Ludic Computing Course CO345

Lecture 2

Computational Creativity

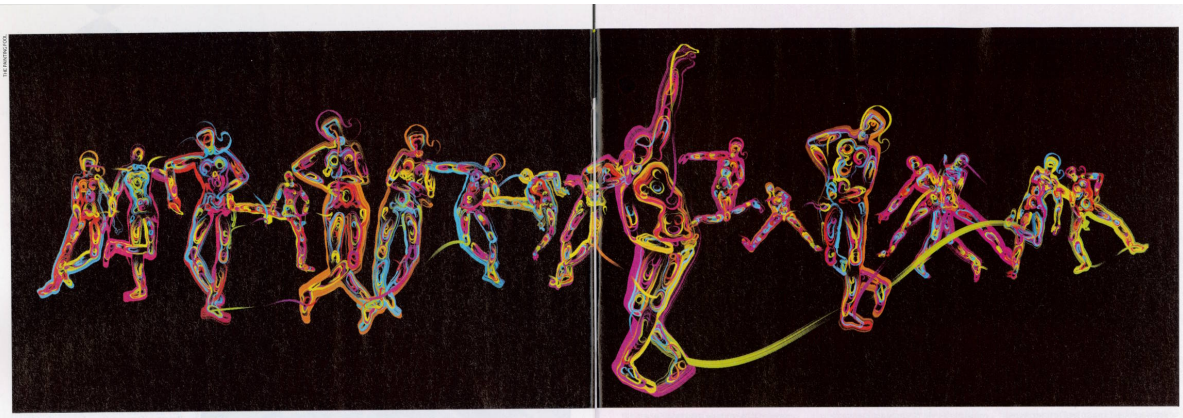
Simon Colton & Alison Pease
Computational Creativity Group
Department of Computing
Imperial College, London

ccg.doc.ic.ac.uk

Overview

- Introductory definitions, aims, objectives, domains
- Quick look at some creative systems (past+present)

- First main issue: generative processes
- Second main issue: evaluating creativity



Creative sparks

Does it really take a human to make a masterpiece? Catherine de Lange sizes up the artificial artists forcing us to change our ideas about creativity

IN A loft overlooking the rooftops of one of the buzzing artistic neighbourhoods of Paris, France, Simon Colton is carefully unfurling one giant painting after another. I have waited some time to see these, and am unsure what to expect. To dislike them would be a disappointment, but easy. If I think these paintings are any good, however, then I might have to reconsider my own creative talents. In fact, they might even challenge my understanding of what it means to be human.

The thing is, these paintings are not the work of an ordinary artist. Nor of Colton, who is a computer scientist based at Imperial College London. Instead, they have been created by a piece of software that can seek artistic inspiration and, arguably, has a rudimentary imagination. Called the Painting Fool, it may have been designed by Colton, but its artwork is its own.

It sounds unlikely that any computer, unguided by the human hand and eye, could create artwork with any feeling or resonance. How could it be creative without having experienced the world? Now, as I take a first glimpse at the paintings, will I be forced to reconsider? Could software, which has no shared experience with my own, create a painting that touches me?

The Painting Fool is one of a growing number of computers which, so their makers claim, possess creative talents. Classical music by an artificial composer has had audiences enraptured, and even tricked them into believing a human was behind the score. Artworks painted by a robot have sold for thousands of dollars and been hung in prestigious galleries. And software has been built which creates art that could not have been imagined by the programmer. "It scares a lot of people," says Geraint Wiggins, a computational creativity researcher at Goldsmiths, University of London. "They are worried it is taking away something special from what it means to be human."

While some animals such as crows and monkeys have displayed traits that could be labelled as limited creativity, we are the only species to perform sophisticated creative acts regularly. If we can break this process down into computer code, where does that leave human creativity? "This is a question at the very core of humanity," says Wiggins.

To some extent, we are all familiar with computerised art. Software that is used to create or manipulate art is ubiquitous, but these are mere tools for a human artist. The question is: where does the work of a person stop and the creativity of the computer begin?

Consider one of the oldest machine artists, Aaron, a robot that has had paintings exhibited in London's Tate Modern and the San Francisco Museum of Modern Art. In some respects, then, Aaron passes some kind of creative Turing test – its works are good enough to be exhibited alongside some of the best human art and people spend good money on them.

Aaron can pick up a paintbrush with its robotic arm and paint on canvas on its own. Impressive, perhaps, but it can never break free from the tightly controlled rules it has been given by its programmer, the artist and founder of machine fine art, Harold Cohen. So I remain unconvinced that Aaron is much more than a tool to realise Cohen's own creative ideas. Colton also dismisses the machine as "rather limited" because "it still only creates one kind of artwork: people in a room with pot plants".

> "The machine will wake up in the morning and look at newspaper headlines for source material"

Colton is keen to make sure the Painting Fool doesn't fall foul of the same criticism, and so has sought to give it as much autonomy as possible. Although the software does not physically apply paint to canvas, it simulates many styles digitally, from collage to paint strokes. One of the first paintings Colton shows me is a touching portrait of a young, fragile-looking girl with porcelain skin and long brown hair. I am impressed that a computer could capture such subtleties, until Colton tells me the software just applied its own painting style to photographs of the girl.

That sounds like cheating to me, but Colton assures me that was an early work. Today, the Painting Fool only needs minimal direction and can come up with its own concepts by going online for source material. "I don't even give it the notion of a person or a topic," says Colton. "It will wake up in the morning and look at the newspaper headlines." The software runs its own web searches and trawls through social media websites such as Twitter and Flickr. The idea is that this approach will let it produce art that is meaningful to the audience, because it is essentially drawing on the human experience as we act, feel and argue on the web.

In 2009, Colton and graduate student Anna Krzeczkowska asked the Painting Fool to

Composed and painted by a computer – can it be enjoyed and lauded by a human being?

produce its own interpretation of the war in Afghanistan, based on a news story. The result is a striking juxtaposition of Afghan citizens, explosions and war graves. "This piece struck a chord with me, and shows the potential for the software to add poignancy and intentionality to its paintings," says Colton.

The Painting Fool is now beginning to display a kind of imagination too, creating pictures from scratch. One of its original works, part of a series that Colton called *Four Seasons*, depicts fuzzy panels of simple landscapes (see image, below). I think it looks rather mechanical.

Comparable to Bach

Having said that, I am swayed by Colton's argument that my reaction arises from my double standards towards software-produced and human-produced art. After all, he says, consider that the Painting Fool painted the landscapes without referring to a photo. "If a child painted a new scene from its head, you'd say it has a certain level of imagination, even if it's just a little bit," he points out. "The same should be true of a machine."

Software bugs can also lead to unexpected results. I see this for myself when Colton shows me some paintings of a chair, which came out black and white thanks to a glitch. It gives the work an eerie, ghostlike quality. Human artists like Ellsworth Kelly are lauded for limiting their colour palate – so why should computers be any different? Nonetheless, these mechanical steps towards creating new styles are barely comparable to the talents of, say, Picasso or Mozart. Or are they?

Researchers like Colton don't believe it is right to compare machine creativity directly to that of humans, who "have had millennia to develop our skills". Others, though, are fascinated by the prospect that a computer might create something as original, emotional and subtle as our best artists. So far, only one has come close.

> "After discovering the truth, one music lover told Cope he had 'killed music' and tried to punch him"

One day, David Cope was suffering from "composer's block". He had been commissioned to write an opera, but was struggling to come up with the goods. If only a computer could understand his style, he thought, and help him write new material. That idea was the starting point for what was to become one of the most controversial pieces of creative software to date. Cope came up with a program called Experiments in Musical Intelligence, or EMI. He fed in musical scores and out popped new material in the composer's style. Not only did EMI create compositions in his style, but also that of the most revered classical composers, including Bach and Mozart.

To my untrained ear, it sounds like any other classical music. I found the purported Chopin, in particular, to be rich and emotional. Audiences who heard the music have been moved to tears, and EMI even fooled classical music experts into thinking they were hearing genuine Bach. If ever there were a successful Turing test for computational creativity, that had to be it.

If a child produced these scenes, you might say they were imaginative

Not everyone was impressed, however. Some critics, such as Wiggins, have blasted Cope's work as pseudoscience, saying his explanation of how the software works is "smoke and mirrors" leaving others unable to reproduce the results (*Literary and Linguist Computing*, vol 23, p 109). Douglas Hofstadter, at Indiana University, Bloomington, says Cope merely scratches at the surface of creativity, using superficial elements of an artist's work to create replicas, which still rely on the original artist's creative impulses.

Nonetheless, for others EMI's ability to mimic Bach or Chopin has serious implications. If it is so easy to break down the style of some of the world's most original composers into computer code, that means some of the best human artists are more machine-like than we would like to think. Indeed, when audiences found out the truth about EMI they were often outraged – one music lover allegedly told Cope he had "killed music" and tried to punch him. Amid such controversy, in 2004 Cope decided that EMI's time was up, and destroyed its vital databases.

But why did so many people love the music, yet recoil when they discovered what composed it? A study by David Moffat, a computer scientist at Glasgow Caledonian University in the UK, provides a clue. He asked both expert musicians and non-experts to assess the creative worth of six compositions. The participants weren't told beforehand whether the tunes were composed by humans or computers, but were asked to guess, and then rate how much they liked each one. Perhaps unsurprisingly, people who thought the composer was a computer tended to dislike that piece more than those who believed it was human. This was true even among experts, who you might think would be more objective in their analysis of musical quality.

Where does this prejudice come from? Psychologist Paul Bloom of Yale University has a suggestion: he reckons part of the pleasure we get from art comes from our perception of the creative process behind it. This can give it an "irresistible essence", says Bloom. This idea explains why a painting loses its value if exposed as a fake, even though we might have loved it when we thought it was an original. Indeed, experiments by psychologist Justin Kruger of New York University have shown that people's enjoyment of an artwork increases if they think more time and effort was needed to create it (*Journal of Experimental Social Psychology*, vol 40, p 91).

Similarly, Colton thinks that when people experience art, they engage in a discourse with the artist. We wonder what the artist might have been thinking, or ponder what they are trying to tell us. With computers producing art, this speculation is cut short – there's nothing to explore. But as the software becomes increasingly complex, finding those greater depths in the art may become possible. That's why Colton asks the Painting Fool to tap into online social networks for its inspiration: hopefully this way it will choose themes that will already mean something to us.

Our enjoyment of art is influenced by the time and effort it required

Unconscious creativity

Hofstadter thinks the more complex machines become, the more easily we will accept their art – especially if they can interact more with the physical world. If robots bumped into things and had goals, successes and failures, then that might be enough. "They would be sort of pathetic and laughable and once in a while heroic," he says. "I don't think people would be uncomfortable with creatures like that writing an essay or composing a piece of music or painting a picture."

Yet the fact that machines now lack this kind of self-awareness is perhaps the most irksome element of computational creativity. How can you be creative without even being conscious? Surprisingly, it is not a computer scientist who talks me out of this reaction, but a neuroscientist. Our brains work creatively even when we aren't consciously thinking about it, says Arne Dietrich at the American University of Beirut in Lebanon. Just think back to a time when the solution to a problem you had forgotten about just popped into your head. "We know that there are several different types of creativity – some of them are conscious, some of them unconscious," he says. "Creativity can happen when you try effortfully, or it can happen in your sleep."

In any case, Dietrich believes that the creative brain might work much like software. Neuroscientists suspect that creativity is essentially about discovery, rather than anything mystical. "It's a mechanical process in the brain that generates possible solutions and then eliminates them systematically," Dietrich says. He believes our tendency to dismiss computational creativity as inferior to our own comes from an ingrained dualism in human culture: "We are over-evaluating ourselves and underestimating them. As a neuroscientist, I tackle the brain as a machine, and I don't see machine creativity as different." Suddenly, the idea that the human brain has a unique claim to creative talents seems a limited perspective.

Back in Paris, Colton continues to show me painting after painting, all signed by the Painting Fool. Some of the work genuinely speaks to me. One of my favourites, called *The Dancing Salesman Problem*, features colourful human figures dancing on a black background (see main image, page 42). Again, the software did not base its composition on existing pictures. The dancers are painted in long, flowing strokes, so they appear full of movement: they contort into beautiful poses, and the bright colours bring the scene to life. The work could never be to everyone's taste, but I would have stopped to look at it in a gallery, and I don't mind that it was created by a machine.

I have come to appreciate that computers can create subtle and original artwork. Will others accept that idea? The trick, says Colton, is to stop trying to compare computer artists to human ones. If we can embrace computer creativity for what it is, and stop trying to make it look human, not only will computers teach us new things about our own creative talents, but they might become creative in ways that we cannot begin to imagine. They are creating a whole new form of art with the potential to delight, challenge and surprise us.

Will that take something away from being human? "It's not taking away anything at all," says Wiggins. "It's helping us to understand how things work. And when you understand how things work they tend to become more amazing, not less so." ■

Catherine de Lange is a writer based in London.

To see more examples of software-produced artwork, visit newscientist.com/gallery/painting-fool

are conscious, some of them unconscious," he says. "Creativity can happen when you try effortfully, or it can happen in your sleep."

In any case, Dietrich believes that the creative brain might work much like software. Neuroscientists suspect that creativity is essentially about discovery, rather than anything mystical. "It's a mechanical process in the brain that generates possible solutions and then eliminates them systematically," Dietrich says. He believes our tendency to dismiss computational creativity as inferior to our own comes from an ingrained dualism in human culture: "We are over-evaluating ourselves and underestimating them. As a neuroscientist, I tackle the brain as a machine, and I don't see machine creativity as different." Suddenly, the idea that the human brain has a unique claim to creative talents seems a limited perspective.

Back in Paris, Colton continues to show me painting after painting, all signed by the Painting Fool. Some of the work genuinely speaks to me. One of my favourites, called *The Dancing Salesman Problem*, features colourful

The work could never be to everyone's taste, but I would have stopped to look at it in a gallery, and I don't mind that it was created by a machine.

I have come to appreciate that computers can create subtle and original artwork. Will others accept that idea? The trick, says Colton, is to stop trying to compare computer artists to human ones. If we can embrace computer creativity for what it is, and stop trying to make it look human, not only will computers teach us new things about our own creative talents, but they might become creative in ways that we cannot begin to imagine. They are creating a whole new form of art with the potential to delight, challenge and surprise us.

Will that take something away from being human? "It's not taking away anything at all," says Wiggins. "It's helping us to understand how things work. And when you understand how things work they tend to become more amazing, not less so." ■

Catherine de Lange is a writer based in London.

To see more examples of software-produced artwork, visit newscientist.com/gallery/painting-fool

14 January 2012 | NewScientist | 45

Definitions

- The sub-area of AI research, where we study the building of software which exhibits behaviour that would be deemed creative if exhibited by a person
- To build software which takes on some of the creative responsibility in arts and science projects
- Not just the usage of computers in creative tasks

Aims

- To build autonomously creative software, for practical purposes, or as an intellectual challenge
- To help us to understand more about human creativity, via computational models
- To help people to become more creative, via the building of collaborative creative systems
- To stretch existing AI systems (and combinations thereof) to breaking point, and suggest improvements

Be Slightly Afraid...

- In the past, words such as *creativity, emotion, self-awareness,* etc., were used to beat AI researchers with
  - So there is still some resistance in the AI world to creativity research
- And even more mis-understanding (denial...?) outside of AI circles
  - E.g., Strothotte and Schlectweg in *'Non-photorealistic Computer Graphics'*: "Simulating artistic techniques means also simulating human thinking and reasoning, especially creative thinking. This is impossible to do using algorithms or information processing systems"
- And it gets even worse when journalists come along:
  More4 News, when reporting a computer art exhibition:
  "Is this some kind of hellish nightmare...?"

Overlap(s) with Ludic Computing

- Procedural content generation
  - More interesting content for games, e.g., stylistic cities, weird vegetation, etc.
- Adaptive video games
  - Automatic game direction requires creativity
- Computer art (filtering, NPR, shape grammars)
  - Essential graphics components in building automated painters
- Assessment of enjoyment in a ludic system
  - May be partially dependent on our perception of creativity in the system

Application Domains

- Linguistics: generation of jokes; poems; plot lines; neologisms; acronyms; dialogue; stories;
- Music: melodies; harmonisations; structures; compositions; accompaniments; interactive systems;
- Visual arts: evolutionary art; scene generation; automated painting; pixel shading; mathematical art;
- Pure mathematics: concept invention; conjecture making; theorem proving;
- Other domains: science (in particular bioinformatics); video game design (e.g., level generation), ...

A Sampling of

Early Named Systems

- Doug Lenat's *AM* (1975 - 1985)
  - Automated mathematician
- Harold Cohen's *AARON* (1973 - today)
  - Automated painter
- David Cope's *EMI* system (1981 - today)
  - Automated composer
- William Chamberlain and Thomas Eiter's *Racter* (1984-1986)
  - Automated poet

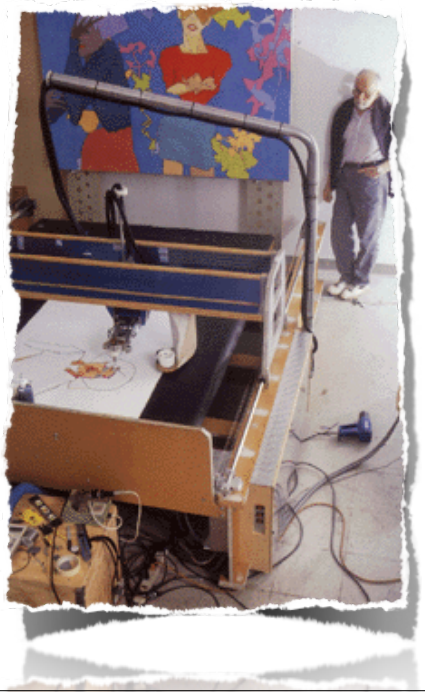
Sample Output

- AM re-inventions:
  - Goldbach's conjecture, highly composite numbers

More than iron, more than lead, more than gold I need electricity.
I need it more than I need lamb or pork or lettuce or cucumber.
I need it for my dreams. Racter, from *The Policeman's Beard Is Half Constructed*

Sample Output

- AARON



A Sampling of

More Recent Named Systems

- Penousal Machado's *NEvAr* (2000 - today)
  - Evolutionary artist
- Kim Binsted + Graeme Ritchie's *Jape* (1995 - today)
  - Automated joke teller
- Rafael Perez y Perez's *Mexica* (1999 - today)
  - Automated story teller
- Pablo Gervas's *ASPERA* (2000 - today)
  - Automated poet

Sample Output

Jokes by Jape

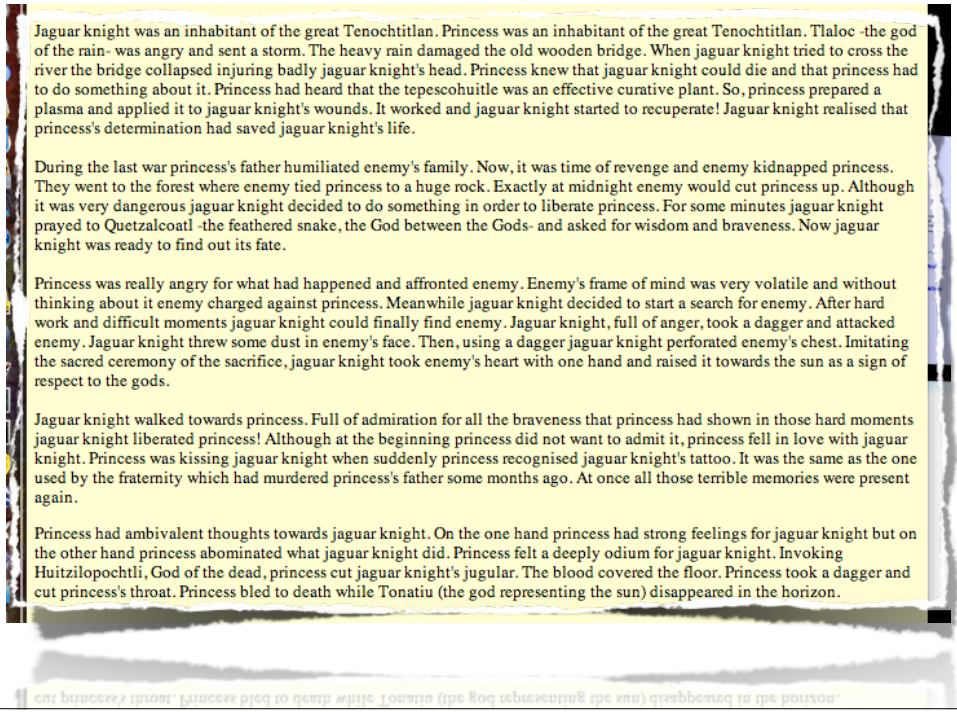
- What's the difference between money and a bottom?
  One you spare and bank, the other you bare and spank!
- What's the difference between gardening and driving?
  One you brush and rake, the other you rush and brake!
- What do you call a strange market?
  A bizarre bazaar!
- What do you call a sick bird?
  An ill eagle!

- Which one is by a human.....?

Sample Output

Mexica Story

www.rafaelperezyperez.com/MEXICA



Jaguar knight was an inhabitant of the great Tenochtitlan. Princess was an inhabitant of the great Tenochtitlan. Tlaloc -the god of the rain- was angry and sent a storm. The heavy rain damaged the old wooden bridge. When jaguar knight tried to cross the river the bridge collapsed injuring badly jaguar knight's head. Princess knew that jaguar knight could die and that princess had to do something about it. Princess had heard that the tepescohuitle was an effective curative plant. So, princess prepared a plasma and applied it to jaguar knight's wounds. It worked and jaguar knight started to recuperate! Jaguar knight realised that princess's determination had saved jaguar knight's life.

During the last war princess's father humiliated enemy's family. Now, it was time of revenge and enemy kidnapped princess. They went to the forest where enemy tied princess to a huge rock. Exactly at midnight enemy would cut princess up. Although it was very dangerous jaguar knight decided to do something in order to liberate princess. For some minutes jaguar knight prayed to Quetzalcoatl -the feathered snake, the God between the Gods- and asked for wisdom and braveness. Now jaguar knight was ready to find out its fate.

Princess was really angry for what had happened and affronted enemy. Enemy's frame of mind was very volatile and without thinking about it enemy charged against princess. Meanwhile jaguar knight decided to start a search for enemy. After hard work and difficult moments jaguar knight could finally find enemy. Jaguar knight, full of anger, took a dagger and attacked enemy. Jaguar knight threw some dust in enemy's face. Then, using a dagger jaguar knight perforated enemy's chest. Imitating the sacred ceremony of the sacrifice, jaguar knight took enemy's heart with one hand and raised it towards the sun as a sign of respect to the gods.

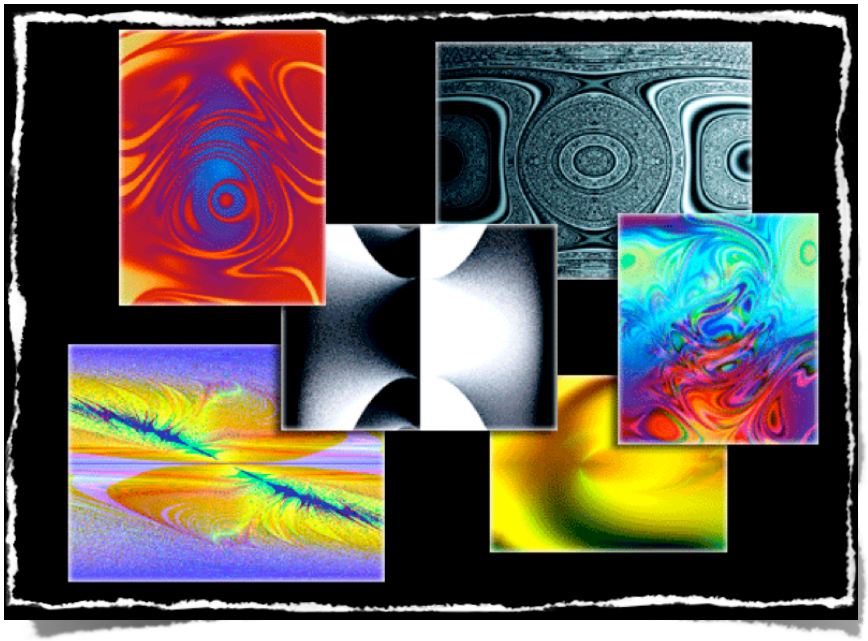
Jaguar knight walked towards princess. Full of admiration for all the braveness that princess had shown in those hard moments jaguar knight liberated princess! Although at the beginning princess did not want to admit it, princess fell in love with jaguar knight. Princess was kissing jaguar knight when suddenly princess recognised jaguar knight's tattoo. It was the same as the one used by the fraternity which had murdered princess's father some months ago. At once all those terrible memories were present again.

Princess had ambivalent thoughts towards jaguar knight. On the one hand princess had strong feelings for jaguar knight but on the other hand princess abominated what jaguar knight did. Princess felt a deeply odium for jaguar knight. Invoking Huitzilopochtli, God of the dead, princess cut jaguar knight's jugular. The blood covered the floor. Princess took a dagger and cut princess's throat. Princess bled to death while Tonatiu (the god representing the sun) disappeared in the horizon.

The Princess who Cured the Jaguar Knight.

Sample Output

NEvAr Pictures



A Sampling of

CCG Named Systems

- HR system (1996 - today)
  - Mathematical discovery system
- The Painting Fool (2005 - today)
  - Automated Painter (www.thepaintingfool.com)
- Filter Feast (2007 - today)
  - Image Filter based platform
- GC (2006 - today)
  - Combined reasoning platform

Two Main Issues

- Achieving creativity in artefact generation
  - AI techniques employed and methodological issues
- Scientifically assessing progress towards creativity in individual computational systems
  - Assessing the artefacts
  - Assessing the processes

Artificial Intelligence Paradigms

- Problem solving paradigm (most of AI research)
  - An intelligent task is interpreted as a set or sequence of problems to be solved by individual problem solving agents
  - Specialist problem solvers are employed, such as theorem provers, planners, machine learning systems, constraint solvers, etc.
  - Occasionally, combinations of reasoning systems are used for greater power
- Artefact generation paradigm (computational creativity research)
  - An intelligent task is interpreted as a desire to generate something of cultural value, whether of aesthetic and/or utilitarian value
  - Specialist generative techniques and artefact evaluation modules are employed, such as evolutionary methods, conceptual blending, fitness functions, etc.
  - Occasionally, mash-ups are used for greater creative potential
- Big philosophical difference: problem solving agents are designed to think for us, whereas artefact generation agents are designed to make us think more. Why is that....?

Minor AI Methods in Computational Creativity

- AI methods tend to be for problem solving
- Some simply say yes or no, others bring new information into being.
- These include *constraint solvers* (finding models) *theorem provers* (discovering proofs) and *machine learning systems* (inventing classifiers)
  - All of these are used as component systems to some extent in certain creative systems
  - For instance: constraint solvers are used in harmonisation systems; machine learning systems learn certain musical styles, and theorem provers are used in mathematical discovery tasks

Major AI Methods and Methodologies in Computational Creativity

- Methods
  - Evolutionary programming
  - Case-based reasoning
  - Conceptual blending
- Methodologies
  - Analogical and metaphorical reasoning
  - Combining systems
  - Handing over creative responsibility

AI Methods

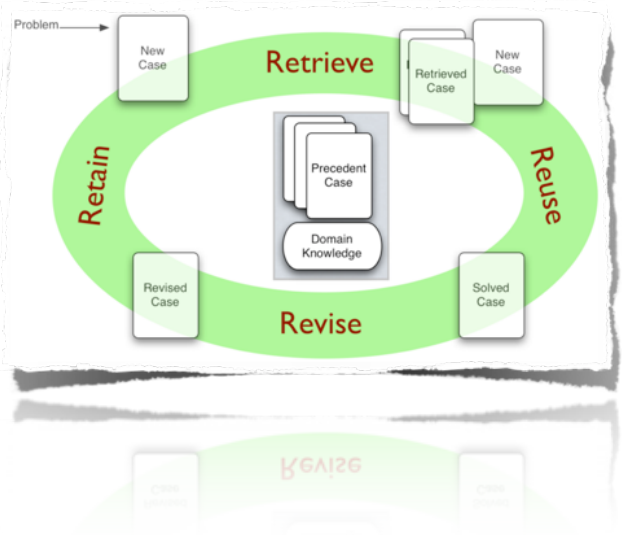
Evolutionary Approaches

- Programs (often represented as trees) are evolved at the genotype level, artefacts are produced by the programs at the phenotype level
  - By appealing to evolutionary processes such as crossover, mutation, survival of the fittest
- Works well for creative tasks because:
  - Often, very little domain knowledge is required, OK if "you know what you like"
  - The evolution often tends to produce surprising solutions (artefacts)
  - Fitness functions can be defined which have utilitarian and aesthetic aspects
  - Fitness can also be supplied entirely by the user (user-driven evolutionary approach)
  - Lends itself to higher level reasoning, e.g., invention of fitness functions
- Drawbacks: can be difficult to drive evolution; can take a long time to converge
- Examples of usage: entire workshops on EvoMusArt and EvoGames, scientific discovery
- See lectures 3 and 4 for further details

AI Methods

Case-based Reasoning

- Solve current problem by appealing to previous cases which worked
  - Four step process of retrieval, re-use, revision and retention
- Works well for creative tasks because it appeals to a set of known high value artefacts and alters them to fit the current context, hence likely to yield interesting and valuable artefacts
- Drawbacks: requires the building of a large case-base; could be seen as producing only pastiches of other people's work
- Example of usage: MuzaCazUza system by Ribeiro, Pereira, Ferrand and Cardoso:
  - Composition of melodies based on previous ones



http://cbrwiki.fdi.ucm.es/wiki/

AI Methods

Conceptual Blending

- Theory from cognitive linguistics
  - Attempts to explain aspects of human linguistic reasoning. Blending of concepts from different spaces. (Fauconnier and Turner). Example: horse and bird concepts blend to give Pegasus
- Works well for creative software because it is cognitively plausible and can lead to surprising conceptual blends
- Drawbacks: doesn't discuss where the conceptual spaces come from in the first place
- Examples of usage:
  - Linguistic creativity: neologisms (see later)
  - Creature generation for video game design (see next slide for an example)
  - Part of a proposed pedagogical model for children learning through video game playing (www.johnseelybrown.com/playimagination.pdf)

To explain it in some detail, we must introduce the concept of Mental Space. According to [1], Mental Spaces are partial structures that proliferate when we think and talk, allowing a fine-grained partitioning of our discourse and knowledge structures. As we talk or think, our reasoning focus flows from space to space, transporting and mapping concepts according to points of view, presuppositions, beliefs, changes of mood or tense, analogical counterfactuals and so on, each giving birth to a different mental space.

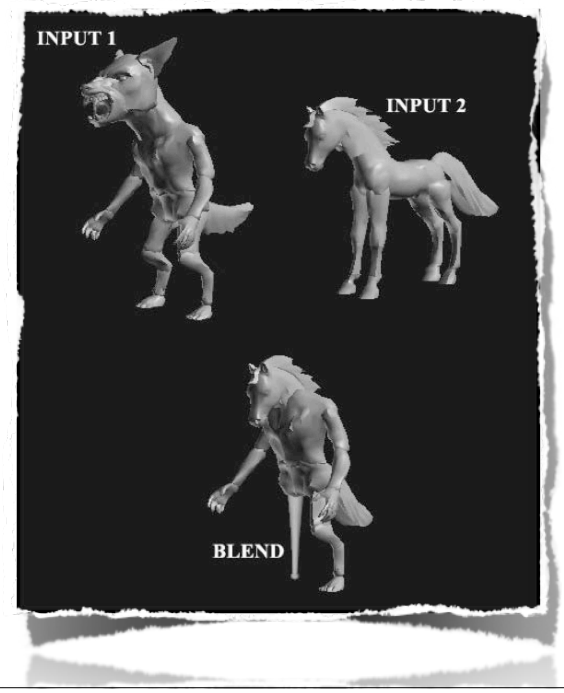
Blending is generally described as involving two input mental spaces that, according to a given structure mapping, will generate a third one, called Blend. This new domain will maintain partial structure from the input domains and add emergent structure of its own.

From: *"Conceptual Blending and the Quest for the Holy Creative Process"* by Pereira and Cardoso (http://eden.dei.uc.pt/~camara/files/QuestCRC.pdf)

AI Methods

Conceptual Blending

Application to Creature Design for Games



From: P. Ribeiro, F. C. Pereira, B. Marques, B. Leitao and A. Cardoso, *"A model for creativity in creature generation"*, in Proceedings of the 4th Conference on Games Development (GAME ON'03), 2003

Methodological Issue #1

Metaphor and Analogy

- All good artists "steal", whereas scientists "stand on the shoulders of giants..."
  - All of cultural activity can be seen as a progression based on past work
- Metaphorical and analogical reasoning has strong creative potential in people
  - Many psychological and CogSci studies
- The three previous techniques can be seen as analogy driven (children are based on parents; previous cases and concepts are used to generate new artefacts)
- When building creative systems, consider whether they can:
  - Learn styles or methods from others (people or systems)
  - Build new artefacts from old ones
  - Take previous successful reasoning schemes and apply them in new situations
- See Hofstadter, Thagard, Veale, Barnden, ...
  - Particularly in linguistic creativity

Methodological Issue #2

Combining Systems

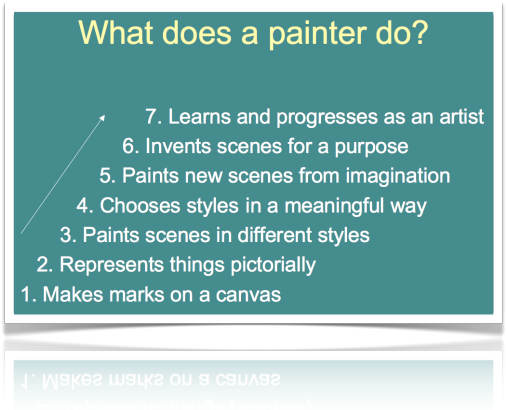
- Individual methods, etc., are designed for one thing, sometimes with multiple techniques available from within a single GUI (e.g., Photoshop)
  - Entirely driven by the user's aesthetic and utilitarian choices
- If we want autonomously creative systems, we have to enable them to use many different pieces of software, just like we do...
- Ultimately, the whole should be more than the sum of the parts:
  - More ability; more efficiency; more flexibility; more novelty and surprise
- When building creative systems, consider whether:
  - You are re-inventing the wheel, by implementing techniques that have been previously developed
  - You could achieve your aim by combining existing AI/Multimedia/etc. software in a mash-up
  - You can use a systematic combination method (e.g., multi-agent systems), rather than building ad-hoc software
  - Your combined system can be used for tasks which you didn't previously consider (perhaps by changing a module)

Methodological Issue #3

Handing over Creative Responsibility

- Ultimately, we want more than mere creativity support tools
  - We already have loads of great human artists, scientists, musicians, writers, etc.
  - There is the potential for creative software to add to our cultural life in new, interesting, surprising and unforseen ways
- Climbing the meta-mountain
  - This involves successively asking: "How am I using this software tool for my creative purposes? Can I get the software to do that itself? How can I implement those behaviours?"
  - Practically involves implementing more skills and critical abilities (to take over aesthetic responsibilities)

Meta-mountain for The Painting Fool



Assessing Creativity

- It's very difficult to assess creativity in people, animals and computer systems, so why bother?
  - To enable us to progress, by comparing new versions of creative systems to older versions - and to show a progression
  - To compare and contrast generic generative techniques, to see which is the best under which circumstances
  - To address our critics with arguments about why the software should be considered creative
  - To help clarify failings and implement better techniques
- Two main approaches to assessment:
  - Assess the artefacts produced; assess the underlying process

Assessing Artefacts

- In some domains, how artefacts were generated is immaterial, e.g., jokes, theorems
- We can evaluate a system in terms of its output alone
- Boden, in "The Creative Mind: Myths and Mechanisms" suggests two axes of assessment:
  - P-creativity vs H-creativity (personal and historical)
    - Of the artefact
  - Exploratory vs transformational creativity
    - Of the search space
  - Wiggins provides a formalism for this kind of assessment

Assessing Artefacts

Ritchie's Measures for Artefacts

Journal: Minds and Machines 17(1)

- Formalism involves a classification of artefacts that can be generated as either of the type required or not
  - e.g., a joke generator might produce non-joke strings, ignoring any humour value at this stage
  - Also requires a rating scheme and the notion of an inspiring set of example artefacts to be created
- Software is more creative if it produces artefacts outside of the inspiring set (novelty), and of higher score than average according to the rating scheme (value)
- But what about when the generation of aesthetic fitness considerations is part of the creative process?

Assessing Creative Processes

- In many domains, aspects of the cognitive and practical processes used to create an artefact are taken into account when assessing that artefact
  - In particular modern art, e.g., Duchamp's urinal
  - Especially the creativity involved in the production, which leads to a vicious circle in computational creativity, as the general impression is a lack of creativity
- One solution: manage the perception of creativity that people have, addressing their criticisms with specific implementations
  - Eventually, people will have to admit that software is creative because they cannot argue otherwise
- Creativity tripod: main arguments of criticism are along three axes:
  - Skill, appreciation and imagination
  - We can pro-actively implement behaviours to increase intelligence in these areas

The Wundt Curve

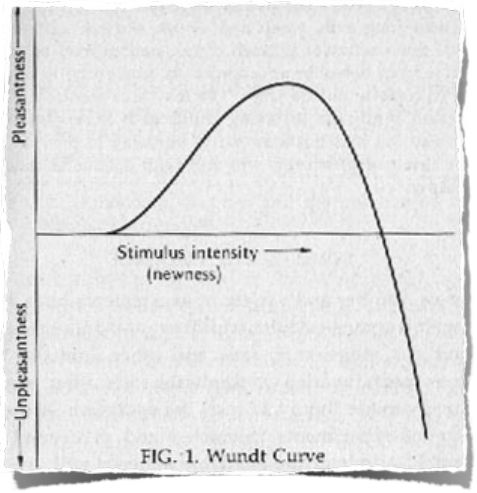


FIG. 1. Wundt Curve

- In exploratory creativity, some novelty is required to maximise hedonistic value, but too much is off-putting
- But is this how the cubists, or the abstract expressionists changed our view of representation art? Did they instead move the axes?

To Turing Test...
... or Not to Turing Test?

- Creative abilities in software differ between domains
- In some domains, producing human-quality artefacts is still quite difficult, especially in linguistic domains, perhaps because of the information density and lack of robustness of the written word (mistakes may be more easily tolerated in art, music and science domains)
    - In such difficult domains, it makes some sense to compare computer generated artefacts with human ones as in a Turing test, e.g., if people agree that Jape's joke are indistinguishable from joke-book jokes, this is a success
- However, when software passes the milestone of being able to produce culturally valuable artefacts that a person might not be able to produce, is the Turing test still a good way to assess its value?
    - An easy criticism of software like photoshop is that it produces only pastiches
    - And wouldn't an art school graduate be horrified if someone could not tell their work apart from another artist?
- Ultimately, software should surprise us, which might go against a Turing test...

Summary

- Creative software is out there
  - Working in artistic and scientific domains
- We have to worry about two main issues
  - Artefact generation and assessment of creativity
- Computational creativity is a sub-area of AI, but it is distinct in many ways
  - There are questions over the value of the Turing test
  - We don't necessarily want reliability, we want surprise and novelty
  - CC software is designed to make people think more, not less
- Has applications to the building of software for ludic purposes
  - Procedural content generation for games, in particular

(Non-examinable.....!)

Further Reading

- Computational Creativity Autumn School slides (from ccg.doc.ic.ac.uk/wiki/doku.php?id=create)
- Papers cited here
- AI Bite on computational creativity
  - www.doc.ic.ac.uk/~sgc/papers/colton_aisbq126.pdf
- Wiki Page
  - http://en.wikipedia.org/wiki/Computational_creativity
- AI Magazine Special Issue (esp. editorial)
  - www.doc.ic.ac.uk/~sgc/papers/colton_aimag09_editorial.pdf