reddit COMICS comments

Welcome to Reddit,
the front page of the internet.

BECOME A REDDITOR and join one of thousands of communities.

Among computer scientists, is the Sloot Digital Coding System actually considered a lost invention?

(self.compsci)

submitted 1 year ago by compsci

I came across the Sloot Digital Coding System on a list of "lost inventions." What I read indicated that in the late-90s, it was able to play multiple movies at once using much less memory than we could compress such files even today, 20 years later. The hoax aspect of it is not something saying it isn't about compression; it was a whole different way of thinking. But it sounds like this wasn't widely demonstrated, I only find a reference to a demonstration to a Philips exec, the company they tried to sell about to" sign a deal with when the inventor died. Afterward, a floppy disk was missing, so the whole thing is lost. Which all, frankly, seems like BS to me. An invention back then so much better than anything today, like saying you could build a 747 that only weighed 1/5 of the weight. One disk of code lost, but nothing left in terms of the underlying theory or other info to sketch out how it was done? No partner knowing how it worked? Not enough time for others to "solve" this last mystery, given how great the performance was? And no one really sees it working? It all sounds like it was going to be a scam... makes me think Theranos and the like. But, there may be something to this invention. So, wanted to ask a community of people who think of bits that other don't, if this considered a real invention that was lost among the computer community? Is there evidence of it existing/working beyond what I mentioned?

EDIT: Thank you all for the replies. I like to ask the experts because not only do you know they "yes or no" of my question, but help fill in more details, so thank you.

34 comments share save hide report

all 40 comments

sorted by: best ▾

Want to add to the discussion?
Post a comment!
CREATE AN ACCOUNT

[–] Technology 105 points 2 years ago

The claim is to store a complete movie in 8 kB of data. I rather suspect the explanation is that the floppy disk could not be found because it never existed.

A movie storage system that is able to have our storage requirements would have to work by having a *highly* abstract model of the movie. You could think about this in terms of the pigeonhole principle. In order for the movie to have a unique representation, it has to be encoded in 65536 bits (8 kB). There are 2^{65536} ways to arrange bits (standing on the left, or the right?). Even accounting for statistical techniques that can be used for compression, unless you're compressing the universe's abstract, you'd end up with more than 8 kB of data.

permalink embed save report reply

[–] 1 point 1 year ago*

[deleted]

[–] Incomprehensible_mate 27 points 2 years ago

I worked for a certain company once and as a joke we had a camera that could produce a beautiful, high dynamic range photo in 50 megapixels. But only that one photo

permalink embed save report reply

[–] 1 point 1 year ago*

[deleted]

[–] Well that makes it easy. Give me any movie and I'll give you a 0kb file and movie sized decompression algorithm.

permalink embed save report reply

[–] 1 point 1 year ago*

[deleted]

[–] mathisdawkins 1 point 2 years ago

Only because it destroys the previous decompression program and replace it with the new one...

permalink embed save report reply

[–] 2 points 1 year ago*

[deleted]

[–] Yeah, the pigeonhole principle does a pretty good job disproving that. A 1 hour long feature film would be 86k frames, about 1 bit would be available for each frame.

My guess, the guy was a fraud. Sold a deal too good to be true and had to deliver. Stress caused the heart attack at just the right time.

permalink embed save report reply

[–] 1 point 1 year ago*

[deleted]

[–] Yeah his bluff got called

permalink embed save parent report reply

[–] 1 point 1 year ago*

[deleted]

[–] 1 point 1 year ago*

[deleted]

[–] Technology 18 points 2 years ago*

That's not just impractical, it doesn't actually work. If you're looking for a file with n bits that matches to a hash with h bits, then on average you'll need to search 2^{n-h} possible files with that property. So, for example, say the original file was only 50MB (419430400 bits) and the hash file was 8kB(65536bits) then there should be about $2^{419430400-65536} = 1.9\times10^{126264}$ = 1×10^{126264} possible files of that length with that hash. So it's not just a matter of having enough processing power to find a match. The problem is identifying which of the matches was the original file. And the problem persists as long as the hash is shorter than the original file.

permalink embed save report reply

[–] Hunter 3 points 2 years ago

The good news is that most of those will not be valid movie files. Throw out anything that deviates from the spec, and make the user tell you which of the remainder don't seem right.

I imagine that won't take too long. Just a few universes and they'll have their full length movie in 8kb and a little bit of processing

permalink embed save parent report reply

[–] RipperCrap 2 points 2 years ago

Just use PIFS instead! 0kb files!

permalink embed save parent report reply

[–] 1 point 1 year ago*

[deleted]

[–] Valeator 22 points 2 years ago

No, many many matches will probably have correct headers / frame positions / whatever other small subset of the bits you think you can look for to find the exact match, because while the chance is astronomically low that a random file would have correct metadata, there are an astronomically huge number of possible matches. Lets say, for example, there are a total of 50kb of metadata that you can identify for that 50 MB movie. 50kb = 2^{409600} possible different configurations of bits. Given that there will be $2^{419430400}$ possible matches, which is $2^{419864000}$ times 2^{409600}, meaning that every possible configuration of those metadata bits will not only show up as a match, but show up innumerably many times. roughly $2^{419864000}$ times each.

permalink embed save report reply

[–] 1 point 1 year ago*

[deleted]

continue this thread —→

[–] hunnajd10 3 points 2 years ago

And how the heck would you know what order it was in too lol

permalink embed save report reply

[–] jeffbell 5 points 2 years ago

Youtube hot-and-new-than-6400bit-different movies.

This would not be able to play all of them

My Bad. See reply. Down voting myself.

permalink embed save parent report reply

[–] Technology 2 points 2 years ago*

You're confusing number of bits (65536) with number of possible values of those bits (2 multiplied by itself 65535 times).

permalink embed save parent report reply

[–] Eponerd 1 point 2 years ago

[–] u86155

It's got almost 20k (decimal, obv) digits

permalink embed save parent report reply

[–] bookworcam 2 points 2 years ago

Theoretically there's a pseudorandom sequence of numbers where a subset of the first 2^{65536} values correspond to all pieces of video media ever created or that ever will be created by humans. Then the 8kB data just corresponds to the nth value which can be generated iteratively. Not sure how you'd ever discover the pseudorandom algorithm and starting seed value but it should still exist.

permalink embed save parent report reply

[–] dasdolf 3 points 2 years ago

Nowadays, such a compression is easily possible by replacing a movie with a link to its VOD.

permalink embed save parent report reply

[–] Technology 2 points 2 years ago*

That's not really compression. That'd be like if I gave you a folded up piece of paper with a number on it, and claimed I'd managed to fit my house into something the size of my thumb nail.

permalink embed save parent report reply

[–] kleuterzo 13 points 2 years ago

One disk of code lost, but nothing left in terms of the underlying theory or other info to sketch out how it was done?

Wikipedia described it as a "complier". Looking through the patent, they really seem like they're just describing a data deduplication system.

The simpler explanations of the system are kind of reinterpreted and expanded on in this article, into a description of a system that sounds much more realistic.

In the context of that article, the "complier" could've been part of the encoder or decoder; "compiling" prototypical frames into a kind of database, and generating smaller data files that act as keys for the database elements. Or otherwise, "compiling" an image to output from stored elements.

permalink embed save report reply

[–] Nuncanne 2 points 2 years ago

The only maybe sense way I could see this working is having a huge library of prototypical chunks i.e. chunks of portions of a video frame pre-stored in local storage. The video file which is then transported to the decoding machine would be a list of addresses of the prototypical chunks.

If the chunks had 1 byte addresses then that would be 2^8=256 chunks, 2 byte addresses 2^16=65536 chunks. This couldn't possibly be enough chunks.

If it was an 8kb file then it would only be able to address 8*1024=8192 chunks in a 1 byte address system or 8*1024/2=4096 chunks in a 2 byte address system.

Someone might have to correct my maths if I'm wrong.

In this kind of scheme the numbers don't even add up with pre-stored chunks.

permalink embed save report reply

[–] rustechnician 2 points 2 years ago

It just read the patent (in Dutch, at https://worldwide.espacenet.com/patent/search/family/019764878/publication/NL1005930C2?q=pn%3DNL1005930C1.

As far as I understand it, the idea presented there is to encode each line of each frame of the video data using an identifier, such that identical lines have the same identifier. Instead of storing the image lines themselves, the system only stores the identifiers, so you get a form of deduplication of the data. There would be a similar system for the audio. I think the table for mapping identifiers back to the full video/audio data is build during encoding and is supposed to be part of encoded data, but maybe I'm wrong about that and the table is supposed to be pre-shared between encoder and decoder.

I see some fundamental problems with that approach.

The patent presents an example using small numbers as 1, 2, 3, 4 as identifiers. But even with the relatively low resolution of the time (the patent doesn't mention frame rate, but it does mention 625 rows (presumably because that's the number of lines in PAL), any video will have lots of different lines so there will have to be lots of different identifiers and using only small identifiers won't suffice. Storing all those identifiers will still take a substantial amount of storage.

The table for mapping identifiers back to the full video/audio data will also use quite some storage space.

I don't believe a working device was ever constructed. https://nl.wikipedia.org/wiki/Jan_Sloot claims that Roel Peper has a working encoding device, but even if true that is useless without a decoder (otherwise I can easily create an encoder that compresses any movie down to 1 byte or even 1 bit). The impression I have is that Sloot indeed didn't really care about the issues above, thinking they were implementation details that could be solved later, digging himself a hole he could never get out of.

permalink embed save report reply

[–] mathisdawkins 2 points 2 years ago

It seems impossible from an Information Theoretic point of view. It was never my strong side, so anyone more familiar with it could probably easily debunk this.

permalink embed save report reply

[–] mikeverity 1 point 2 years ago

As everyone pointed out, it is scammy and surely didn't work.

But theoretically, given enough time to decode (i.e.: an absurd amount of time) this may be possible with encryption or code, where the movie is converted in a program that generates the movie (or more realistically, a lossy version of it).

This is the basic principles used to generate 8k/64k demos, where about 90% of the reality of the world are encoded in much smaller form than their representations on screen.

As a (very absurd) thought exercise - if our universe is deterministic we could maybe describe a few initial conditions, some simple physical laws and then a spacetime coordinate to anything in the universe, to effectively encode it in much smaller form than describing the thing itself.

permalink embed save report reply

[–] CMDrunkenBreaking 1 point 1 year ago*

I studied this intriguing case for a few years.

What I read indicated that in the late-90s, it was able to play multiple movies at once using much less memory.

Showing 16 movies at once was more of a technical trick than anything. With traditional methods you would indeed use 16 movies separately. I believe Jan Sloot circumvented this, because his system just did a lookup table database to get pixel values, pixel blocks, and pixel rows. For two movies (split screen), there are no more database connections, or database queries necessary, than for showing one movie full screen.

The former CTO put out something saying it isn't about compression; it was a whole different way of thinking.

Indeed, Jan Sloot was able to get a maximum of 8x compression. Only more than that, you need a smarter and simple (as in the case of repair manuals, which include bikes, lines, fonts, that are very generic). SDCS was likely based on an earlier invention of Sloot, called Repobase.

Repobase was a collection of repair manuals in vector format, used by television repair people. It was heavily compressed with Sloot coding, due to its redundancy and low complexity. You could call for a key, and decrypt/decode a specific manual with that. So you don't have to pay for the entire catalogue, but only those manuals you'll ever be working on.

The novelty in compression, is that Sloot was able to compress all movies together, where current codecs only look at a single piece of media, and don't have an extremely long lookback (if two frames are exactly the same, but first frame is the beginning of the movie and second frame is at the end, then traditional compression would unlikely encode those frames separately. Sloot's system would encode the second frame with a pointer to the first frame. The intro of a season of episodes, would have to be stored only once in Sloot's system, traditional compression techniques don't refer to (parts) of other movies encoding. It's an interesting and useful novelty though, but nowhere near mentaining an 8kb key as the compression of a full-size movie, or talking of digital DRM, current instructions can encode full-size movies by sharing blocks).

The inventors were incredibly hoped up, but most lacked the technical knowledge to make sound judgment. Only inventors talked about compressing movies to 8kb's, but the company did little to dispel that, as hype was in their best interest.

[only find a reference to a demonstration to a Philips exec

The Sloot and his inventor were flown to Silicon Valley to demonstrate to big investment companies and technicians from Oracle. Later only his invention went to SV, and Sloot claimed they had opened up his invention (he had added tamper proof tape). A later invention was opened and shown to contain a hard disk, so perhaps they found that (and proposed as million-dollar hoax on seasoned IT by unknown TV repair man), or actually succeeded in reverse engineering the invention.

Afterward, a floppy disk was missing, so the whole thing is lost.

The floppy disk only had the decoder main key (how to go from compressed database to pictures on the screen, given 8kb key to decode relevant parts of the database — those that form a movie). The decoder is the same for the current invention to lost (unless you crack the key). But you can still rebuild the technology with your own encoder/decoder scheme and master key.

Sloot's "laboratory" in the attic was also completely cleaned out after his death. Perhaps some investors paid for this (broked a family member, or actually broke in).

An invention back then so much better than anything today.

It was technologically possible back then, as it is now. There are no hardware or network constraints in the 90s that would have made it impossible.

Figure that BitTorrent was invented around 2 years after Jan Sloot's death. Imagine a commercial BitTorrent, where seed stations are placed by many customers (you need massive scale for this, but with investment and cooperation with entertainment industry not impossible). The 8kb key becomes a Torrent file.

Maybe never mention the storage capability of the device (half just for the sake of algorithms and drivers, around 370MB). So let's assume a 16GB solid state disk. Very popular movies could be pre-loaded, and else, you'd have to wait a night to download it, or perhaps physically transfer the compressed movie data from a local video stores. Companies like Canal+ were using a TV channel to broadcast garbled movies, and you'd need a hardware key to decode it; going that route would risk the need for downloading over phone line — the inventory could be continually updated that way, at scale, at low cost for the consumer.

The result would be a commercial PirateBay with DRM. Deals with music studios and software companies would be made, so every DVD player now contained 16GB of the most popular music of the time, unlockable by a specific key, every desktop computer would come with Adobe, Autocad, Coreldraw etc. and you could install from a key, instead of shipping a DVD. I think that's the scale that Roel Peper saw, which justified propositions of a billion dollar company. They'd go the Dolby Surround route, adding a small SDCS logo to hardware supporting this new distribution model.

Nothing left in terms of the underlying theory or other info to sketch out how it was done?

I think at least some people know all about the theory or info, enough to replicate a peer-to-peer digital-entertainment-system setup box. Perhaps some parts of this technology made it to other companies. At least now, the bandwidth of the internet is enough for 1:1 streaming, and this solution would be outdated. But also think of the success of Popcorn time (illegal, but just add DRM and industry cooperation) and Netflix or Akamai (also does load balancing and peer-to-peer sharing).

It all sounds like it was going to be a scam

The scammy parts are more in the hype, and in the investors. This was pre-tech bubble. Some investors were shady/part of the underground. Early investors benefited from unduly hype (compress videos down to 8kb!) to make their investment worth more. I think it is more likely that Jan Sloot actually had something of substance, rather than that an obscure TV repair man was able to swindle the absolute top of IT professors and investment funds. Roel Peper publicly said his CEO job at Philips to work on this venture. Would the Philips executive board risk their reputation on his wild hoax demonstrations? Perhaps he used a suitcase with a radio transmitter?

What if someone invented the core beginnings of the Pirate Bay, Netflix, and Akamai, when BitTorrent did not exist yet, Netflix was sending DVDs by mail, and Akamai did not invent peer-to-peer sharing for content distribution? How much could that company be worth after 20 years? Where would the data distribution industry be? Sales of DVDs and floppies decreased by 50%? Media distribution has a very long tail and just a few movies nearly everyone sees. How many can you serve and how much data distribution can you negate, if you pre-store just the top 16 Bollywood movies of the year 1999? A lot!

Even at the time, information theorists claimed Bits compression was provably impossible. Roel Peper was professor of IT and must have surely heard of the Pigeon Hole principle. You don't get the founder of Kleiner Perkins Caufield & Byers to state you are going to be a very very rich man, with a theoretically impossible invention. Sloot deceased inside a Reddit thread. Some technicians stated that storing the movies on the device and unlocking them with a key was cheating or that it would make the idea useless. But these technicians were from the commercial cable that The Fifth Force (company that owned the tech) was thinking at. Such a DRM solution and media distribution system is far far from useless.

Neither it was a hoax or a true invention, it is likely that multiple intelligence agencies were actively seeking more information and influence, or that rich investors had access to private agencies or the underground (they get rich from illegal ways). Sloot himself got fairly paranoid, and I don't think that was completely unjustified. This is what happens with $1+ billion $ estimated companies and take private sector investors to meet with the top of SV. And when you are Sloot, all you can do is become a fan of John Graham spy novels, and try to learn about things that go unspoken. No doubt, he contributed to the whole mystery/intrigue of the case of Sloot and his invention. If it was a hoax, and rich investors got burned, that's certainly stuff that people get professionally killed over (injected with a heart-attack inducing agent in the car and dropped in his garden). If it was the lost Bill Gates compression digital library, and he stole the information, then it certainly wasn't lost at all, and agencies would have stolen this through economic espionage, and sued to improve their economies (still possible with the distribution system, but less likely).

If you go against the mainstream, certainly in well-established fields that attract the weird thinkers, such as quantum physics, coding theory, and standardized distribution models, then either you will be on the fringe (and judged as crazy), or you go on to win a Nobel prize. So Sloot's invention, no matter how massive and positive, will always be regarded as fringe and without authority. This paper reference by Holtz is far from authoritative either, but try to imagine what would happen when you apply these theories with a lot of funding:

permalink embed save report reply

[–] DirtyLoundMouthedKing 1 point 1 year ago

Multimedia data may be transmitted or stored either according to the classical Shannon information theory or according to the newer Autosophy information theory. Autosophy algorithms combine very high "lossless" data and image compression with virtually unbreakable "codebook" encryption. Shannon's theory treats all data items as "quantities", which are converted into binary digits (bit), for transmission in meaningless bit streams. Only "lossy" data compression is possible. A new "Autosophy" theory was developed by Klaus Holtz in 1974 to explain the functioning of natural self-assembling structures, such as chemical crystals or living trees. The same processes can also be used for growing self-assembling data structures, which grow like data crystals or data trees in electronic memories. This provides true mathematical learning algorithms, according to a new Autosophy information theory. Information in essence is only that which can be perceived and which is not already known by the receiver. The transmission bit rates are dependent on the data content only. Applications already include the V.42bis compression standard in modems, the gif and tif formats for lossless image compression, and Autosophy Internet television. A new 64bit data format could allow all future communications compatible and solve the Internet's Quality of Service (QoS) problems.

http://www.autosophy.net/SPEE4Qed.pdf

permalink embed save report reply

[–] 1 point 1 year ago*

You know what? I have a beautiful awesome bridge to sell to you.

I saw that Youtube video about that so called "lost" invention and it claimed that this dude was able to compress a 16GB video down to 8kb. That was about one million the size. Using your Boeing 747 analogy you could build the same aircraft that is only one millionth of 404k lb / 183k KG. Not 1/5. One 2 hour movie consists of almost 180 000 individual pictures.

permalink embed save report reply

[–] shadgreene comment score below threshold (10 children)

[–] dmandmmmm 1 point 5 months ago

Yep we can consider lost that a valuable invention each year people change because of memory and downgrade operating system

permalink embed save report reply

[–] 1 point 1 year ago*

I instantly thought about an algorithm who calculates fractals.

permalink embed save report reply

Log in or sign up in seconds | English

Bro just was submitted on 28 Jul 2020
122 points (96% upvoted)
shortlink: https://redd.it/hu5t3d

Submit a new link
Submit a new text post

Get Reddit premium
Get an ad-free experience with special features and directly support Reddit.
Get Reddit Premium

compsci
2,111,340 readers
187 users here now
Welcome Computer Science researchers, students, professionals, and enthusiasts!
We share and discuss content that computer scientists find interesting.

Guidelines
Self-posts and Q&A threads are welcome, but we prefer high quality posts focused directly on graduate level CS material. We discourage most posts about introductory material, how to study CS, or about careers. For those topics, please post in one of the subreddits in the sidebar instead.

Want to study CS or learn programming?
Read the original free Structure and Interpretation of Computer Programs (or see the online conversion of SICP)

Related subreddits
Other topics are likely better suited for:
/r/cscareerquestions: Job, internships, etc.
/r/askscience/computerscience
/r/learnprogramming: Resources for learning how to code.
/r/compressionideas: A collection of video compression ideas, and educational computer science topics
/r/codegolf
/r/math: Despite popular misconceptions, Computer Science is mostly about math.
/r/programming: Not for those occasionally implement things.
/r/algorithms: Another computer science subreddit, but more specifically, we will fight to the death.
/r/programminglanguages
/r/rust
/r/machinelearning
/r/hardware
/r/dsp: Digital signal processing
/r/tinycode: Cool algorithms, tiny implementations.
/r/cryptography
/r/CryptoCurrency
Other online communities:
CS Theory Stack Exchange
Computer Science Stack Exchange
HaskellOptimize: Machine Learning
Stackexchange Ultimate: Programming Languages

If you are new to Computer Science please read our FAQ before posting. A list of book recommendations from our community for various topics can be found here.

MODERATORS
r/compsci is a community for CS people

MESSAGE THE MODS
Moderator list hidden. Learn More

Discussions in r/compsci
How to be better at solving problems ?

about
Blog
Careers
advertising
careers

help
site rules
frequently asked questions
reddiquette
mod guidelines
contact us

apps & tools
Reddit for iPhone
Reddit for Android
mobile website

<3
reddit premium
reddit coins

Use of this site constitutes acceptance of our User Agreement and Privacy Policy. © 2022 reddit inc. All rights reserved.

Cookies help us deliver our Services. We only use essential cookies. Learn more about Reddit's use of cookies.
ACCEPT ESSENTIAL COOKIES