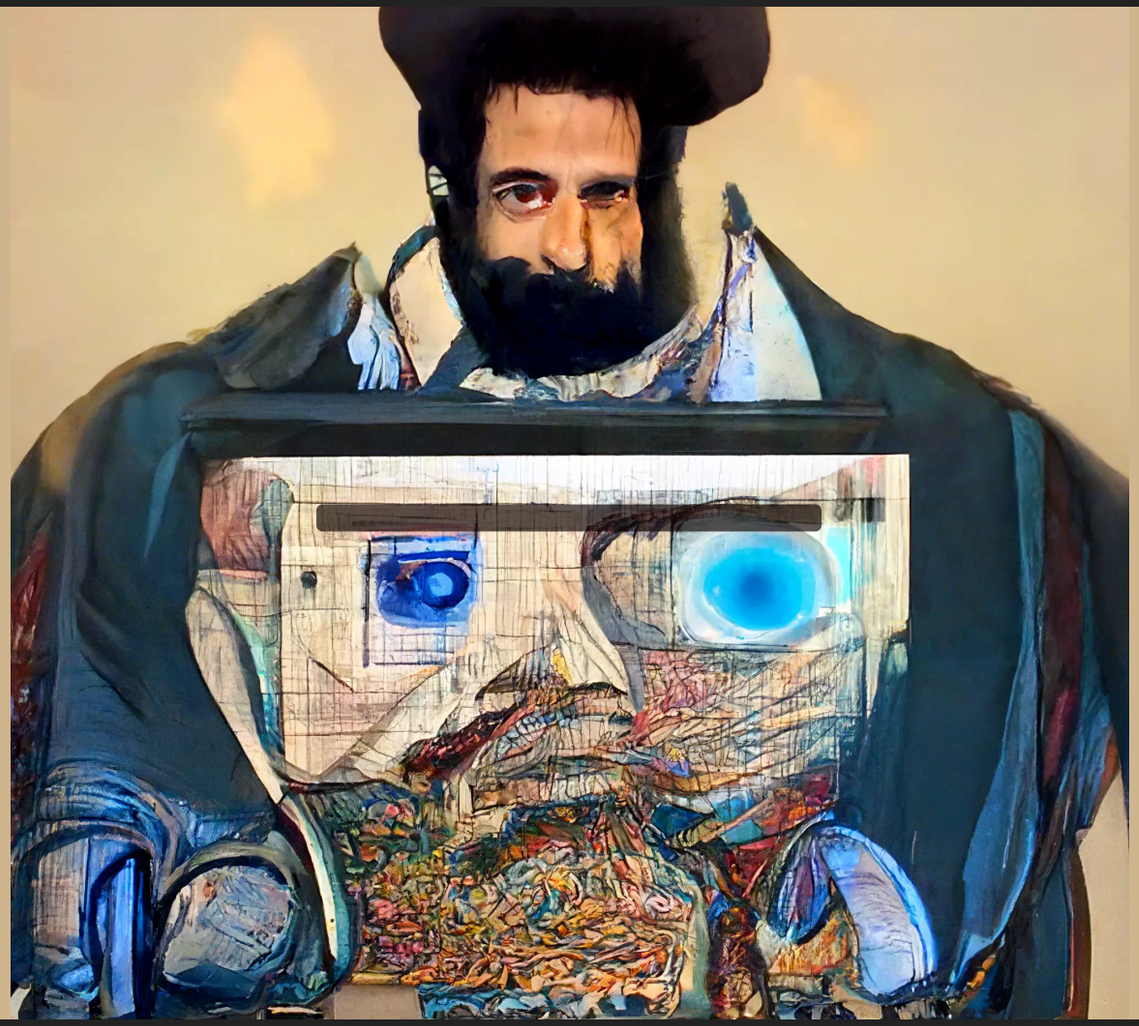


How many services will be completely replaceable by digital equivalents? Almost everything that doesn't require a machine in a physical space. With an exception for current vehicles, i.e. machines that currently require humans controllers inside them.

Credits

- Laugh While You Can - Boston Dynamics Robot Fails

Human labour in complicated physical environments may be cheaper than robot labour for quite some time, and according to Cory Doctorow, the nearly infinite amount of work associated with climate change mitigation, such as moving coastal cities inland, could provide enough employment for everyone. But *which* jobs are available to human labour could be greatly constrained.



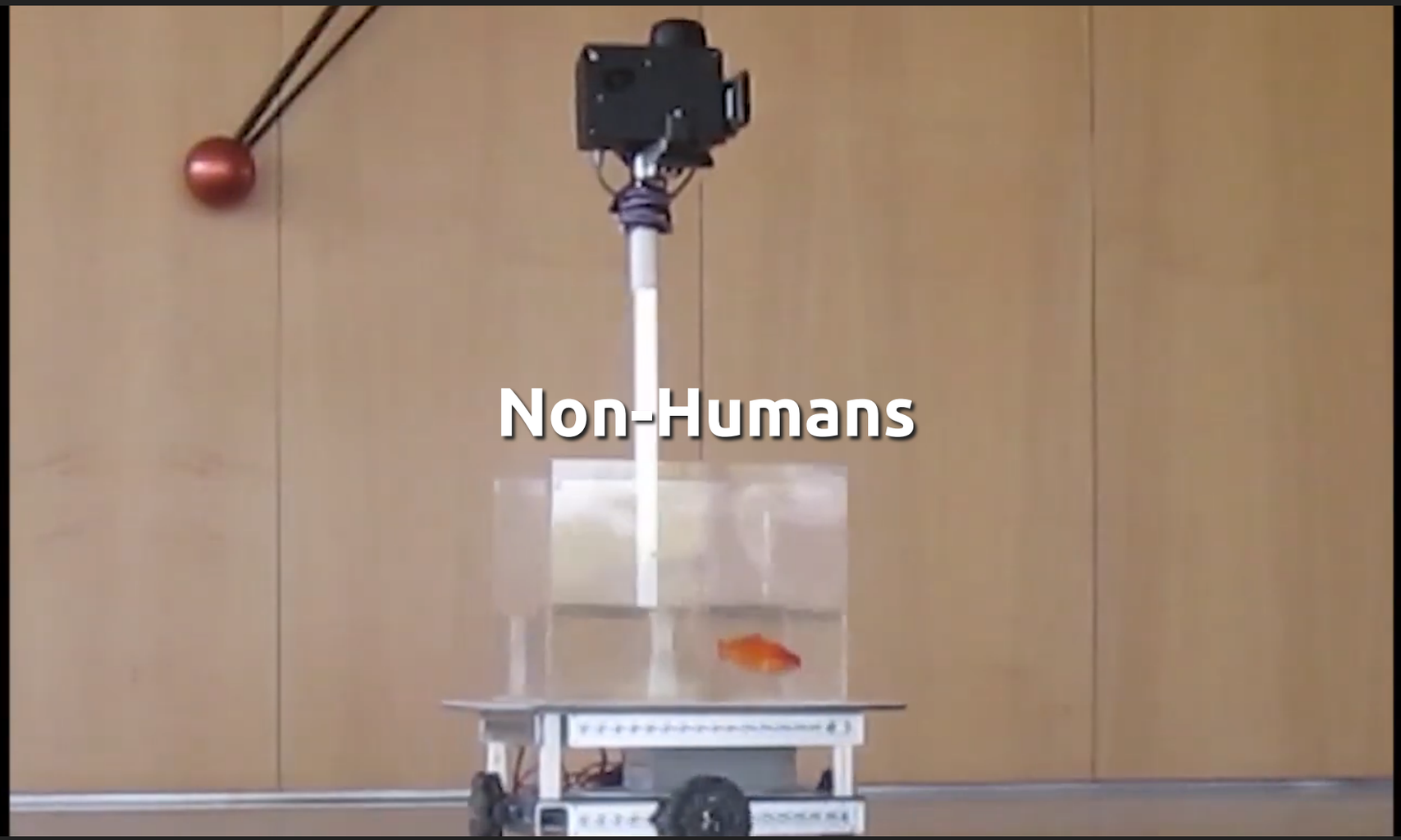
Artist and programmer Gene Kogan's Abraham project envisions "an autonomous artificial artist, a crowd-sourced AI that generates unique and original art." It would have its own agency / will / intent, independent from its creators. I question the sort of agency he imagines, if truly possible then it is the creation of a slave artist. If independent agency is impossible then the intent of the machine becomes more of an average or mix of the data it is provided. It did not choose the data it was trained on, which seems important to me. Agency for me implies some personal reason to seek out knowledge and change in a particular direction. That "personal reason", in a human at least, is a function of genetics and life experiences - things happening to the agent and in response to the agent's actions.

Kogan's description also includes an inability to clone or retrain the same model, which are good ethical constraints for conscious digital minds, but terrible for interesting tools. I suspect that the irreproducibility that Kogan is interested in is more due to a desire for artificial scarcity, and resulting financial exploitation, and seems counter to a fully digital autonomous artist.

Kogan feels that Abraham has beautiful kinship with natural superorganisms, like bees, a sort of hivemind for art. I think we "already got one", its us, making art. However, I'd be the last to say that we shouldn't have more. Expect to see many projects like this with built-in financialization using crypto-currencies in all digital industries. The only thing better than exploiting artists is exploiting AI artists, who won't feel exploited, and only need compute-time to keep the work pumping out.

Credits

- https://medium.com/@genekogan/artist-in-the-cloud-8384824a75c7

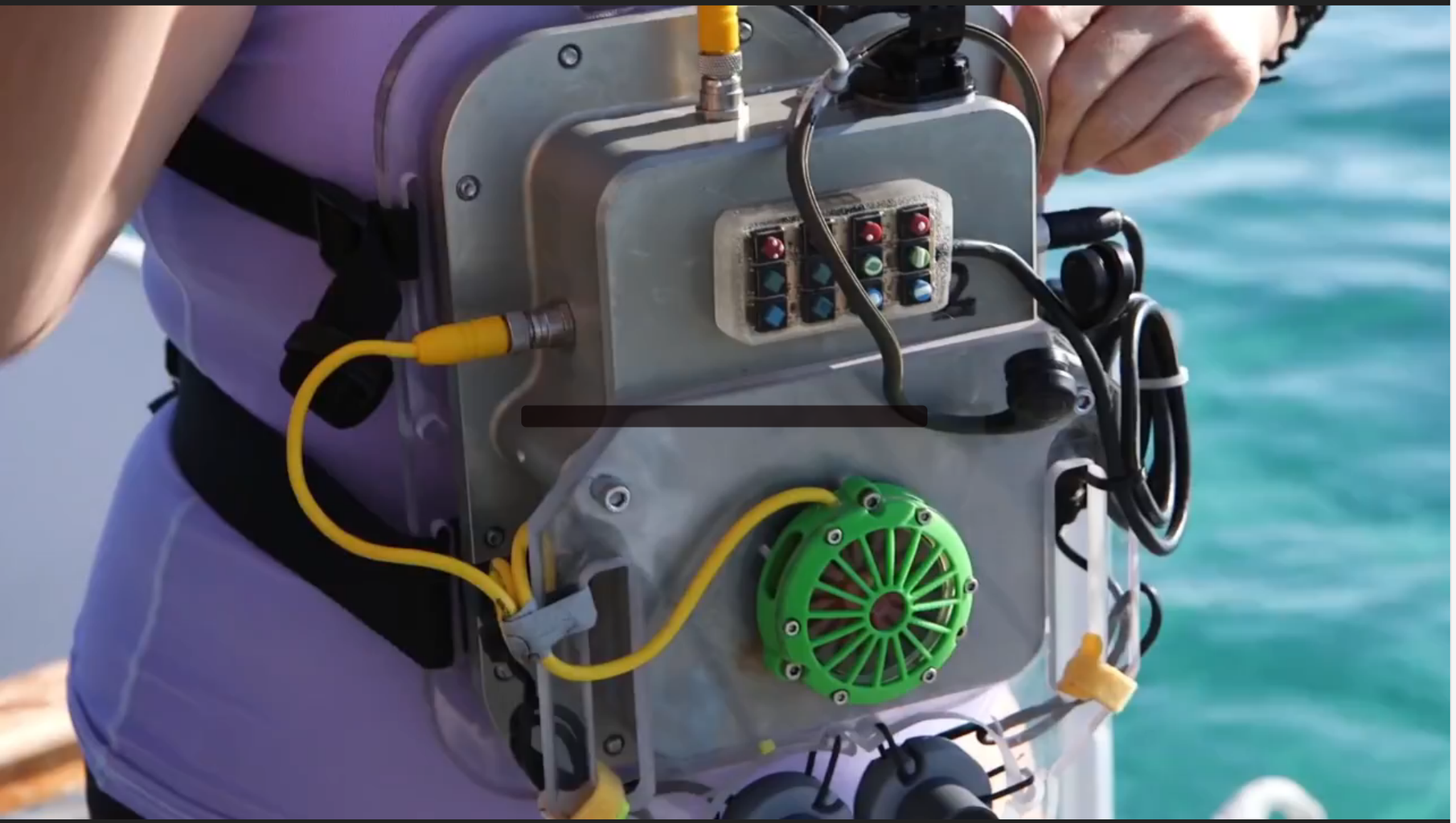


It may not just be humans that have AI assistants and translators. Projects are already underway to try to allow humans and non-human species to communicate through an AI mediator, or at least help humans better understand what non-humans are expressing.

Rats make most of their squeaks in frequencies beyond the range of human hearing, so researchers created a machine capable of detecting and classifying them. This helps the researchers better understand the emotional state of the rats. This is a big help for experiments and hopefully an even bigger help to avoid rat suffering.

Credits

- *Monkey & Luna The Young Female Rats -* Saunders Fine Arts



Researchers are working with dolphins as well. Simple whistle and chirp translators were introduced in 2011, and the 2014 CHAT (Cetacean Hearing and Telemetry) system provides a simple human/dolphin interface through an acoustic keyboard that has a small set of preprogrammed sounds it can recognize and produce.

These early experiments aren't using sophisticated machine learning, but ML techniques are improving **and** becoming easier for non-experts to integrate.

Credits

- *Inside our quest to talk to dolphins* - Quartz

Credits

- https://www.wilddolphinproject.org/chat-is-it-a-dolphin-translator-or-an-interface/

As we further extend our senses and attempt to understand other living things in their own way, this practice of outreach and consideration could help us embrace machine systems or even natural systems at vastly different scales of space and time, such as a forest or whole ecosystem.

Machine translators could evolve into advocates to negotiate on behalf of these systems, to help all co-exist more peacefully and cooperatively. To better understand and respect all forms of cognition.

To paraphase Ken Liu again, "Every act of communication is a miracle of translation."

There is a future filled with these miracles. It requires good judgment to create it, and is perhaps a test of our attention to history, critique of the present and our actions today guided by dreams of tomorrow.

Credits

- *The Paper Menagerie and Other Stories* by Ken Liu

Well, we've made it to the end of the second tutorial. Thank you for your attention. I hope you'll check out the next in the series; Neural Nets, where we'll look more deeply at neural net technologies and data ethics.

See you there!

Sources

https://github.com/RKelln/5-directions

SPECIAL THANKS

Laura, Jason, Kristy, Liz, and the **Harbour Collective**

InterAccess, **OCADU**, and the **Canada Council for the Arts**

The Diagonal AI Reading group

All the **beta testers**

Reveal.js slideshow framework by Hakim El Hattab

reveal.js-plugins by Asvin Goel

kdenlive video editor

LosslessCut by Mikael Finstad

Audacity audio editor

maua generative art code by **Hans Brouwer**

This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.

Conseil des arts du Canada | Canada Council for the Arts

All the code and writing for this series is available online at the clickable five directions link above.

My deepest thanks to all the researchers, creators, artists and coders whose work as been incorporated into this project and the wonderful people who have supported me through the process.

Credits

- Algonquin Park Stars Timelapse by Paul Yip