Chapter 9

COMPUTATIONAL INFORMATION GEOMETRY

PURSING THE MEANING OF DISTANCES

Frank Nielsen

Abstract

In this chapter, I would like to introduce and explain to the Readers my current motivations and driving forces that spurred my interests for shaping a new area of computer science: Computational information geometry. I am making this task all the most challenging for me by enforcing a minimum use of relevant background materials. Furthermore, I will limit equations to the very strict minimum, if any. This is to contrast with the remarkable masterpiece work of Penrose 'The road to reality: A complete guide to the laws of the universe' (Penrose'2005) that choose to embark on a journey of mathematical conceptualizations to explain our current

understanding of "reality". First, I would like to share with the Readers and organize some thoughts on the methodology of science itself, its targeted goals and its evolution along the centuries. This is all the more important in the 21st century as I will show that there is no ultimate science and therefore it is of the utmost importance for us, scientific researchers, to timely frame our activities while leaving us, yet, opportunities for serenpidity, fortunate unexpected discoveries that lead to quantum leap progresses. My claim is that *abstraction* is the basic mechanism mastered by human intelligence and the driving force for *open-ended science*. After these brief considerations on science, I will then introduce *computing* not as a mere tool as it is unfortunately often described, but rather as a *fundamental principled law* of nature related to time. I will discuss on the relationships of science and technology as well as how the controlled societal actions on our world yield our dynamic perception of ephemeral reality. Once these important preliminaries are discussed on, I will then consider both abstraction and computing under the auspices of geometry by briefly reviewing the historical developments of geometry itself, and how its various conceptualizations have yielded profound revolutions in our way to perceive and grab reality. I will finally introduce into that context my novel research field, computational information geometry, and discuss about its prospects for future stance of computing and technological innovations. To illustrate this latter point, I will describe the example of our on-going project of *distance personalization* in information retrieval systems for next generation Internet search engines.

1 What are the methods and purpose of science?

Let us first look at the modern definition of the word "Science" as found in Wikipedia: "Science is a body of empirical, theoretical, and practical knowledge about the natural world, produced by a global community of researchers making use of scientific methods, which emphasize the observation, experimentation and explanation of real world phenomena".
Although this definition gives a good flavor of what is science, its role and methods are rather not mentioned. Personally speaking, the primary purpose of science is to gain better understanding of our surrounding world that in turn also yields better control of it

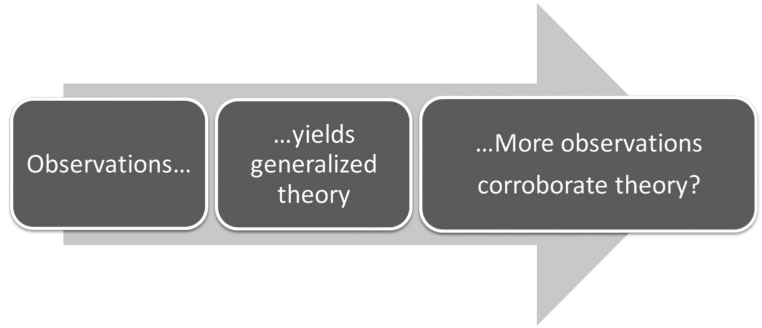

Figure 1: *The old inductivism scheme for science is passive and became obsolete in nowadays' 21st century.*

by various technological instruments.

For a very long time, the major approach of science was considered to be inductivism. Inductivism builds on a set of observations to generalize these into a theory that is then corroborated by more experimental evidences (Fig. 1). The Occam's razor principle comes in handy for find a better explanation: Find the shortest one! This minimal principle is still at the heart of modern computer science as attested by the MDL principle in statistics: Minimum Description Length (Rissanen, 1978).

Although humanity has sticked to this model for a very long time, this is not anymore the scheme of modern science. The essential reason being that they are potentially many theories that will explain the same set of observations and that these observations are not necessarily independent of each other. Thus inductivism is rather a *passive scientific method* that awaits eventually for an observation that will not fit in the theory to invalidate it. Yet inductivism is a fundamental concept for mathematical theorem proving, and is often used in IQ tests where one is asked to find the next likely pattern given the first pattern elements. These inherent limitations and criticisms of the inductivism method were first strongly repudiated by Sir Karl R. Popper in his 1934's masterpiece essay on the philosophy of science: Logik der Forschung (Popper 1994, translated by Popper himself in english a few years later as The Logic of Scientific Discovery). Popper advocated the empirical falsification point of view instead of the classical observationalism and justificationism and introduced the trend of *critical rationalism*.

To my opinion, the current method of science that prevails nowadays is rather based on the philosophy of *abstraction* and *active* boundary checks of the theory. The main difference with inductivism is that we seek for criticisms or limitations of the theory at the same time we build it so that a theory is inherently a wrong yet useful model. This reminds me the citation of industrial statistician George P. E. Box, the son-in-law of reknown statistician Sir Ronald Fisher that once said: "All models are wrong, but some are useful." So let us quickly review the spiral pipeline of abstractions and criticisms. We start the problem-solving task by first conjecturing a solution/theory that is then actively criticized to find its limitations (id est., when the theory falls short). Compared to the inductivism, this methodology is a proactive stance of doing science as we all agree that a theory is just but not more than a convenient generalization of observations that is nevertheless inherently limited. Once a bad example for a theory is discovered, we then try to generalize the former theory by abstracting it so that both former and the new examples fit into the novel abstract concept, and so on. This yields the *open-ended spiral of science* that highlights the fact that there is no ultimate scientific study. This may also explain why after a few decades of versatile research in efficient arithmetic operations, it is still a hot topic to find a better algorithm for multiplying matrices (Robinson 2005). Popper strongly argued that scientific theories are abstract by essence and can only be checked indirectly so that theories are born from creative imagination and only conjectural. The good benefit of abstraction is that it can yield also to new problems that would not have been studied otherwise (Fig. 2). A contemporary

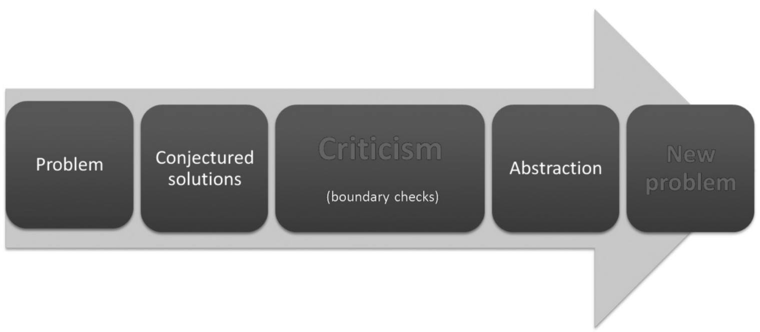

Figure 2: *Modern science relies on abstraction to proactively seeks for the inherent limitations of models. The method of criticism and abstraction further yields novel unexpected problem tasks.*

example of this cycle of abstraction/cristicism/novel problem method of science is the studied of Hermitian matrices with complex coefficients in mathematics that proved to be a key tool for analyzing quantum systems in Hilbertian spaces (Rae, 2006).

What is the rate of progress in Science? It is tempting to believe that science progresses linearly by stacking new knowledge. However this is clearly wrong as empirical evidences attest it. Indeed, we are all familiar with scientific revolutions called paradigm shifts that unavoidably happen in the course of scientific enquiry. Although Thomas S. Kuhn did not coin the term paradigm shift himself, he elaborated the theory that science periodically grows and collapses into the following three stages: First, there is the prescience era that lacks a central paradigm. It is followed by the usual "science" long era where one tackles and solves puzzle questions in order to enlarge the central paradigm. This is different from Popper's approach in a sense that the failure of a result to adhere the central paradigm is perceived as a mistake of the scientists and not as the refutability of the criterion. As these anomalous results build up, the status of "science" reaches a non-turning point stage where scientists acknowledge a serious crisis and flaws in the current paradigm. This crisis yields eventually to the new paradigm which subsumes the former results and allows one to explain the so-far anomalous results. This process explained in Kuhn 1962's book "The Structure of Scientific Revolutions" is termed revolutionary science.

Although it is tempting to elaborate more the science of science, known as the epistemology, we refer the interested Readers to the book entitled "The fabric of reality" (Deutsch, 1997) that devotes a full chapter on this topic. I actually agree with Deutsch in the sense that there is a need for describing a theory of evolution of science, the same way (but on very different time scales) as Darwin did in his "Origin of species" treaty in the 19th century (Darwin, 1859).

Since we have seen that there is no ultimate science, one may ponder which problems or science we should be aiming at. On one hand, there is an urgent need to solve the very well media-covered topics such as sustainable ecosystems and pollution by providing new eco-friendly technologies. On the other, there is a growing interest to explore the spiral of thoughts of computing and deeper conceptualizations that may bring a full spectrum of innovations. Steels, in this book, proposes to use the current Internet infrastructure and sensor network devices to build community memories (Steels, 2008) for taking immediate eco-friendly actions.

In contrast, my approach and devotion is to look at the combination of computing and geometric representations to provide an unprecedented modular algorithmic paradigm for better modeling and solving real-life problems. The next section will describe computing not as a mere machinery tool but rather as a principled law of nature that is nowadays carried on silicon transistor chips, and discuss about the science/technology cycle as I have experienced it the past ten years. It will then be followed by a section describing my current research focus and its potential impact: Computational information geometry.

2 Computing and computers

Computers are ubiquitous in our daily lives of the early 21st Century. Computers are simply everywhere, omnipresent, fairly visible and cumbersome to manipulate today, hidden and intuitive tomorrow. In fact, for most of us, we already live in full symbiosis with them without having real consciousness of this fact; Let me take a simple example to illustrate this point. When we go to work, commuting by train, we simply do not have conscious of all the computing infrastructures that let this make possible. For example, do we think of real-time schedule adjustment? controlled security measures? speed optimization? and so forth. Actually, this is only whenever a traffic accident occurs or an unexpected labor strike arises (I'm French!), that we suddenly realize and get full conscious that everything is swiftly and smoothly adjusted real-time by computers. Not only does our train need to be locally rescheduled, but also all other trains in circulation potentially need global adjustments. Modern railway companies benefit and operate with such a computing train system that is fortunately invisible to passengers. I may say that living in Japan for the past ten years, I myself have personally attested and been surprised quite a few times by this aspect of pervasive computing reality. In a word, we are already living in full symbiosis with computers. We have certainly passed the non-return point as we cannot anymore live without these computing devices. But first, let me remove a misleading expression: "computer science."

Nowadays, it becomes quite popular, if not trendy, to complain about the booting time of personal computers (PCs), or to just keep our hands still or off the mouse whenever the computer "tells" us it is busy (meaning computing) using the now de-facto sand-filled hour glass icon. Thus, it is also not that surprising for people to equate computing with computer, and quite unfortunately computer scientists with computer troubleshooting experts. Indeed, how many times have we not been requested by our family relatives or friends in helping setting software, cabling correctly peripherals, updating device drivers, etc. because we are after all computer scientists, are not we? People just do not realize that computers are the *physical hardware architecture* supporting efficiently a truly beautiful and puzzling principle of Reality that we pursue: Computing. Maybe we should propose to rename our job of computer scientists as *computing scientists*, a more appropriate term that will further raise the awareness of our profession, a job title likely less confusing for all of us. Computer Science should then be called the Science of Computing. ACM, the Association for Computing Machinery (born from Berkeley in 1947), actually retrospectively carefully chose to use the term computing instead of computer for avoiding this confusion. It did a good job at combining both concepts of computing and computers as *computing machineries*.

To convince one that computing is, at least in principle, different from computers, let us quickly review a few historical milestones of the early days of Computing. Establishing the discovery of computing is in itself challenging as Chinese already used abacus circa 5000 years ago to perform elementary arithmetic operations. Although the additions and multiplications carried on the abacus are undeniably primitive forms of computing, the *real power* and *versatility* of computing (branching algorithms) was presumably first stressed and harnessed by Muslim thinker Musa al-Khwarizmi (780-840 AD) in the 9th Century. Musa al-Khwarizmi, an influential scholar working at the then house of wisdom research and teaching center in Bagdad, wrote a treatise entitled *Hisab al-jabr w'al-muqabala* that yields the branch of algebra, stemming from arabic word al-ajbr. The book written around 830 AD describes in plain sentences from given carefully chosen examples detailed steps for solving linear and quadratic equations. What is retrospectively remarkable in this masterpiece (besides that fact that al-Khwarizmi already knew that quadratic equations may admit two real solutions!) is that al-Khwarizmi was interested in *teaching* the reader *general methods* for reaching the solution from six basic template cases (IF…THEN), and not solving the specific instances at hand, that were rather given for illustrating his solutions: the computational workflows on prescribed instances. Readers could then "compute" manually from new quadratic equation instances the solutions following the steps described in the workflow recipe. These recipes are today commonly named *algorithms*, a word that stems from the latinization of his name (algoritmi) in remembrance of his pioneer work.

Although the very basic concepts of computing and (branching) algorithms were born in the 9th Century, this is only a thousand years later that the very first computer, a mechanical device (a machinery), was dreamed of. This computer machine was born in the imagination of an English mathematician: Charles Babbage (1791-1971). Babbage's celebrated *differential engine* is an analog computer designed to tabulate polynomial functions, and was targeted at replacing "human computers" from such a daunting task. The dedicated machine was invented and described on drawings. It was only after the death of Charles Babbage that his son built physically partial modules. Today, there is a fully working "replica" built from the seminal plans on display at the London Science Museum, and even a few amateur Meccano™ prototypes built by Lego® aficionados worldwide. This emphasizes that computing is a fundamental principle of reality and computers are merely efficient automated physical hardware for performing computations. The 20th Century was marked by the steep acceleration of the computing paradigm and its establishment as a science field in its own. Computing efficiency, the number of time increments required to perform a task, was further improved by shifting from the sequential to the parallel computing paradigm. The very first vector computers followed in tandem in the early 1960's, and yielded the bright era of high performance computing (HPC).

A masterpiece work that kicked off computing science was revealed to us by Alan Turing (1912-1954) who presented and proved in 1936 a universal model of computation: The Turing machine. The Turing machine describes the basic model of computations and emphasize on the duality of data/program. Any computer can in principle be programmed (id est, emulated) on the universal logical Turing machine. Further, Turing showed some limitations of computing. Namely, some problems are *not decidable*, meaning that these problems cannot be solved by algorithms. Classical examples include deciding whether a program will end or loop forever, whether an equation admits a solution or not,

or simply whether a program is buggy or not…

The discovery of Deoxyribonucleic acid (DNA) by James Watson (1928-) and Francis Crick (1916-2004) in 1953 and its role in the biological mechanisms of replication in cells for transmitting the genetic information to siblings had many profound implications beyond the Nobel prize attributed to their discoverers in 1962. One of them, being that it reinforced human thinking of computing and algorithms as the common language of Nature, Man and Machine, three aspects of reality. Our understanding of the perceived reality has made a major quantum leap.

The beginning of the 21st Century further attested the massive adoption of the distributed computing paradigm, the birth of Internet and peer-to-peer networks, and the consumer adoption of HPC teraflops game consoles

3
What is the limit and consequence of Computing?

With the fast technology pace of innovations in computer machineries, it is easy to get engrossed into the techno-mania wave and forget about the outcomes of all the technology extravaganza. But let us pause a minute and ask us what is the ultimate or at least long term impact of Computing on us? A drastic point of view is to consider computing and computers are novel reality affordance of humans. Let me explain this point in greater details. Computing (and computers) is similar to what was the telescope invention (allegedly evangelized by Hans Lippershey) was in the 16th Century. Telescopes indeed eventually yielded a milestone breakthrough in human society perception of Reality by Galileo: The Earth is not flat! Before the telescope invention several theories were confronted that either advocated a flat or curved spherical Earth. The great Erathostene even computed the Earth spherical radius in 250 BC. But telescopes provided undeniable evidences for the mass, even to the skepticists. Telescope was a novel reality affordance of its time that let us unravel some bits of our surrounding Reality, a non coming back turning point in the history of *human thinking*. What I mean is that, we homo sapiens sapiens, follow our developmental trajectory since our apparition on Earth, approximately 2.5 million years ago. Sometimes, the discovery of a new "toy" (affordance) let us make a big jump in our understanding of nature by accepting

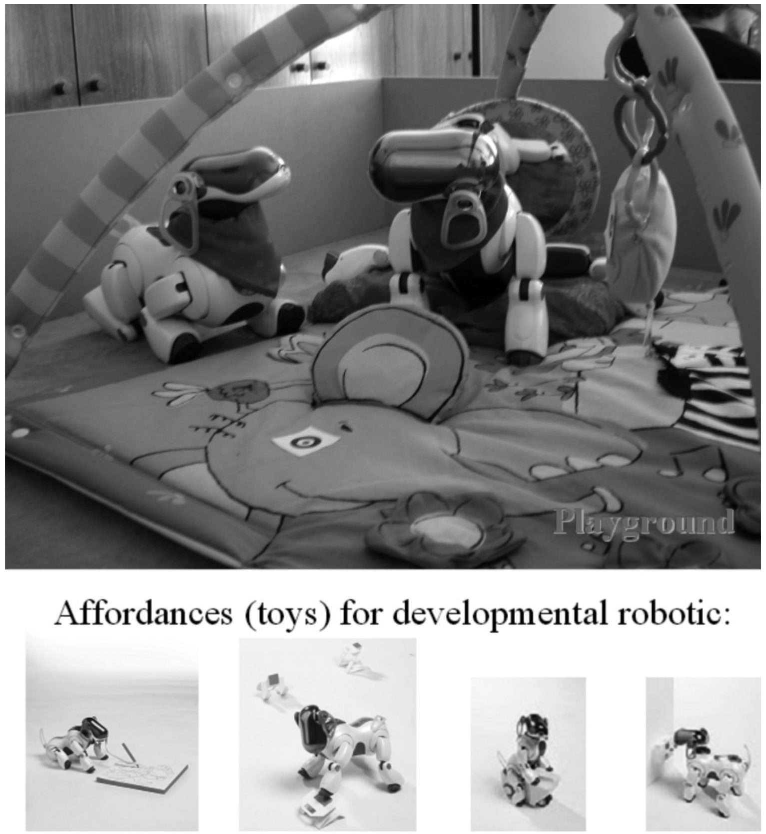

Figure 3: *Providing robots with new toys give them novel affordances for exploring and interacting with the surrounding world.*

hypothesized theories, and these affordances can even spur new species. Even (computational) experiences in robotics have shown with great success that by providing new toys to robots (like a mirror, bicycle, pen, etc.), robots which were programmed with an intrinsic motivation could enhance the hierarchical representation of their perceived world thanks to these toys (see Fig. 3, adapted from Kaplan & Oudeyer 2006).

Will computing and computers lead to homo cyberniticus? In fact, computing had already allowed us to break barriers and explore new spatial frontiers: Space. The first manned space mission of Yuri Gagarin in 1961 could not have been done without computers (nor mastery of computing). Nor the moon mission in 1969. This territory spread of human race is often attested by anthropologists for anterior human species. It is worth mentioning that since November 2000, the International Space Station in space) has always been inhabited. How many of us realize it yet? Maybe a hosted webcam and a daily space weather forecast will accelerate the social collective consciousness. How far are we from this specie transition? Well, citing the double exponential law of returns of Ray Kurzweil, explained in great details in his book "The Singularity is near" (Kurzweil 2005), not that far. Ray Kurzweil offers us in his book a detailed panorama of technology pace figures, and localize this "jump" at what he called the singularity point, sometime around 2040, where machine intelligence takes on human biological intelligence. I know that this date may seem to some of the readers highly unrealistic. This is indeed in a mere 35 years. However, even today (2008) computer algorithms beat human not only on chess, as we are already aware of, but also on face recognition contests. So let us extrapolate doubly exponentially to 2040, and that is well… quite unpredictable, is not it?

4 Todays' interactions of computing with reality

Computers are another glass of reality: Namely, the *computational lens* at our disposal.

Rekimoto will further report in another chapter of this book on his own viewpoint of the digitality of our world. Computers offer brand new perspectives of our perception of reality, and changed drastically in a record time the ways of doing Science worldwide with the paradigm of Computational Science. This is yet another testimony of pervasive computing. For example, in the image processing domain, computers allowed us to stitch massively thousands of pictures into a seamless composite mosaicking picture that has much larger resolution than our naked human eye can discern (Nielsen, Yamashita, 2006). When such giga pixel telescope pictures' become widely available to the consumer market, we can ask ourselves whether our perception of reality will change or not? Are we going to get extra-lucidity (clairvoyance) or extra-consciousness from these pictures? That is, these extra resolution pictures will be processed to seek for pertaining information to show us. We will soon need intelligent contact lens.

The computer will provide partial reality windows of the reality. To give an analogy, this kind of high resolution imagery will bring

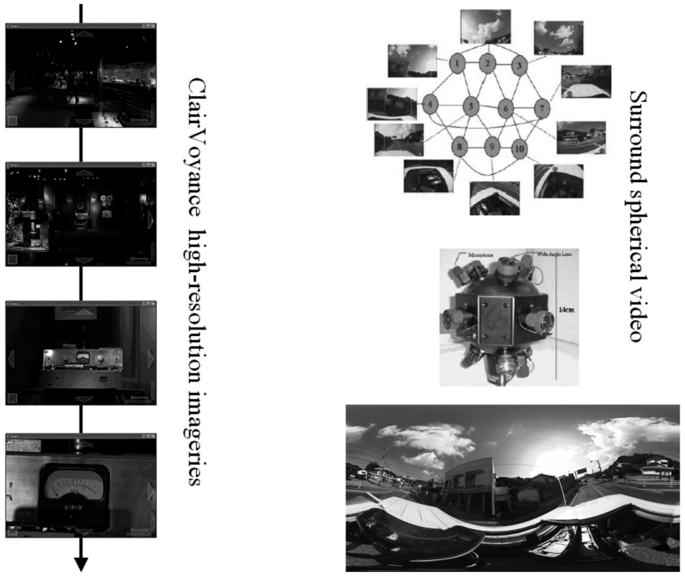

Figure 4: *Overview of the clairvoyance gigapixel imagery system (left), and the FourthView full spherical immersive video system (right).*

us what telescope images brought to Astronomers: New discoveries yielding better understanding of reality. It is actually not that surprising to see that such an *abundance of sensorial information* has already leads us to surprising discoveries by communities of on-lookers. Take for example, the satellite imageries and its navigation tool offered by popular service Google Earth. Some novice users allegedly discovered new meteorite craters just by carefully and methodically inspecting such a huge database. Imagine the possibilities when image understanding beats human rate… This kind of massive stitching project is just the top of the iceberg of *computational photography* that is literally changing the landscape of photography by redefining the notion of photos. To further illustrate computational photography, let us take another example: Surround video (Nielsen, 2005a). By assembling compactly a cluster of cameras around their joint nodal points, we can capture a full spherical image of the abstract camera without having any visibility dead zone. We get what could be called a cyclop omniview eye (Fig. 4). What is particularly nice from a standpoint of computer vision, is that the image delivered by this cluster-camera is free of intrinsic camera parameters and offer both novel theoretical and experimental possibilities in the range of computer vision applications.

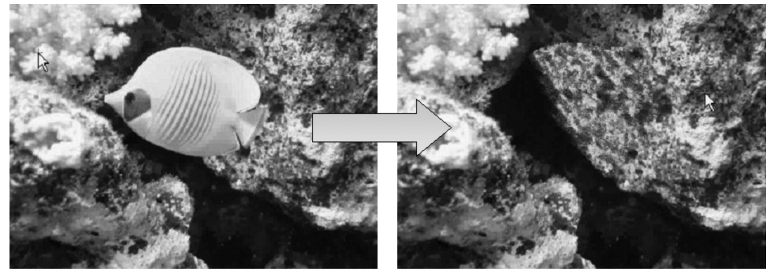

Figure 5: *The ClickRemoval system (Nielsen and Nock. 2005) allows novice users to easily erase undesirable object and fill the background of digital image by a simple mouse click. This raises the questions of spammed reality and authentication of digital media (DRM).*

But at the same time imageries shift from purely sensing to become more computational, we face more and more the risk of digital forgery, and potentially form of *spammed reality*. We are all too familiar with the spectacular visual effects in the Hollywood-driven movie industry. But sometimes, we simply forget about it and take it for granted, as some kind of reality. Pictures are easily retoucheable (Nielsen and Nock, 2005), and nowadays it becomes more and more difficult to authenticate them (see Fig. 5). Securing this digital reality is going to become a major challenge in the forthcoming years, and will likely be solved by cryptography-everywhere digital world.

In 2007, virtual reality (VR) has also entered massively our lives, not only our spirits, at least for gamers. And this tendency will be even further accelerated in the fall of 2007, with massive VR 3D lounges in market game consoles (Playstation® Home). Whether we like it or not, social changes and net criminality of avatars will surely happen. Should criminals be in jails or just their avatars?

I would like point out, as Descartes already previously did, that virtual reality is not that virtual but all the more real. VR is indeed computed by a physical process, a real machine that obeys the *unbreakable law of physics* (the key structural postulate of reality?). There is no zero-gravity flight simulator on Earth ground. We really need to take an airplane and follow its parabolic flight trajectory to experience it. No way of bypassing it. Recently, Stephen Hawking experienced himself zero-gravity in such a flight. This simply could not have been physically done using VR on the ground. Vir-

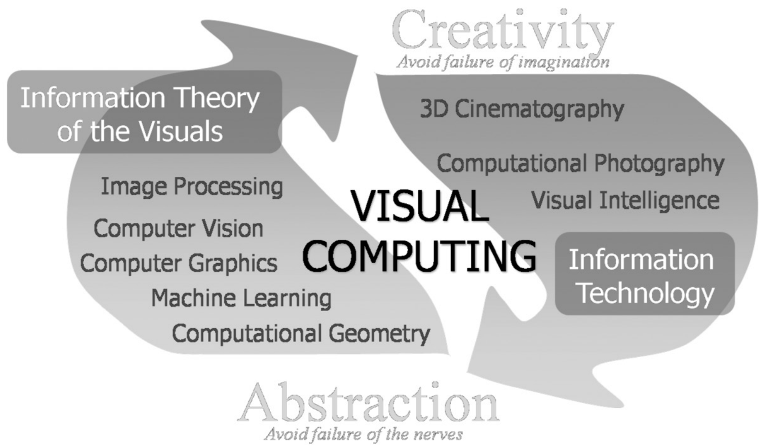

Figure 6: *An overview of visual computing as a cross-disciplinary field of theories and applications for the visuals (Nielsen, 2005b).*

tual reality offers novel facets of realities that are yet constrained by the physically-driven reality. David Deutsch in his wonderful book "The fabric of reality" (Deutsch, 1997) further elaborates on the possibilities and limits of VR. Deutsch differentiates two types of user experiences in VR systems: the external experiences like looking through the VR cockpit as if you were inside the real cockpit, and the internal experiences that are linked to emotional states, and difficult to read and render today, but potentially possible to control to some extent in tomorrow virtual realities.

I have been working the last ten years or so at the junction between theory and applications of computing for the visuals in the multi-disciplinary areas of computer vision, computer graphics and computational geometry. Visual computing (Nielsen, 2005b) allowed me to establish a balance between theories and applications (see Fig. 6).

Furthermore, this research field allowed me to frame my research by considering simultaneously the advances of these different communities that have traditionally evolved independently, for historical reasons. I would like to cite verbatim Professor Guibas who honored me in writing the preface: "..I am really excited to be able to write a foreword to Frank Nielsen's new book *Visual Computing: Geometry, Graphics, and Vision*. The fusion of computer graphics, computer vision, computational geometry, and discrete algorithms that this book presents is truly unique and, in afterthought, so obvious. Geometry, graphics, and vision all deal in some form with the shape of objects, their motions, as well as the transport of light and its interaction with objects --- yet historically they have been covered by separate courses in curricula, grown around a distinct set of conferences, and cultivated separate communities. This book clearly shows how much they have in common and the kinds of synergies that occur when a common core of material is presented in way that both serves and is enriched by all three disciplines."

5 Computing is not Understanding

Since the early days of computer vision, understanding images have always been an endeavour goal. However, even nowadays performance are still very far from the performance of humans, although I have mentioned above that computers recently have beaten human performance on face recognition contests. We have attested drastic performance improvements over the past five years in computational learning and visual computing supported by Moore's law two-fold computational increase every 18 months. Scaleable recognition methods are on the way, and face recognition systems are already available in embedded in consumer digital cameras. Yet, everybody agrees to say that computer abilities (and the computing paradigm by the way) are inherently (?) limited. So far, limitations in computer science have been addressed under the auspices of tractability, and unsolved challenging pending issues such as the million dollar P=NP question (Clay Millenium problem)? Notice that although some problems can be tagged as NP-hard, this does not mean that there is no good heuristic in practice to tackle large data sets. One such example is the so-called 3-SAT that asks for checking whether a Boolean formula that consists of conjunctions of three laterals admits a true solution or not. 3-SAT is at the heart of designing and checking logic circuits in processors, and are daily use in the competitive industry of processor manufacturers. Yet, artificial intelligence (AI) and its comparison with human abilities raise another level of limitation: Is all numbers as the Pythagoras school believed, or is there something `undigiteable' that makes us really human, not like a machine. Finding any clue will allow us to potentially explain our qualia experiences? Take for example, the image segmentation problem that

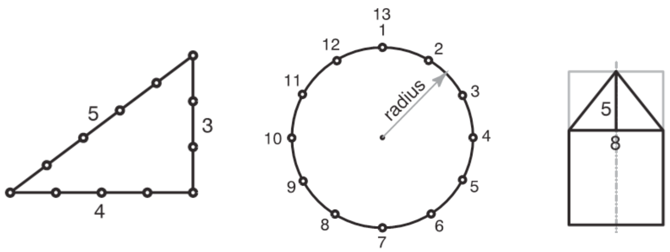

Figure 7: *The surveying rope of skilled harpedonaptae consisted of 13 regularly spaced knots that allowed them to build not only remarkable geometric figures but also magnificent constructions such as the Sphinx.*

consists in grouping contiguous pixels (a task called clustering) into homogeneous groups (called segments hence its name) in a given image (Nock and Nielsen, 2004). Modeling mathematically (transforming it into numbers) the problem is difficult since so far it has rather constrained the problem "segmentation" *per se*. There are many perceptual phenomena such as the perception of shapes (exempla gratia, Kanisza triangle), or the perception of colors from pure black and white high frame rate movies, lightness, etc. Moreover, even for complex plausible models, most image segmentation algorithms are currently too time-consuming. But keep in mind that this is with today's computing efficiency. The main difference with the human visual pathway in the brain is that the segmentation task occurs from a fully dynamical system approach combining both the bottom-top approach (the aggregation phenomena) and the top-bottom approach (*id est*, recognizing portions of the image allows one to better guess the object shapes and refine the low-level segmentation by bootstrapping). Thus the current bottleneck of image segmentation is not in itself the tractability of the modeled problem at hand, but rather its *modeling*. To give an analogy, Google search engine was a breakthrough because it proposed a new web page ranking paradigm named PageRank rather than solving more efficiently an already well-posed problem. Recently, Bernard Chazelle of Princeton University (USA) wrote an already popular science column in Nature (Chazelle, 2006), describing the impact of a recent computer science achievement: the probabilistic checkable proof (PCP) framework that shows with high probability how to check any proof (once written in an appropriate format) by checking a mere few bits. This means that if the proof of Andrew Wiles on Fermat's last theorem were given in the *appropriate format*, the computer could theoretically check it probabilistically with 99% confidence by checking only 3 bits! This does not mean that computers understand the proof. In fact, not at all, since computer programs coding algorithms all but *crunch binary strings* according to the instruction set. Recall that the control part of computers is trivial (a ribbon Turing machine), making them universal. But because input need to be preliminary encoded into binary strings too, input can be re-interpreted in many ways (called dual ways) so that the computation result, another binary string, is also interpretable in other dual ways. To summarize informally that computing is not understanding, let us say that computing requires formatted numbers that are encoded in binary strings but the semantic (id est, meaning) of these strings can be reinterpreted in many dual ways. For example, let us compute the cross-product of two 3D homogeneous vectors p and q. This simple computation can be interpreted as the computation of the line passing through the two points encoded by p and q, or as the intersection point of the two lines encoded by the same homogeneous vectors p and q.

Thus to disambiguate on the nature of the cross-product computation result, we need to know *a priori* the meaning of the encodings of p and q.

6 Computing and creativity

With the advent of artificial intelligence techniques, it is still somewhat surprising that research itself has not been more modeled and automated as a process. Epistemology, informally known as the science of science is also benefiting from the computing paradigm. To cite a few: Online encyclopedia (such as the famous and de facto Wikipedia), computational science on grids, etc. There are yet potentially bigger and unsuspected innovations expected in this area. Can creativity be modeled by algorithms? Although it is misleading to claim at this stage that creativity is tractable within the context of computing, we believe that some methodologies are computable. James Webb Young wrote a 48-page concise celebrated book entitled "A Technique for producing ideas" (1939) that basically states that ideas can be created by digesting contents (Young 1992), combining them, forgetting about them until a Ah-ha experience happens, and nuturing the new idea. That is, put on your desk a set of notes and examine them carefully by digesting them and trying to link them by finding connections.

Then do simply something else (I personally recommend sleeping) so that you let your unconscious activities process the newly absorbed materials. Likely, at some point, new ideas will spark as you do this, a so-called Ah-ha phenomenon in cognitive science. You will then have to confront this new idea to reality to decide on whether you continue with it or just forget it. To summarize the methodology with a simple expression, let us say that this is like breaking silos, getting into the cross-disciplinary view of ideas. This kind of transitive closure of ideas explored by our unconsciousness yields many novel ideas: A real niche of massive creativity prone by a famous figure in advertising marketing: James Webb Young. I have tested myself this model on my last 10 years research by developing interests for the field of visual computing that combines computational geometry with computer graphics and computer vision among others. Yet, to my opinion, there is another greater source of creativity that yields especially better understandings: Abstraction. Abstraction may be thought as reductionism in a sense that it "merges" two different ideas into a meta-generic idea explaining both former ones. Yet, the good part of abstraction is that it yields potentially also *other interpretations* of the meta-idea, sometimes with unexpected outcomes. Abstraction occurs naturally in Science after a while as a consolidation stage of the idea explosion process as described by Young. But abstraction provides a deeper understanding of principles, a better understanding a structural reality.

Today, we are facing the same situation al-Khwarizmi had in the 9th Century, as we are missing the tailored machineries (quantum computer, DNA computer, etc.) for unleashing and further fostering the foundations of exotic computing (quantum computing, DNA computing, etc.)

Thus we have seen so far that computing is a principled law of nature and abstraction is a human gift for better understating our reality. I will then proceed by reviewing a brief history of geometry that will explain my current research field: computational information geometry.

7 A brief history of geometry

Geometry is at the heart and the root of science. Let us quickly review its historical developments. Geometry all started with the basic needs

to *measure*. In ancient Egypt, 3500 before Christ, religious ceremonies were held by harpedonaptae, dedicated skilled people that had the knowledge to measure, and to plan for Pharaonic constructions. Harpedonaptae used as a basic tool to measure a mere rope that consists of 13 knots regularly spaced by one cubit, 0.5236 meter. Equipped with this surveying rope, these highly respected masters were not only able to measure parcels of lands but also sketch remarkable constructions such as pyramids and the Sphinx. The rope allowed them to build right-angle triangles, regular 13-gons of radius precisely one meter, and even figures with golden ratio, as show in Figure 7. Yet these early days of sacred geometry were not formalized into a body of science, but rather kept as a set of sacred recipes.

The birth of geometry really started with Euclid's famous celebrated work: The Elements, circa 350 BC (Joyce, 1997). Euclid laid out the basic foundation of geometry by providing a simple axiomatization and the deductionism method for proving facts. The five axioms of Euclid's geometry are stated as follows:

1. For every point P and every point Q not equal to P there exists a unique line that passes through P and Q.
2. For every segment AB and for every segment CD there exists a unique point E such that B is between A and E and segment CD is congruent to segment BE.
3. For every point O and every point A not equal to O there exists a circle with center O and radius OA.
4. All right angles are congruent to each other.
5. For every line l and for every point P that does not lie on l there exists a unique line m through P that is parallel to l.

These axioms reflected the intuition that at the scale of the human being, the Earth is perceived flat and that object dimensions do not change under rigid motion. Furthermore, the 5th axiom allows us to build remarkable geometric figure such as the square. This is precisely by the area of the squares that we confirmed the underlying geometric distance: the Euclidean distance from Pythagoras' theorem (Fig. 8).

The Euclidean distance is invariant by change of coordinate systems in accordance with our daily life experience that moving a solid object does not change its measures.

Looking back at the 5th axiom, the parallel line axiom, scholars felt that this "assumption" was rather complicated compared to the

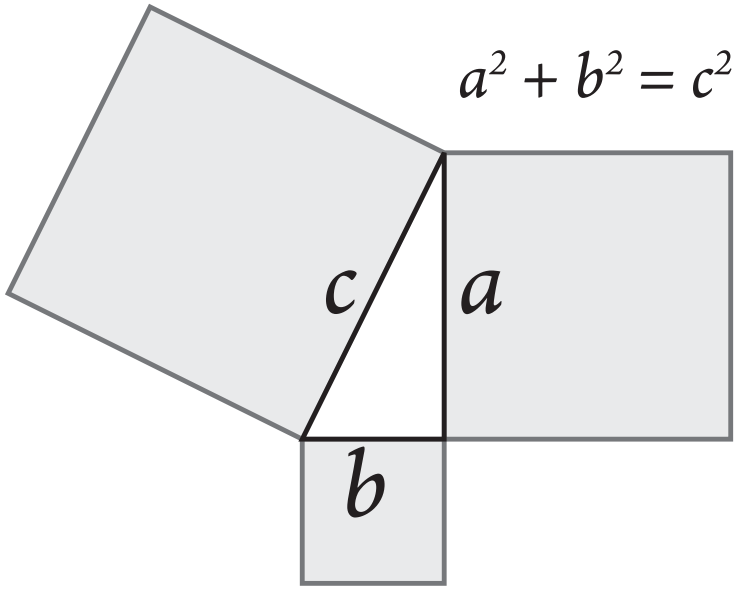

Figure 8: *Pythagoras' theorem yields the traditional Euclidean distance that found its limitations in the 18th century.*

others, so that this 5th postulate should be able to be derived from the former ones. Eventually, failing to do so yielded to one of the *biggest mathematical discovery*: the birth of non-Euclidean geometry (Greenberg, 1993). This discovery happened and was scientifically recognized as "valid" in the 17th/18th centuries with the elliptical (spherical) and hyperbolic geometry. In spherical geometry, there are no parallel lines as lines always intersect in antipodal points. Spherical geometry came in handy for sailing as it was then massively accepted that the Earth is round and that the shortest path between two destinations is not the straight-line segment, but rather a portion of a great arc. In elliptical geometry, there are surprisingly infinitely many parallel lines that pass through one given point, and more strikingly an hyperbolic square is incident to five squares in a tiling. Although these spherical/hyperbolic geometries are primitive old geometries compared to the recent developments that we will mention, the fact that the Euclidean geometry is not the *only one* geometry is yet not so publicly known as a plain and obvious fact to all of us. Yet, people know by looking at world maps that plane trajectories are not straight lines… How do we define the notion of distance in these imaginary geometries? In fact, we should think of distances between any given two points as

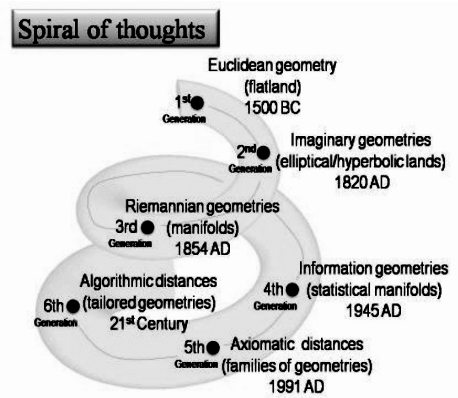

Figure 9: *The spiral of thoughts of geometry from the early days of Euclid to the 21st century. These various abstractions and conceptualizations yielded novel developments of technologies such as quantum systems.*

length of straight lines, lengths of geodesics. To help understand properties of these "imaginary" or "abstract" geometries we consider various embeddings into our real-world, the "small-scale" ambient space geometry: the Euclidean geometry. This is what is mathematically called a conformal mapping in the ambient space. But we should keep in mind that nowadays a perfectly consistent geometry is good if it *cannot be visualized*!

The 19th century was a decisive turning point for the history of science with the discovery and formalization by Riemann that there exist infinitely many consistent geometries that generalized the three former ones (Euclidean, Spherical and Hyperbolic) into the same framework of metric tensors. A metric tensor defines at each point of the space, a so-called Riemannian metric that allows one to define the distance between any two points as the integral of the infinitesimal distance elements (Fig. 9).
To illustrate the open-endedness nature of the spiral of thoughts, let us mention the work of Gil and Michor that studied the Riemannian manifold of all Riemannian metrics (Gil and Michor, 1991).

8 Information geometry

But wait? Did someone say what was contained in the points, or what the points themselves represent? In fact, so far we have rather considered points as "void" quantities, indexed by coordinates. Rao in 1945 first investigated the space of probability distributions as a Riemannian manifold (Rao, 1945). That is, Rao *conceptualized* in this pioneer work that *points represent information* and that there is a natural underlying geometry defined on that space of information. More precisely, in technical term, this Riemannian geometry is completely defined by the so-called Fisher information matrix. Thus we can manipulate probability distributions exactly as geometric entities, and give former statistical estimation algorithms a new flavor with geometric interpretation. Eventually, this approach was further developed, mainly in Japan, by the work of Amari (Amari, 1985) that emphasizes on the dual nature of flat relative entropy manifold for an important family of distributions in statistics: the exponential families. I am not going to give technical details of these exponential families nor on their properties, but I would just like to mention that exponential families in statistics include most famous distributions such as Poisson, Bernoulli, Gaussian, Rayleigh, etc. and that there exists a bijection between these distributions and a particular family of distance measures called Bregman divergences (Banerjee et al., 2005). We can visualize spheres in these abstract Bregman geometries as the equi-distance lines to a given point. These Bregman balls once printed out in our physical world do not look Euclidean-round, although they are perfect balls in their corresponding geometries (Fig. 10).

Figure 10: *Three entropic "balls" printed and visualized in our ambient Euclidean space: Relative entropy (Kullback-Leibler), Itakura-Saito and Logistic divergence balls.*

9 The birth of computational information geometry

In 2004, I first realized and was deeply convinced by the needs of better understanding distances. Distances can be interpreted as a notion of projection and carries the important property of being orthogonal. In computational geometry (Preparata and Shamos, 1985), researchers often used primarily the Euclidean distance, and sometimes the Minkowski norms, but rarely convex distance functions or metric tensors. Moreover, these works on non-Euclidean distances are mostly theoretical and are not really implemented nor tested on practical applications. On the other hand, the theory of information first pioneered by Shannon (Shannon, 1948) considerably evolved into a larger body of activities ranging from channel coding to information representation and compression. Since information theory is originally concerned with communications, many of these theories have been implemented and yielded directly practical impacts.

Just to give a recent example, let us mention the Reed-Salomon codes which are daily use as error correcting algorithms in consumer electronics. These Reed-Salomon codes, thought to be quasi-optimal, have nevertheless been revisited and recently significantly improved (see for an overview: Chazelle, 2007).

So what is computational information geometry?

For short, computational information geometry is the field that emerges from information geometry (Amari and Nagaoka, 2000) and computational geometry (Preparata and Shamos, 1985). Computational information geometry is concerned with the study of the combinatorial properties and design of discrete algorithms by representing information as geometric entities (points, balls, half-planes, etc.)

In computational information geometry, we want to design generic algorithms that are independent as much as possible of the selected distance function. For example, in regression line fitting, or clustering, we would like to design algorithms that can apply to a broad class of distortion measures. For example, the seminal clustering work of Lloyd (Lloyd, 1957) investigated a simple iterative method that enjoys convergence property. Lloyd's algorithm goes by the name of k-means as was originally developed for the vector quantization problem (efficient codebook compression). The k-

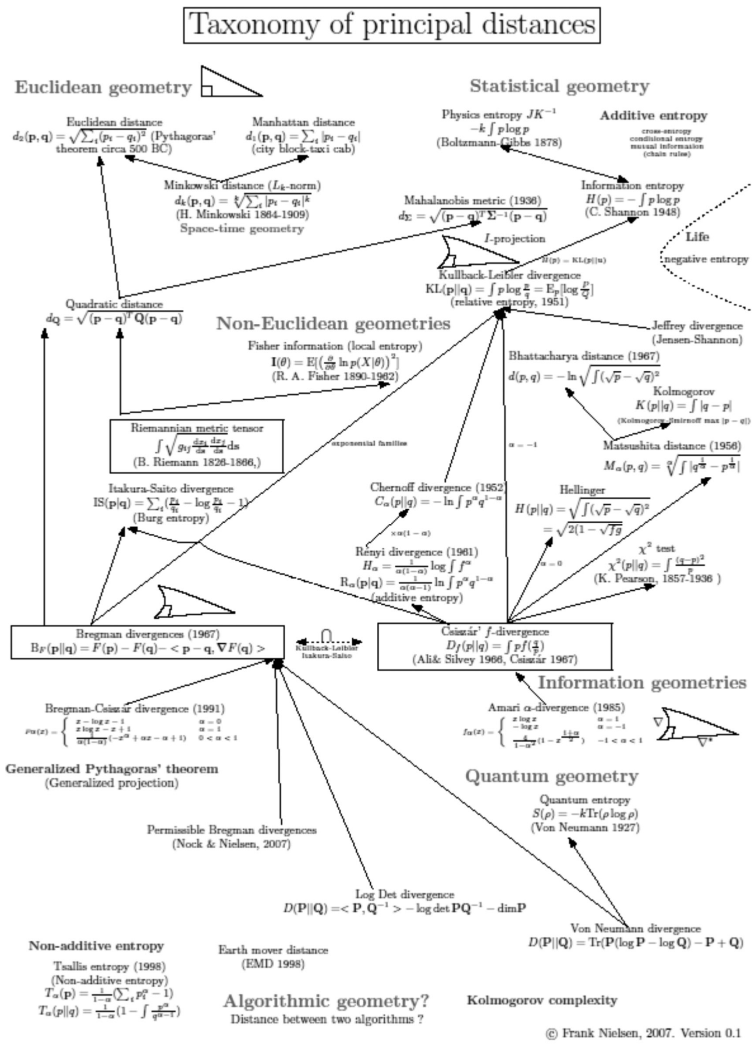

Figure 11: *A historical panorama of distances. Nowadays, these distances are studied in the parametric axiomatic way yielding important families of distances such as Bregman, Csiszar and Burbea-Rao (extending Jensen-Shannon) divergences.*

means method is nowadays widely used in all areas of computational science. Surprisingly, half a century later, the k-means method was revisited and shown to apply only for Bregman divergences (Banerjee et al., 2005). Bregman divergences are an important parameterized family of distortion measures. Every Bregman divergence yields a dually flat geometry that was formerly studied by Amari (Amari

and Nagaoka, 2000). The remarkable property is that the Euclidean Pythagoras' theorem extend similarly to these geometries.

Let me give another simple recent example to illustrate this encapsulation of distances. Suppose you are given a set of "points" meaning, information points) that you would like to simplify or reduce to a best single representative. The way this 1-clustering task is performed in the Euclidean setting, is to either consider the point that minimizes the maximum distance: the circumcenter (MaxMin point), or by choosing the point that minimizes the average distance of the squared Euclidean distance: the center of mass also known as the centroid (MinAvg point). The notion of centroid generalizes also to barycenters by weighting input points, and thus come in handy for performing interpolation. Now, suppose you are given a set of normal (Gaussian) distributions, say on the plane, and you would like to find its centroid. What does this mean and how can we find such a Gaussian centroid? A computational information geometry approach starts by modeling each bi-variate normal distribution as a point in 5D (2D for the mean vector and 3D for the variance-covariance symmetric matrix). Then we define the distance between any two normal distributions as the symmetrical relative entropy (Cover and Thomas, 2006). Once the distance is properly defined on the 5D parameter space, we define the centroid similarly as in the Euclidean case, by considering the minimum average distance (Fig. 11). Recently, we have shown how to compute the symmetrized centroid efficiently but approximately these entropic centers by relying on an exact geometric characterization (Nielsen and Nock, 2007).

In computational geometry, the Voronoi diagram is the most fundamental data-structure on which relies many algorithms. The Voronoi diagram was allegedly first discovered by French philosopher Descartes that studied the influence of galaxies in the universe. The Voronoi diagram of a set of points partitions the full space into Voronoi cells that encode proximity information: the points contained in a given cell are closer to the seed of that cell than to any other cell seeds. The dual structure of Voronoi diagrams are called the Delaunay triangulations, and are widely used in reverse engineering software packages for reconstructing the mesh structures from unorganized point clouds. Similarly, under the auspices of computational information geometry, we have recently revisited these fundamental data-structures and showed how they ex-

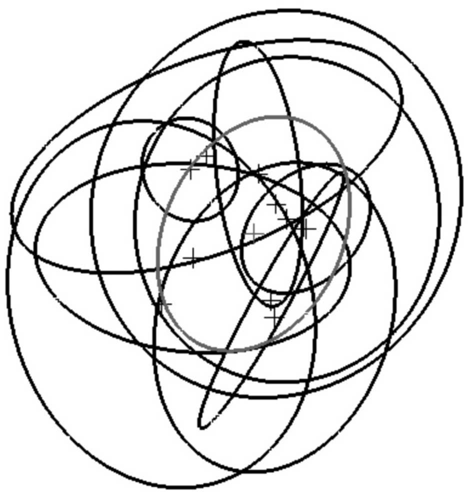

Figure 12: Entropic centroid of a set of ten bivariate normal distributions with respect to the relative entropy distortion measure.

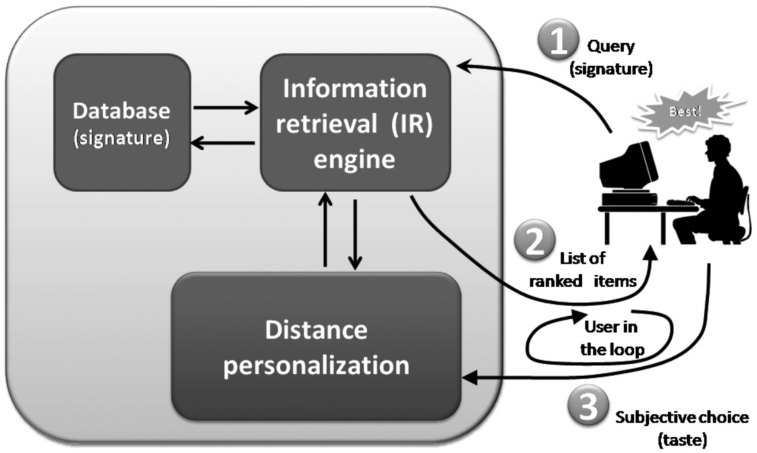

Figure 13: In modern information retrieval (IR) systems, users are able to dynamically steer the distance learning process of search algorithms to reflect accurately and efficiently their subjective tastes: First, a user query the IR engine (1), and a list of top ranked items are returned (2), from which the user selects his/her best match (3), thus allowing the system to personalize accordingly the distance function for future queries.

tend naturally to the class of Bregman divergences (Nielsen et al., 2007a). In particular, this generalization highlighted the fact that for asymmetric distance functions there exists a convex duality that relates Voronoi diagrams to each other (Fig. 12). This duality collapses for symmetric distances such as the Euclidean distance, and thus could not have been revealed otherwise. That is, studying a broader notion of distances under a same meta-algorithm yielded to a better understanding of Euclidean Voronoi diagrams themselves as self-dual partition structures.

Many famous computational geometry problems can be revisited under the auspices of information geometry. This general abstraction framework will not only enable us to better and deeply understand the nature of the original problems, it will also yield to important novel applications and potential technologies. The next section will describe a case-study where we envision a paradigm shift.

10 Application of computational information geometry: The case study of steering self-learning distance algorithms

The most appropriate distance functions of complex high-dimensional data sets met in computational sciences cannot anymore be guessed manually but rather need to be fully automatically learnt, or even better partially user-steered for personalization (Fig. 13).

The concept of distance expresses the distortion measure between any pair of entities lying in a common space. Distances are at the very heart of geometry, and are ubiquitous in science, needless to say in computational science. From the vantage point of physics, distances may be interpreted as the smallest amount of energy required to go from one location to the other, or to morph from one state to the other. Unfortunately, there is a lot of confusion in popular press about what is exactly meant by using the wording distance. For example, people quite often interchange “distance” with “metric” without caring much about the implicitly underlying mathematical properties: In this case, to know whether the triangle inequality axiom is satisfied or not. A great deal of efforts was achieved by Deza and Deza in 2006 by publishing the first dictionary of distances (Deza and Deza, 2006) presenting succinctly but unambiguously the various properties of distortion measures (eg., metrics, semi-metrics, distances, quasi-distances, divergences, etc.), and listing an extensive although non-exhaustive catalog of principal distances with their domain of applications encountered in both natural sciences (biology, chemistry, physics, cosmology, etc.), and computer sciences (coding theory, data mining, audio/video processing, etc.).

Algorithm designers and researchers in computational sciences daily face the daunting task of choosing the most appropriate distance functions for solving their specific problems. It is clearly understood nowadays that the usual flatland Euclidean distance is rarely appropriate for solving tasks on high-dimensional heterogeneous datasets rather lying on manifolds. Consider for example 3D partial shape retrieval, where a user queries a database of 3D objects with a given part. Solving this problem requires to consider an oriented distance to break the symmetry rule. That is, one would like the distance part→object to be greater than the distance object→part, for all parts belonging to the given object ( ). Indeed, say the distance of a 3D wagon to a 3D train model consisting of a locomotive attached to several wagon units should be strictly greater than the distance of the same 3D train to the said wagon. This kind of asymmetric property is fulfilled by the relative entropy distance, known also as the Kullback-Leibler divergence that acts on statistical distributions. Liu and his colleagues (Liu et al., 2006) built an efficient and accurate 3D part search engine inspired by probabilistic text analysis technique by considering 3D objects as documents covering a small number of topics called "shape topics." They experimentally showed that the relative entropy distance behaves significantly better than the Euclidean or vector space model weighted cosine distances. It is natural to ask oneself whether this Kullback-Leibler distance is the best ultimate distance function for 3D search engines or not?

It turns out that this subtle question cannot be settled in a static way as it depends on the considered input database and on the not-yet-known online queries to be processed in the future. Otherwise, adversorial input sets could be purposely designed to prove the sub-optimality of any prescribed distance function. That is, distances need to be tuned up for every single input set by a built-in learning process. Further, these algorithmic distances need to be dynamically maintained as objects are added, edited, or deleted in the database. This dynamic paradigm of selecting distances bears

much similarity with the recent concept of self-improved algorithms (Ailon et al., 2006) that devote some of their computational time to learn distribution characteristics of the input data sets to be able to speed up the overall process.

Since the space of potential distance functions is uncountably infinite, designing self-learning distance algorithms need to proceed first by choosing a small set of axiom rules (such as symmetry, triangle inequality, separation, etc.) specifying the type of distances, and yielding parameterized distance families for each class. For example, back to 1991, Csiszár axiomatically derived the 1D parametric family of so-called Bregman-Csiszár distances by generalizing the principles of orthogonal projection measures in least-square-type optimization problems (Csiszar, 1991). This generalization let us discovered some counterintuitive facts *a priori*, such as the non-necessarily commutative property of orthogonal projections. The Bregman-Csiszár parametric family includes the Kullback-Leibler and Itakura-Saito divergences at its extremities. Thus learning the best Bregman-Csiszár distance for a given input amounts to find its best member subject to problem-specific constraints.

In information retrieval (IR) systems such as the former partial shape search engine, a set of features playing the role of signature are first extracted from every single input element, and an overall appropriate distance function is properly defined on the signature space. Then, given a query object, its signature data point is first computed (feature extractor) and its nearest-neighbor is searched for among all input signatures. In practice, better classification methods such as taking the major class of the k nearest neighbors, or using kernel machines such as popular support vector machines (SVMs) are employed. Geometrically speaking, the input signatures yields a partition of the signature space into discrete elementary proximity volumes, called Voronoi cells that represent the locus of signature points closer to the cell's signature than to any other input signature. Such a discrete Voronoi diagram implicitly encodes the shape of signature data points. Interestingly, Voronoi diagrams have been recently generalized to the parameterized family of Bregman divergences (Nielsen et al., 2007a and 2007b) as well, unifying both the classic ordinary Euclidean diagram with entropic statistical diagrams into a single template framework.

Furthermore, we seek for efficiency reasons to reduce the number of input signatures to keep only its representative elements. This is achieved by using a technique called vector quantization that clusters the points into groups such as to minimize the intra-cluster distance, while maximizing the inter-cluster distance. The seminal centroid-based k-means cluster algorithm originally established in 1957 by Lloyd has also been recently generalized to its provably most generic class of distance measures by a breakthrough result of Banerjee et al. in 2005: Namely, the class of Bregman divergences (Banerjee et al., 2005).

One can legitimately ponder whether such self-learning distance algorithms are indeed the best suited strategy to get the optimal solution. These tweaked algorithms are indeed presumably the best whenever the objective or loss functions are unambiguously defined from the input datasets. But no one would doubt on the subjective part of defining the "closest" 3D shape to a given collection of 3D shapes. Human perception then plays a determinant role, and answers are all but subjective, reflecting the different tastes of individuals. Therefore another recent line of research is to let users steer themselves the distance learning process by loosely entering preferences. These personal user preferences are entered either by clicking on the best subjective ranked item in a list of top matches, or by providing prior information such as "I find these two images quite similar but these two others are rather far apart, etc." These semi-supervised learning problems become a hot topic in machine learning as attested by the increasing number of publications related to this area. For example, at the last international conference on machine learning (ICML 2007), Hillel and Weinshall described such a semi-supervised learning algorithm where users give positive/negative equivalence constraints denoting intra-cluster/inter-cluster pairs of points (Hillel and Weinshall, 2007). The problem one faces then is to extract as precisely (ie., numerically) and reliably as possible the information provided by users.

In the near future, we envision a whole new generation of scaleable personalized information retrieval systems driven by novel algorithms incorporating self-learning built-in distance modules, and providing light user interfaces. These brand new search engines would be able to better listen to the voice of their users, and more importantly give adequate feedbacks to the full information retrieval engine about users/groups subjective tastes, all at the clicks of a mouse. Today's key challenges in geometric algorithmic are mainly related to tackling high dimensions and guessing the right distortion measures for heterogeneous data sets. There have been already tremendous progresses done over the last decade proposing an approximation framework (based on dimension reduction techniques and core-sets) that bypasses the curse of dimensionality: An exponential running time with respect to the dimension. We believe that our work on proposing and further opening up the new field of computational information geometry offers many opportunities not only for addressing these core challenges but more importantly for creating and solving full fledged sets of novel problems and possibilities popping out from this new theory framework. How exciting is it to be at the heart of this adventure as there are, fore sure, plenty unforecasteable milestones in the open-ended spiral of thoughts.

References

Ailon N., Chazelle B. , Comandur S., Liu D., in: *Proceedings of the 17th ACM-SIAM Symposium on Discrete Algorithms* (SIAM press, Philadelphia, PA, 2006), pp. 261-270.

Amari S., *Differential Geometrical Methods in Statistics*, Springer Lecture Notes in Statistics, 28, 1985.

Amari S. and Nagaoka N., *Methods of Information Geometry*, AMS & Oxford Univ Press, 2000.

Banerjee A., Merugu S., Dhillon I. S., Ghosh J., *Machine Learning Research* 6, 2005.

Chazelle B., "Proof at the Roll of a Dice", in: *Nature*, VOL. 444, pp. 1018-1019, 2006.

Chazelle B., "Coding and Computing Join Forces", in: *Science* VOL. 317, pp. 1691-1692, 2007.

Clarke A. C., "Hazards of Prophecy: The Failure of Imagination", in: *Profiles of the Future*, 1962.

Cover Th. M., Thomas J. A., *Elements of Information Theory*, 2006.

Csiszar I., *Annals of Statistics* VOL. 19(4)., pp. 2032-2066, 1991.

Darwin C., *The Origin of Species*, 1859.

Deutsch D., *The Fabric of Reality*, Penguin Books, 1998.

Deza E., Deza M. M., *Dictionary of distance*, Elsevier Science, 2006.

Gil O., Michor P. W., "The Riemannian manifold of all Riemannian metrics", in: *Quart. J. Math. Oxford*, VOL. 42, Oxford, 1991.

Greenberg M. J., *Euclidean and Non-Euclidean geometries: Development and history*, W. H. Freeman, New York, 1993.

Hillel A.B., Weinshall D., in: *Proceedings of 24th International Conference on Machine Learning*, Omnipress, Madison WI, 2007, pp. 65-72.

Joyce, Euclid's Elements (circa 300 BC), 1997. online: http://aleph0.clarku.edu/~djoyce/java/elements/toc.html

Kaplan F. and Oudeyer Py., 'Intrinsically Motivated Machines' in: Lungarella M., Fumiya I., Bongard J.C., Pfeifer R. (Eds.): *50 Years of Artificial Intelligence*. Lecture Notes in Computer Science 4850 Springer 20, 2006, pp. 303-314.

Kuhn T. S., *The Structure of Scientific Revolutions*, University of Chicago, 3rd edition, 226 pages, 1996 (reprinted from the original 1962 edition).

Kurzweil R., *The Singularity is Near: When Humans Transcends Biology*, 2005.

Liu Y., Zha H., Qin H., in: *Proceedings of IEEE Computer Vision and Pattern Recognition Conference*, IEEE Computer Society, Los Alamitos, CA, 2006, pp. 2025-2032.

Lloyd S. P., *Least squares quantization in PCM*. Technical Note, Bell Laboratories, 1957.

Nielsen F. and Nock R., "ClickRemoval: Interactive pinpoint image object removal", *ACM Multimedia*, 2005.

Nielsen F., "Surround Video: A Multi-Head Camera Approach", *The Visual Computer journal*, VOL. 21 (1-2), Springer-Verlag, 2005.

Nielsen F., *Visual Computing: Geometry, Graphics, and Vision*, Charles River Media, 2005.

Nielsen F. and Nock R., "On approximating the smallest enclosing Bregman Balls", *Symposium on Computational Geometry*, 2006, pp. 485-486/

Nielsen F. and Yamashita N.i., "CLAIRVOYANCE: A Fast And Robust Precision Mosaicing System for Gigapixel Images", *IEEE Industrial Electronics*, DOI 10.1109/IECON.2006.347546, 2006.

Nielsen F., Boissonnat J.-D., Nock R., "Bregman Voronoi Diagrams: Properties, Algorithms and Applications", in: *INRIA Research Report 6154,* Sophia-Antipolis, France, 2007. Also appeared in ACM-SIAM Discrete Algorithms: 746-755, 2007.

Nielsen F., Boissonnat J.D., Nock R., "Visualizing Bregman voronoi diagrams", *Symposium on Computational Geometry*, 2007 pp. 121-122.

Nielsen F., Nock R., "On the symmetrized Bregman centroids", *arXiv*, 2007.

Nock R. and Nielsen F., "Statistical Region Merging", *IEEE Transactions on Pattern Matching and Machine Intelligence*, 2004.

Popper K. R., *Logic der Forschung* (The Logic of Scientific Discovery), Mohr Edition, 1994 (reprinted from the original 1934 edition).

Penrose R., *The Road to Reality : A Complete Guide to the Laws of the Universe*, Alfred a Knopf Inc, 2005.

Preparata F. P. and Shamos M. I., *Computational Geometry: An Introduction*, Springer-Verlag, New York, 1985.

Rae A. I. M., *Quantum Physics: A Beginner's Guide*, Oneworld Publications, 2006.

Rao C. R., "Information and accuracy attainable in the estimation of statistical parameters", *Bull. Calcutta. Math. Soc.* VOL. 37, pp. 81--91,. 1945.

Rissanen J, "Modeling by shortest data description", *Automatica*, VOL. 14, pp. 465-471, 1978.

Robinson S., "Toward an Optimal Algorithm for Matrix Multiplication", *SIAM News* 38(9), November 2005.

Shannon C.E., *A mathematical theory of communication*, Bell System Technical Journal, VOL. 27, pp. 379-423 and pp. 623-656, 1948.

Steels L, *This book*, 2008.

Young J. W., *A Technique for Producing Ideas*, McGraw-Hill Contemporary, 1992.

AFTERWORD

Mario Tokoro

In the spring of 1987, Dr. Toshi T. Doi of Sony visited me when I was studying computer systems and the Internet as an Associate Professor of Keio University. Doi was successfully launched in the preceding year the business of NeWS workstation based on M68000 and 4.2BSD Unix. He knew by intuition that the Sony's future would have everything to do with computer technology, and asked me what Sony should do for it.

At that time, I was involved in a number of advanced and impactful works, brought up talented students and was traveling around the world in order to present my research results at international conferences. The research itself was fulfilling, but the graduate stu-