Defining Computationally Minimal Art (Or, taking the "8" out of "8-bit")

Ville-Matias Heikkilä
a.k.a. viznut/pwp

2010-03-15

1 Introduction

"Low-tech" and "8-bit" are everywhere nowadays. Not only are the related underground subcultures thriving, but "retrocomputing esthetics" seems to pop up every now and then in mainstream contexts as well: obvious chip sounds can be heard in many pop music songs, and there are many examples of "old video game style" in TV commercials and music videos. And there are even "pixel-styled" physical products, such as the pictured watch sold by the Japanese company *&design*. I'm not a grand follower of popular culture, but it seems to me that the trend is increasing.

The most popular and widely accepted explanation for this phenomenon is the "nostalgia theory", i.e. "People of the age group X are collectively rediscovering artifacts from the era Y". But I'm convinced that there's more to it – something more profound that is gradually integrating "low-tech" or "8-bit" into our mainstream cultural imagery.

Many people have became involved with low-tech esthetics via nostalgia, but I think it is only the first phase. Many don't experience this phase at all and jump directly to the "second phase", where pixellated graphics or chip sounds are simply enjoyed the way they are, totally ignoring the historical baggage. There is even an apparent freshness or novelty value for some people. This happens with audiences that are "too young" (like the users of Habbo Hotel) or otherwise more or less unaffected by the "oldskool electronic culture" (like many listeners of pop music).

Since the role of specific historical eras and computer/gaming artifacts is diminishing, I think it is important to provide a neutral conceptual basis for "low-tech esthetics"; an independent and universal definition that does not refer to the historical timeline or some specific cultural technology. My primary goal in this article is to provide this definition and label it as "Computationally Minimal Art". We will also be looking for support for the universality of Computationally Minimal Art and finding ur-examples that are even older than electricity.

2 A definition: Computationally Minimal Art

Once we strip "low-tech esthetics" from its historical and cultural connections, we will be left with "pixellated shapes and bleepy sounds" that share an essential defining element. This element stems from what is common to the old computing/gaming hardware in general, and it is perfectly possible to describe it in generic terms, without mentioning specific platforms or historical eras.

The defining element is LOW COMPUTATIONAL COMPLEXITY, as expressed in all aspects of the audiovisual system: the complexity of the platform (i.e. the number of transistors or logic gates in the hardware), the complexity of the software (i.e. the length in bits of the program code and static data), as well as the time complexity (i.e. how many state changes the computational tasks require). A more theoretical approach would eliminate the differentiation of software and hardware and talk about description/program length, memory complexity and time complexity.

There's little more that needs to be defined; all the important visible and audible features of "low-tech" emerge from the various kinds of low complexity. Let me elaborate with a couple of examples:

- A low computing speed leads to a low number of processed and output bits per time frame. In video output, this means low resolutions and limited color schemes. In audio output, this means simple waveforms on a low number of discrete channels.
- A short program+data length, combined with a low processing speed, makes it preferrable to have a small set of small predefined patterns (characters, tiles, sprites) that are extensively reused.
- A limited amount of temporary storage (emerging from the low hardware complexity) also supports the former two examples via the small amount of available video memory.
- In general, the various types of low complexity make it possible for a human being (with some expertise) to "see the individual bits with a naked eye and even count them".

In order to complete the definition, we will still have to know what *low* means. It may not be wise to go for an arbitrary threshold here ("less than X transistors in logic, less than Y bits of storage and less than Z cycles per second"), so I would like to define it as "the lower the better". Of course, this does not mean that a piece of low-tech artwork would ideally constitute of one flashing pixel and static square-wave noise, but that the most essential elements of this artistic branch are those that persist the longest when the complexity of the system approaches zero.

Let me therefore dub the idealized definition of “low-tech art” as *Computationally Minimal Art* (CMA).

To summarize: *Computationally Minimal Art is a form of discrete art governed by a low computational complexity in the domains of time, description length and temporary storage. The most essential features of Computationally Minimal Art are those that persist the longest when the various levels of complexity approach zero.*

3 How to deal with the low complexity?

Traditionally, of course, low complexity was the only way to go. The technological and economical conditions of the 1970s and 1980s made the microelectronic artist bump into certain “strict boundaries” very soon, so the art needed to be built around these boundaries regardless of the artist’s actual esthetic ideals. Today, on the other hand, immense and virtually non-limiting amounts of computing capacity are available for practically everyone who desires it, so computational minimalism is nearly always a conscious choice. There are, therefore, clear differences in how the low complexity has been dealt with in different eras and disciplines.

I’m now going to define two opposite approaches to low complexity in computational art: optimalism (or “oldschool” attitude), which aims at pushing the boundaries in order to fit in “as much beauty as possible”, and reductivism (or “newschool” attitude), which idealizes the low complexity itself as a source of beauty.

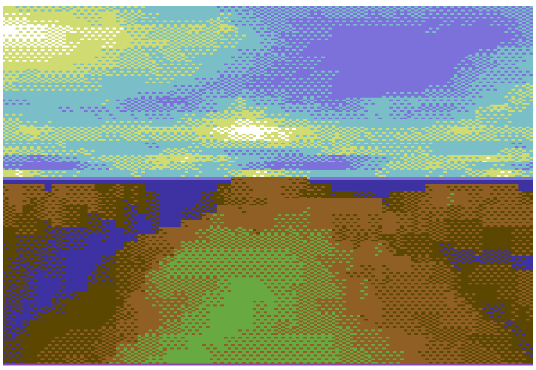
Disclaimer: All the exaggeration and generalization is intentional! I’m intending to point out differences between various extremities, not to portray any existing “philosophies” accurately.

3.1 Optimalism

Optimalism is a battle of maximal goals against a minimal environment. There are absolute predefined boundaries that provide hard upper limits for the computational complexity, and these boundaries are then pushed by fitting as much expressive power as possible between them. This approach is the one traditionally applied to mature and static hardware platforms by the video game industry and the demoscene, and it is characterized by the appreciation of optimization in order to reach a high content density regardless of the limitations.

A piece of traditional European-style pixel graphics (*Frog, Landscape and a lot of Clouds*[2] by *oys*) exemplifies many aspects of optimalism. The resolution and color constraints of a video mode (in this case, non-tweaked C-64 multicolor) provide the hard limits, and it is the responsibility of the artist to fill up the space as wisely and densely as possible. Large single-colored areas would look "unfinished", so they are avoided, and if it is possible to fit in more detail or dithering somewhere, it should be done. It is also avoidable to leave an available color unused – an idea which leads to the infamous *Dutch color scheme*[7] when applied to high/truecolor video modes.

When applied to chip music, the optimalist dogma tells, among all, to fill in all the silent parts and avoid "simple beeps". Altering the values of as many sound chip registers per frame as possible is thought to be efficient use of the chip. This adds to the richness of the sound, which is though to correlate with the quality of the music.



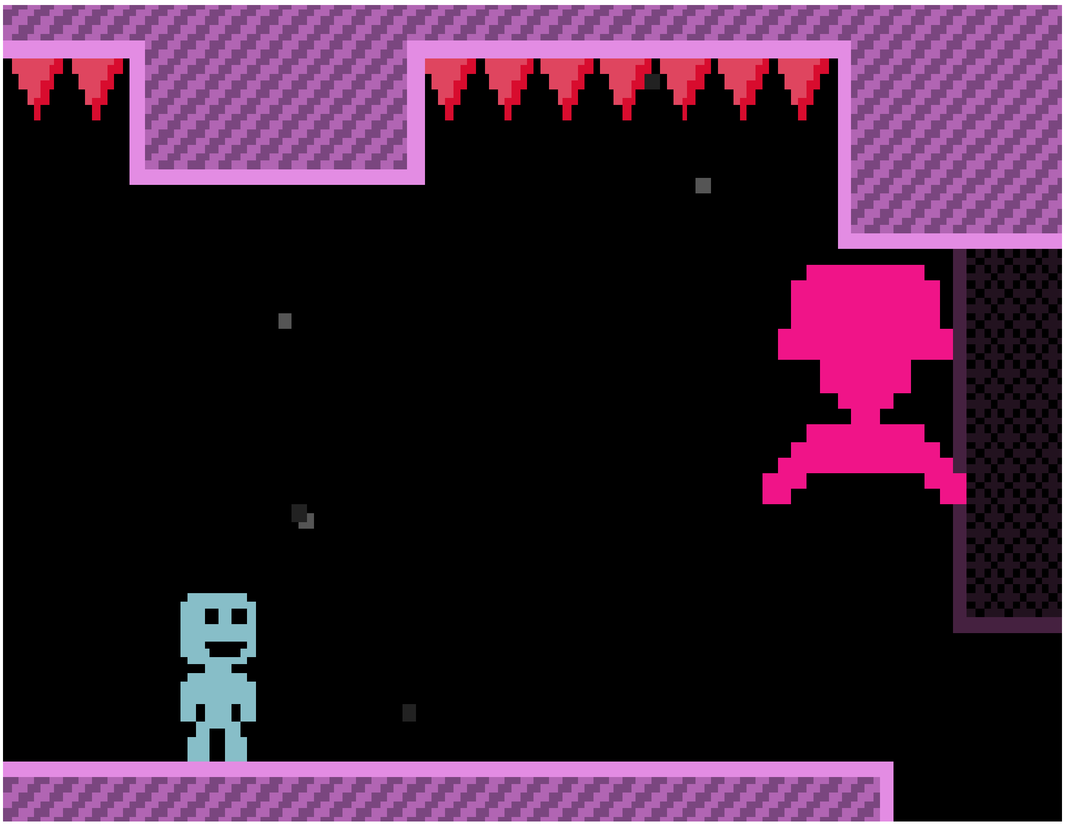
On platforms such as the Commodore 64, the demoscene and video game industry seem to have been having relatively similar ideals. Once an increased computing capacity becomes available, however, an important difference between these cultures is revealed. Whenever the video game industry gets more disk space or other computational resources, it will try to use it up as aggressively as possible, without starting any optimization efforts until the new boundaries have been reached. The demoscene, on the other hand, values optimality and content density so much that it often prefers to stick to old hardware or artificial boundaries in order to keep the "art of optimality" alive. The screenshot is from the 4K demo *Artefacts*[3] by *Plush* (C-64).

Despite the cultural differences, however, the core esthetic ideal of optimalism is always "bigger is better"; that an increased perceived content complexity is a requirement for increased beauty. Depending on the circumstances, more or less pushing of boundaries is required.

3.2 Reductivism

Reductivism is the diagonal opposite of optimalism. It is the appreciation of minimalism within a maximal set of possibilities, embracing the low complexity itself as an esthetic goal. The approach can be equated with the artistic discipline of minimal art, but it should be remembered that the idea is much older than that. Pythagoras, who lived around 2500 years ago, already appreciated the role of low complexity – in the form of mathematical beauty such as simple numerical ratios – in music and art.

The reductivist approach does not lead to a similar pushing of boundaries as optimalism, and in many cases, strict boundaries aren't even introduced. Regardless, a kind of pushing is possible – by exploring ever simpler structures and their expressive power – but most reductivists don't seem to be interested in this aspect. It is usually enough that the output comes out as "minimal enough" instead of being "as minimal as possible".



The visuals of the recent acclaimed Flash-based platformer game, *VVVVVV* [5], are a good example of computational minimalism with a reductivist approach. The author, *Terry Cavanagh*, has not only chosen a set of voluntary "restrictions" (reminiscent of mature computer platforms) to guide the visual style, but keeps to a reductivist attitude in many other aspects as well. Just look at the "head-over-heels"-type main sprite – it is something that a child would be able to draw in a minute, and yet it is perfect in the same iconic way as the Pac-Man character is. The style totally serves its purpose: while it is charming in its simplicity and downright naivism, it shouts out loud at the same time: "Stop looking at the graphics, have fun with the actual game instead!"

Although reductivism may be regarded as a "newschool" approach, it is possible to find some slightly earlier examples of it as well. The graphics of the 1986 computer game Thrust, for example, has been drawn with simple geometrical lines and arcs. The style is reminiscent of older vector-based arcade games such as Asteroids and Gravitar, and it definitely serves a technical purpose on such hardware. But on home computers with bitmapped screens and sprites, the approach can only be an esthetical one.

3.3 Optimalism versus Reductivism

Optimalism and reductivism sometimes clash, and an example of this can be found in the chip music community. After a long tradition of optimalism thru the efforts of the video game industry and the demoscene, a new kind of cultural branch was born. This branch, sometimes mockingly called *cheaptoon*, seems to get most of its kicks from the unrefined roughness of the pure squarewave rather than the pushing of technological and musical boundaries that has been characteristic of the "oldschool way". To an optimalist, a reductivist work may feel lazy or unskilled, while an optimalist work may feel like "too full" or "too refined" to a reductivist mindset.

Still, when working within constraints, there is room for both approaches. Quite often, an idea is good for both sides; a simple and short algorithm, for example, may be appreciated by an optimalist because the saved bytes leave room for something more, while a reductivist may regard the technical concept as beautiful on its own right.

4 Comparison to Low-Complexity Art

Now I would like to compare my definition of Computationally Minimal Art to another concept with a somewhat similar basis: *Jürgen Schmidhuber's Low-Complexity Art*.[1]

While CMA is an attempt to formalize "low-tech computer art", Schmidhuber's LCA comes from another direction, being connected to an ages-old tradition that attempts to define beauty by mathematical simplicity. The specific mathematical basis used in Schmidhuber's theory is Kolmogorov complexity, which defines the complexity of a given string of information (such as a picture) as the length of the shortest computer program that outputs it. Kolmogorov's theory works on a high level of generalization, so the choice of language does not matter as long as you stick to it.

Schmidhuber sees, in "down-to-earth coder terms", that the human mind contains a built-in "compressor" that attempts to represent sensory input in a form as compact as possible. Whenever this compression process succeeds well, the input is perceived as esthetically pleasing. It is a well-studied fact that people generally perceive symmetry and regularity as more beautiful than unsymmetry and irregularity, so this hypothesis of a "mental compressor" cannot be dismissed as just an arbitrary crazy idea.

Low-Complexity Art tests this hypothesis by deliberately producing graphical images that are as compressible as possible. One of the rules of LCA is that an "informed viewer" should be able to perceive the algorithmic simplicity quite easily (which also effectively limits the time complexity of the algorithm, I suppose). Schmidhuber himself has devised a system based on indexed circle segments for his pictures.

The above picture is from *Superego*[4], a tiny pc demo I made in 1997. The picture takes some tens of bytes and the renderer takes less than 100 bytes of x86 code. Unfortunately, there is only one such picture in the demo, although the 4K space could have easily contained tens of pictures. This is because the picture design process was so tedious and counter-intuitive – something that Schmidhuber has encountered with his own system as well. Anyway, when I encountered Schmidhuber's LCA a couple of years after this experiment, I immediately realized its relevance to size-restricted demoscene productions – even though LCA is clearly a reductivist approach as opposed to the optimalism of the mainstream demoscene.

What Low-Complexity Art has in common with Computationally Minimal Art is the concern about program+data length; a minimalized Kolmogorov complexity has its place in both concepts. The relationship with other types of complexity is different, however. While CMA is concerned about all the types of complexity of the audiovisual system, LCA leaves time and memory complexity out of the rigid mathematical theory and into the domain of a "black box" that processes sensory input in the human brain. This makes LCA much more theoretical and psychological than CMA, which is mostly concerned about "how the actual bits move". In other words, LCA makes you look at visualizations of mathematical beauty and ignore the visualization process, while CMA assigns an utmost importance to the visualizer component as well.

5 Psychological considerations

Now, an important question: why would anyone want to create Computationally Minimal Art for purely esthetical reasons – novelty and counter-esthetic values aside? After all, those "very artificial bleeping sounds and flashing pixels" are quite alien to an untrained human mind, aren't they? And even many fans admit that a prolonged exposure to those may cause headache.

It is quite healthy-minded to assume that the perception mechanisms of the human species, evolved during hundreds of millions of years, are "optimized" for perceiving the natural world, a highly complex three-dimensional environment with all kinds of complex lighting and shading conditions. The extremely brief technological period has not yet managed to alter the "built-in defaults" of the human mind anyhow. Studies show, for example, that people all over the world prefer to be surrounded by wide-open landscapes with some water and trees here and there – a preference that was fixed to our minds during our millions of years on the African savannah.

So, the untrained mind prefers a photorealistic, high-fidelity sensory input, and that's it? No, it isn't that simple, as the natural surroundings haven't evolved independently from the sensory mechanisms of their inhabitants. Fruits and flowers prefer to be symmetric and vivid-colored because animals prefer them that way, and animals prefer them that way because it is beneficial for their survival to like those features, and so on. The natural world is full of signalling which is a result of millions of years of coevolutionary feedback loops, and this is also an important source for our own sense of esthetics. (The fish in the picture, by the way, is a Synchiropus splendidus, photographed by Luc Viatour[**?**].)

I'm personally convinced that natural signalling has a profound preference for low complexity. Symmetries, regularities and strong contrasts are important because they are easy and effortless to detect, and the implementation requires a relatively low amount of genetic coding on both the “transmitter” and “receiver” sides. These are completely analogous to the various types of computational complexity.

So, why does enjoying Computationally Minimal Art require “mental training” in the first place? I think it is not because of the minimality itself but because of certain pecularities that arise from the higher complexity of the natural world. We can’t see individual atoms or even cells, so we haven’t evolved a built-in sense for pixel patterns. Also, the sound generation mechanisms in nature are mostly optimized to the constraints of pneumatics rather than electricity, so we don’t really hear squarewave arpeggios in the woods (although some birds may come quite close).

But even though CMA requires some special adjustment from the human mind, it is definitely not alone in this area. Our cultural surroundings are full of completely unnatural signals that need similar adjustments. Our music uses instruments that sound totally different from any animal, and practically all musical genres (apart from the simplest lullabies, I think) require an adjustment period. So, I don’t think there’s nothing particularly “alien” in electronic CMA apart from the fact that it still hasn’t yet integrated in our mainstream culture.

6 CMA unplugged

The final topic we cover here is the extent where Computationally Minimal Art, using our strict definition, can be found. As the definition is independent from technology, it is possible to find ur-examples that predate computers or even electricity.

In our search, we are ignoring the patterns found in the natural world because none of them seem to be discrete enough – that is, they fail to have “human-countable bits”. So, we’ll limit ourselves to the artifacts found in human culture.

Embroidery is a very old area of human culture that has its own tradition of pixel patterns. I guess everyone familiar with electronic pixel art has seen cross-stitch works that immediately bring pixel graphics in mind. The similarities have been widely noted, and there have been quite many craft projects inspired by old video games.[6] But is this just a superficial resemblance or can we count it as Computationally Minimal Art?

Cross-stitch patterns are discrete, as they use a limited set of colors and a rigid grid form which dictates the positions of each of the X-shaped, single-colored stitches. “Individual bits are perceivable” because each pixel is easily visible and the colors of the “palette” are usually easy to tell apart. The low number of pixels limits the maximum description length, and one doesn’t need to keep many different things in mind while working either. Thus, cross-stitch qualifies all the parts of the definition of Computationally Minimal Art.

What about the minimization of complexity? Yes, it is also there! Many traditional patterns in textiles are actually algorithmic or at least highly repetitive rather than “fully hand-pixelled”. This is somewhat natural, as the old patterns have traditionally been memorized, and the memorization is much easier if mnemonic rules can be applied.

There are also some surprising similarities with electronic CMA. Many techniques (like knitting and weaving) proceed one complete row of “pixels” at a time (analogous to the raster scan of TV-like displays), and often, the set of colors is changed between rows, which is corresponds very well to the use of raster synchronization in oldschool computer graphics. There are even peculiar technique-specific constraints in color usage, just like there are similar constraints in many old video chips.

The picture on the right[8] depicts a pillow knitted with the traditional Fair Isle technique. It is apparent that there are two colors per “scanline”, and these colors are changed between specific lines (compare to rasterbars). The patterns are based on sequential repetition, with the sequence changing on a per-scanline basis.

Perhaps the most interesting embroidery patterns from the CMA point of view are the oldest ones that remain popular. During centuries, the traditional patterns of various cultures have reached a kind of multi-variable optimality, minimizing the algorithmical and technical complexity while maximizing the eye-pleasingness of the result. These patterns may very well be worth studying by electronic CMA artists as well. Things like this are also an object of study for the field of ethnomathematics, so that’s another word you may want to look up if you’re interested.

What about the music department, then? Even though human beings have written music down in discrete notation formats for a couple of millennia already, the notes alone are not enough for us. CMA emphasizes the role of the rendering, and the performance therefore needs to be discrete as well. As it seems that every live performance has at least some non-discrete variables, we will need to limit ourselves to automatic systems.

The earliest automatic music was mechanical, and arguably the simplest conceivable automatic music system is the musical box. Although the musical box isn't exactly discrete, as the barrel rotates continuously rather than stepwise, I'm sure that the pins have been positioned with an engineer's accuracy as guided by written music notation. So, it should be discrete enough to satisfy our demands, and we may very well declare the musical box as being the mechanical counterpart of chip music.

7 Conclusion

I hope these ideas can provide food for thought for people interested in the various forms of "low-tech" electronic art as well as computational art or "discrete art" in general. I particularly want people to realize the universality of Computationally Minimal Art and how it works very well outside of the rigid "historical" contexts it is often confined into.

I consciously skipped all the cultural commentary in the main text on my quest for proving the universality of my idea, so perhaps it's time for that part now.

In this world of endless growth and accumulation, I see Computationally Minimal Art as standing for something more sustainable, tangible and crafty than what the growth-oriented "mainstream cultural industry" provides. CMA represents the kind of simplicity and timelessness that is totally immune to the industrial trends of fidelity maximization and planned obsolescence. It is something that can be brought to a perfection by an individual artist, without hiring a thousand-headed army of specialists.

As we are in the middle of a growth phase, we can only guess what kind of forms Computationally Minimal Art will get in the future, and what kind of position it will eventually acquire in our cultural framework. We are living interesting times indeed.

References

[1] Jürgen Schmidhuber. Low-Complexity Art. Leonardo, Journal of the International Society for the Arts, Sciences, and Technology, 30(2):97-103, MIT Press, 1997. ftp://ftp.idsia.ch/pub/juergen/locoart.pdf

[2] oys of Arsenic, Oxyron. Frog, Landscape and a Lot of Clouds. C64 graphics released in 16 November 2006. http://noname.c64.org/csdb/release/?id=43591

[3] Plush (Krill, Fanta). Artefacts. Commodore 64 demo. **Breakpoint 2006** demoparty, C-64 4K intro competition. http://www.pouet.net/prod.php?which=24450

[4] Viznut/PWP. Superego. PC 4K intro released in October 1997. http://pouet.net/prod.php?which=34585

[5] Terry Cavanagh. VVVVVV. Flash game. 2009. http://thelettervsixtim.es/

[6] Spritestitch. A video game inspired craft weblog. http://www.spritestitch.com/

[7] Optimus. Dutch Color Scheme??? A thread on **pouet.net**. http://www.pouet.net/topic.php?which=332&page=1

[8] Sarah Peasley. Introduction to Fair Isle. http://www.sarahpeasley.com/

[9] Luc Viatour, photographer. http://www.lucnix.be/