Humans Create, Occasionally.
Computers Operate, Always.
Self-Evident & Trivial

Frieder Nake
nake@informatik.uni-bremen.de

This essay is a bit of a grumble. I will try to be subjectively trivial. Not exactly scientific. Some important names will be mentioned. If some readers enjoy reading this, it's enough from my standpoint. The title exactly says what I want to say. The essay is only longer.

In recent years, combinations of terms have become popular where one component is „computational" or some relative of it, whereas the other is „aesthetics" or „art" or „creativity", etc. The Dagstuhl seminar, to which this personal note owes its appearance, had the number 09291 and the title Computational Creativity: an Interdisciplinary Approach.

Good old C. P. Snow would probably like such terms as signs of a conscious effort to bridge the gap between the „two cultures": from the scientific culture (home of „computational") to the literary („creativity"), and back again[1]. We have seen computational aesthetics, computer art (the oldest of them all), aesthetic computing, machine aesthetics, creative computation, computational creativity, and more.

It is always absolutely boring to ask for a precise definition – a definition, not an explanation[2] – of a complex term, in hope of making our ideas clear[3], even though philosophers throughout the centuries have tried nothing but exactly this, and not only philosophers did so. How much have we, when we were young, indulged in endless debates about the meaning of a word. Ludwig Wittgenstein has given us the ultimate answer to all[4] questions of the kind: „What is x?" He says: the meaning of a word is its use in the language (Philosophical Investigations, no. 43). Clear, helpful, and pragmatic.

Don't think, look – is Wittgenstein's advice to his fellow philosophers. Look, that is, how real living people are using the word you want to understand. They will, quite likely, use it differently from place to place, and from time to time.

I guess, it is allowed to extend the contextual condition of „… in the language" to „in society". It is a society that develops a language. And it is a language that determines to a large extent what a society is. From a certain perspective, and certainly from the perspective of C. P. Snow, language and society are more or less the same.

Apply Wittgenstein's insight to the torture sessions professors and teachers like so much. They arrange for meetings with their students that they call „exams". The only purpose those meetings appear to have is to fail those poor guys and girls. They fail them if students upon certain questions do not pro-

---

[1] If C. P. Snow is really unknown to you, google him, and his idea of the Two Cultures. The current Third Culture is kind of an answer to it.

[2] We encounter definitions everywhere and, therefore, we may believe it is possible to define a term without doubt. However, only mathematics is mildly successful in this endeavor. This is so because mathematical objects are creations of the mind.

[3] An implicit reference to Charles S. Peirce's essay „How to make our ideas clear" which, to many, is not very clear. But not to all.

[4] Well, not quite „all" of them. For many cases of the use of the word „meaning", not for all, Wittgenstein says.

duce exactly those definitions that the professor had told them. He now wants to hear his definitions mirrored by the student. Creativity?

Why do I write this as some sort of an introduction into something I don't really know what it should be and where it will take me? I write it because my first reaction, upon receiving the invitation to the Dagstuhl seminar on computational creativity – an invitation that I immediately take very seriously, because it is an invitation to Dagstuhl, – my first reaction was: oh no, not again!

I later realized this reaction of mine was a reaction coming from older thinking, from times of the merciless lust for discussion of terms. I went to Dagstuhl nevertheless. My younger part triumphed, and the decision was not only because great names were announced whose owners would be marvelous to spend a full week with, no matter how the topic of the meeting would be spelled out. I did not regret a single minute of the stay and effort. If a weird and odd title of a seminar is capable of bringing together an interesting and fascinating bunch of personalities, it is a good title. To sound odd does not exclude a title from being good.

Creativity for me has become one of those horrors that I would rather avoid having around. The more people in a group discuss creativity, the more you may safely assume that their own level of creativity is low. Imagine a job where the boss in the morning, with a bright smile on his face, greets you cheerfully: „Well, folks, let's be creative today!" Would you not rather turn around, go home, and take another nap?

The poor young professionals nowadays in the so-called creative professions have all my sympathies. Should they not be allowed to sit and read, or prepare a cup of wonderfully smelling, strong espresso? Instead, they are supposed to be creative all the time just becaus the company's name promises this.

In all modesty, I believe I had one or two, certainly not many more, moments in my life when I was creative in some serious sense of the word. The funny thing about this idea is, I don't know very well when those moments (outbursts?) have happened, and where it was, and what I was doing. I really think, what I was doing was nothing but my regular and everyday work. To exist as a human being (thus not as a moron), is to be active for a purpose. And to be active for a purpose is almost the same as being creative. You do something, and it results in a change of state in the environment. Isn't that creation, the beginning of it, anyhow?

Paul Klee, the artist and teacher, belongs to my flock of good friends. He was, of course, already dead when I heard or read his name for the first time. But most of our friends are dead when we get to know them. Isn't that so?

So when I had discovered Paul Klee as one of my friends, I read what he had said in 1928 when he was working at the Bauhaus, and this was in a context of, then, great relevance to many artists who were trying to better understand the relative weights of construction, exactness, mathematics, strict rules as opposed to intuition, vagueness, psychology, permitted aberrations. Paul Klee said and wrote this:

`We construct and construct, and yet intuition still has its uses.`

Klee had, of course, said something different, only similar to this translation a year later from German into English. His words had been:

`Wir konstruieren und konstruieren, und doch ist Intuition immer noch eine gute Sache.`

The uses and the Sache. What a difference! Miles between the two when you start thinking about it. The impossibility of translation. But its absolute necessity, at the same time. It cannot work but we must do it. Umberto Eco has recently published a thick volume on translation[5]. Play it again, Sam, only slightly different. There is, in a strict sense, no translation. There is no precision. It's all sloppy out there. Translation is saying something different, in a different language, and different culture. Translation is creation by inspiration. We are acting intuitively. The less we understand what is happening when we act, the more we are inclined to call the action an act of creativity.

Paul Klee puts the rationalistic mind and action in relation, and partial opposition to something he calls intuition. He knows, this is okay. He knows what intuition is, what it is to him. We know also. When, however, we are asked to explicitly say what we implicitly know, we are in trouble. We fail, or we are failed by teachers.

Here is a paraphrase of Paul Klee's phrase (that I have forcefully extracted from its context and thus stripped off a lot of its meaning):

`We create and create, and yet just doing something may still be quite useful.`

It is often a good idea to just let people do what they want to do. It may later appear as if they had been creative during those moments. Surprise is one of the nice features of creativity.

Computational creativity is obviously something that, when it happens, makes someone exclaim: oh, look at this, he has just created it! An aspect of surprise may be – must be! – present in computational creativity (if that term makes sense at all).

Even more obviously, computational creativity involves some kind of computation. It is a sort of creativity that, for some reason or other, depends on computers doing something. Something has been externalized, must have been objectified before, or else no computer would be capable of doing it. That something is in the state of an other – an other that exerts an influence on a situation of human activity, and that activity depends, to an important extent, on the computer's operation. If that activity turns out to be „creative" (i.e. surprising), we may be inclined to speak of „computational creativity".

Syntactically, the words „computational creativity" are structured like „generative art" or „generative aesthetics". These terms once had a good sound (and I hear, generative design has recently become popular). „Generative art" started into its career on the 5th of February, 1965. On this day, the first exhibition of algorithmic art opened at the Studiengalerie of TH Stuttgart (now, University of Stuttgart). The mathematician, Georg Nees, working for the Siemens Corporation, had written short programs in Algol60 whose output controlled an automatic drawing machine[6]. Nees was asked to call those drawings „statistical graphics" in order to avoid the word „art". It was not to be used in an industrial context, the logic of the two cultures required. Was that an indication that some innovation had happened, an act of creativity had occured?

The exhibition was shown in a seminar room used by the institute of philosophy. Max Bense, who resided there, regularly used it for experimental exhibitions of concrete art. Nees' show was definitely experimental[7]. Bense himself wrote a short essay for the occasion. He gave it the title, „Projects of genera-

---

[5] Umberto Eco: Experiences in translation. Various editions and translations. 2001 and later

[6] The machine was a high precision flat-bed plotter, as the US-Americans called it. It was the Graphomat Z64, the last construction by the computer pioneer, Konrad Zuse.

[7] Later the same year, in April and November, 1965, two other exhibitions were put up of algorithmic art: A. Michael Noll at Howard Wise in New York, and the author at Wendelin Niedlich in Stuttgart.

tive aesthetics". It was later translated into English, and published several times. In retrospect, I consider it the manifesto of algorithmic art.[8]

By the way, Georg Nees became the first, as far as I can tell, three years later, to earn a doctoral degree (in philosophy) about a topic from digital art. The thesis was published by Siemens in 1969, and republished as a facsimile in 2006.[9]

In the mid-1960s, if you were thinking of doing something with the help of a computer, your thinking had to concentrate on algorithms. You had to model and describe your task in such a way that even a machine could carry it out. To this end, the description must be precise, unambiguous, effective, operational. Algorithmic thinking during those years was thinking in terms of individual, isolated algorithms. You arranged them into a structure where one algoritmic component depended on others. But to be successful required that every component was powerful and separate, a unit in itself, performing a specialized task. Your thinking was systematic, and it created a system. Only later, emergent behavior of complex systems was discovered[10]. The behavior of algorithmic systems in the 1960s was in principle predictable even if the prediction was within probability limits only.

The machine – since it was a machine – performed according to the algorithm that you had prepared. Your preparation was part of an ultimate form of thinking: You had to think in the way the machine would be thinking if it could. Even though the program, as a decent program and worthwile to be designed, contained piles of random decisions according to various probability distributions, the artist could name and predict important aspects and structural features of the drawing. He was drawing from the distance and with the brain.

Such was the thinking early computer artists had to invent and to grow into. They lived in their algorithms. And yet, they were often surprised by the final material results created by their programs[11]. Such a surprise was possible to happen, and always created an exciting feeling running down the spine and up into the brain, because of a great discrepancy that comes with the execution of a program. In a way, the programmer is observing two processes. On one hand, he is watching the program's execution. Is it still operating properly? On the other hand, he is observing the much slower process the drawing machine is performing[12]. His judgement is a mild form of interest only in the case of the calculating part of the system. The drawing operation, quite differenty, generates more excitement. An astonishing visual process, slow but steady, can be perceived. Sensation! The difference between the material and perceivable object in contrast to the mental and imaginable object is the particular incentive of algorithmic art.

The algorithmic artist, before anything is appearing visually as ink on paper, holds in her hands (or keeps in her mind) the entire abstract and unlimited class of all those patterns, drawings, and paintings that the current individual piece belongs to in all its limitations. Creation in the case of algorithmic art is creation of infinities, of classes, of sets. The work, in material terms, is a description. But this description is executable. It is a sign and a machine at the same time. In so far, it is a monstrosity. It is a concept, the conceptual artist would say, shrugging his shoulder. The algorithmic artist would light-heartedly and happily agree. But he would continue: my work, so ungraspable and intangible as it is, is an *executable* concept. That's very different from yours.

The material artist, in order to produce something that passes as art, must break the resistance of a piece of matter. The material he or she uses resists against the artist's will and idea. She wants to force

---

[8] The original publication appeared, in German, in the legendary series rot as no. 19: „computer grafik".

[9] Georg Nees: Generative Computergraphik. Kaleidoskopien Band 6. Berlin 2006

[10] It makes sense to say: emergent behavior emerged.

[11] „Created"? Well, yes and no.

[12] In actual fact, in the mid-1960s this was done off-line.

the material into a form according to her idea and pre-conception. The material does not want to be forced into that idea's form, or into the form the idea needs to materialize. In the case of the algorithmic artist, it is the algorithm that usurps the role of the material: its resistance. Before the algorithm appears as a finite description of an infinite set, the artist must break all of the algorithm's resistance. The algorithm just does not want to be forced to react to signals in exactly one and only one way only to please the artist and the audience.

Nowadays, the situation appears in many aspects to be simpler. There now is a rhizome of algorithmic components. They are available to everyone for almost random use. If, what the digital artist in present times is trying out, fits, it is fine. If it doesn't fit, it is of no great harm. Some other path will do.

The fact that the building blocks of the algorithmic rhizome are still of an algorithmic nature (if you like), often goes unnoticed. If the system's behavior deviates from what the artist has expected, there are usually alternative options that are safe to try out. A playful attitude is good enough to find a way, and after some trial and error a work will be the result.

Recently, creativity researchers have discovered „social creativity". Creativity, in accord with general assumptions about the individual in a typical modern society, is usually considered a capability of the individual citizen. He or she can collect considerable advantage from showing creative behavior. A lot of ordinary ideology rests upon the individual-creativity assumption. This is in contrast to the greatest creators' repeated insisting on how small their individual contribution was in comparison to what had before been accumulated by others.

The recent fad, however, is that we are told we must study social aspects of creativity in order to understand the computer's creativity, or to make computers creative, or, at least, to make sense of computational creativity. This is funny.

The human being exists first and for all as a group animal. There is no human survival of an isolated human. The social existence of the human being is the natural part besides pure biological existence as an individual.

In a daringly huge jump we may depict the situation, thousands of years later, thus: humans exist in society. Society develops to the point where it creates the individual whose final break-through comes with the renaissance, enlightenment, and French Revolution.

Society, and social reality and survival also lead to tools, devices, techniques, machines. One of the latest and, indeed, most radical such transformations comes with the computer and information technology. It is most radical, because now it has become possible to externalize mental labor. Mechanization (manual) was turned into semiotization (mental).

Now, the existence and use of computers in turn changes the individual that had before been separated by an act of abstraction from its existence as part of the social fabric. Here we see a dialectics where society and computer technology create and develop the individual while, at the same time, individuals influence societal and technlogical transformation. Between the three poles of social, individual, and computational creativity, I see a constant motion. None of the three can properly be thought without the other two.

Our individualism has conquered much, if not all, of our thinking to such depth and extent that we deny and ignore the dialectic interdependence of societal and individual existence and, in consequence, of social and individual creativity. We rather turn to computational creativity as a partner of individual creativity, thus ignoring a fundamental fact of all machinery. Considered on a very general level, what

machines can do is always abstract repetitive motion[13]. Machine motion is abstracted, by close observation, from human behavior and activity. In steps of often historic dimension, human activity may be decontextualized, concentrated, condensed, and standardized so much, so densely, and so far, that it becomes possible to build a machine that is doing „the same". It is doing „the same" only in terms of resulting functions, never in terms of the performing processes.

The computer itself – the computing machine – is the best example of this. Down to its bones, what it does is to carry out a computable function applied to one of its feasible arguments. Alan M. Turing succeeded to propose a computing machine (a paper machine, interestingly) by functionally describing what each and every human being is doing when he or she is performing a calculation although none of them ever did, nor will ever do, it the way Turing's machine does it.

Think of this! Humanity reached a point in the mid-1930s, where some of human mental activity (called calculation) could be described in such an unambiguous, precise, operational, and effective way that it became trivial to describe a machine that would realize the same function – although the process according to which the machine performs its calculation is totally different from each and every human's calculations.

Only an intricate interplay of social groups and individual members of those groups could lead up to such a climax. Once it was gained, the whole picture got gradually turned around, and what humans had always done by virtue of their being humans now appears as if what they do is only a reflection of the machine's operations. This is what Karl Marx calls the necessarily false consciousness. We are standing on our heads when we observe what we do and, therefore, turn things and realtions around into their opposites.

If we allow ourselves, for a moment, to think as unprejudiced as ever possible, there can be no doubt in us that what a computer does, under the control of a program, is but a sequence of elementary operations. Down in its very last guts, it is performing a first simple operation, then a second, and a next one, and so on. It was built as a machine and the engineers have built into it a finite set of elementary operations, and nothing else. Each one of those operations is clearly defined. We, of course, cannot but interpret these operations on higher levels of accumulation, association, superization, etc. What is remarkable here is our great (and creative) performance in interpretation; not remarkable at all is the computer's effectiveness in operation.

Without any doubt, computers operate, and they only and always operate. They never do anything else. Without any doubt, humans create, but they only occasionally are what we call creative, and often do other things.

On the lowest level of creativity, we are all and always creative because we are active. We do something here and now and thus, and this activity results in something, tangible or not, that was not there before. So it was created. If we call this trivial form of human activity, „trivial creativity" (it is the creativity we are forced into by nature), we can distinguish from it a „historic creativity". It leads to results no-one else has done or seen before – a claim that is a bit idealized. But as a thought-figure we may accept it. Such a view generates without great pains a scale from trivial to historic creativity, and all kinds of shades in between.

Such a scale allows us, if we want, to identify areas where it is likely or even desirable to construct machines that can do „the same". Computational creativity then loses its myth. This should be good for

---

[13] Reinhard Budde and Heinz Züllighoven, in their extensive recollection of Heidegger's *Time and Being*, characterize the machine as abstract repetitive motion and as materialization of formal aspects of social activity. „Maschine ist in der Konstruktion eingefangene, aus dem Zusammenhang gelöste und wiederholbare Bewegung." (p. 137) „Maschine verstehen wir als Vergegenständlichung der formalen Aspekte sozialen Handelns." (p. 145) Their great thesis (Software-Werkzeuge in einer Programmierwerkstatt, München: Oldenbourg 1990) is a treasure hardly known. It contains a subchapter on machine and creativity.

human creativity. Human creativity is not an abstract ability existing unchanged throughout time. Human creativity is always concrete. It happens here and now, in context and situation. Therefore, its forms are determined historically and socially.

Creativity as such is the general predicament of the human race. But whenever it occurs, it is the actual and current result of the development of society. To call for „more creativity", in particular for more creativity generated by computational means, appears as odd to me. Creativity is a human quality that comes in shades. It is happening always and everywhere: trivial and self-evident. Historic creativity is historic because it happens only occasionally.