On Acid Drops and Teardrops: Observer Issues in Computational Creativity

Simon Colton, Michael Cook, Rose Hepworth[1] and **Alison Pease**[2]

Abstract. We argue that the notion of creativity in a person or software is a secondary and essentially contested concept. Hence, in Computational Creativity research – where we aim to build software taken seriously as independently creative – understanding the roles people take as process observer and product consumer is paramount. Depending on the domain, there can be a natural bias against software created artefacts, and Computational Creativity researchers have exacerbated this situation through Turing-style comparison tests. Framing this as a modified Chinese Room experiment, We propose two remedies to the situation. These involve software accounting for its decisions, actions and products, and taking the radical step of thinking of computer generated artefacts as fundamentally different to their human-produced counterparts. We use two case studies, where people interact with an automated painter and with computer-generated videogames, to highlight the observer issues we raise, and to demonstrate partial implementations of our remedies.

1 Introduction

The definition of Computational Creativity research as a subfield of Artificial Intelligence research given in [1] is as follows:

> *The philosophy, science and engineering of computational systems which, by taking on particular responsibilities, exhibit behaviours that unbiased observers would deem to be creative.*

While this definition is not universally accepted, there have been no significant challenges to it so far, and variations of it have been used to describe the field for many years.

There are a number of things to notice about this definition. Of most interest here is the prominence of the notion of 'unbiased observers', whose judgement about the behaviours of software systems is to be used as a prime evaluation tool. We also note that while it is natural to assume human observers, the definition does not explicitly say that software must behave in ways deemed to be creative in people. This leaves open the possibility that people could become accustomed to using the word 'creative' to describe non-human-like behaviour. In fact, for many in the field, achieving human-level or human-like creativity is not the goal, but rather expanding the notion of creativity to include computational behaviours and/or enjoying the fruits of non-human-like creativity are more interesting aims.

The usage of the word 'unbiased' in this definition hints at a problem encountered when evaluating projects where generative software produces artefacts (poems, paintings, sonatas, recipes, theorems, etc.) for human consumption. In particular, people can have natural biases against, and occasionally in favour of artefacts produced by computers over those produced by people. Negative, so called 'silicon', biases have been observed under experimental conditions [2, 3]. Hence, in stipulating that observers must be unbiased, the definition above emphasises a scientific approach to evaluating progress in the building of creative systems, whereby experimental conditions are imposed to rule out, or otherwise cater for, such biases. One such experimental setup is the Turing-style comparison test, where computer-generated and human-produced artefacts are mixed and consumers make choices between them with zero context given about the processes involved in their production.

Such experimental conditions are not sustainable if we are to enhance society with creative software, and biases about machine creation need to be embraced and managed. To begin this, we advocate bringing observer issues in Computational Creativity into the open, and in section 2 we present a perspective about human creativity which acts as a foundation for the later exposition. In section 3, we express the problems faced in Computational Creativity and argue that Turing-style comparison tests actually serve to make things worse, rather than better. Addressing the problem as a modified version of Searle's Chinese Room experiment in section 4, we propose two potential remedies. The first involves creative software being more accountable, while the second involves considering computer-generated artefacts as being fundamentally different to their human-produced counterparts. To illustrate the issues raised and remedies proposed, in sections 5 and 6, we use two case studies where people have interacted with an automated artist and computer-generated videogames respectively. We conclude by considering a future where software is routinely perceived as creative, and how observer management is important in realising such a prospect.

2 A Perspective on Creativity

We hold that creativity is a secondary and essentially contested quality of a person, and that linguistic usage of terms related to creativity can often be declarative illocutionary speech acts. We unpack these assertions below.

We believe that attributions of creativity are contextualist, having no truth value which is independent of context, perception and interpretation. In this way we see creativity attributions as analogous to the Lockean notion of a secondary quality [4]. Locke distinguished *primary* and *secondary qualities*, where the former is intrinsic to an object, such as mass, and the latter is perception-dependent, such as colour. While these Lockean qualities are directly tied to sensory perception, as opposed to creativity, the distinction is still a useful one here, since it highlights different types of properties. Dennett's intentional stance [5] is also of interest here: we may adopt a "cre-

[1] Computational Creativity Group, Department of Computing, Goldsmiths, University of London ccg.doc.gold.ac.uk

[2] School of Computing, University of Dundee

ativity stance" towards a person and interpret their work *as though they were being creative*, in order to better understand (rather than predict) their behaviour. Likewise, we may find that the "creativity stance" provides a new way of understanding the behaviour of a piece of software which goes beyond the physical details of the program.[3]

Gallie introduced *essentially contested concepts* as those for which "the proper use . . . inevitably involves endless disputes about their proper uses on the part of their users" [6, pp. 169], to which Gray added that the disputes ". . . cannot be settled by appeal to empirical evidence, linguistic usage, or the canons of logic alone" [7, pp. 344], and Smith noted that ". . . all argue that the concept is being *used inappropriately* by others" [8, pp. 332]. In the *Cambridge Handbook of Creativity*, Plucker and Mabel assert that:

> . . . despite the abundance of definitions for creativity and related terms, few are widely used and many researchers simply avoid defining the relevant terms at all. [9, p 48]

Clearly, certain notions such as *art* are essentially contested concepts, looking at the multitude of articles written each year in the popular and cultural press asking: "But is it Art?" Indeed, Gallie points out that the assertion: "This picture is painted in oils" can be disputed with the disputants agreeing on the proper usage of the terms involved, whereas the assertion "This picture is a work of art" is likely to be contested

> . . . because of an evident disagreement as to – and the consequent need for philosophical elucidation – of the proper general use of the term "work of art" [6, pp. 167].

As a very recent example, the question of whether videogames should be classed as art was raised by a Guardian art critic [10], to which the Guardian games editor responded:

> Here is a good way to tell if a critic is having a moment of madness: they will attempt to define art. The greatest philosophers in history have floundered on the question, many simply avoided it altogether, preferring to grapple with more straightforward questions – like . . . the existence of God. Art is ethereal, boundless, its meaning as transient as the seasons. When you think you have grasped it, it slips through your fingers [11].

While this is only one example, it serves as an exemplar of the kinds of debates that occur daily in the arts about the nature of art.

While the preoccupation with expressing creativity is a relatively modern aspect of the visual arts, it is clear that if the notion of art is essentially contested, then the notion of the creativity that went into producing art should be seen accordingly, which is a position taken in [12], with which we agree. We can further justify the idea that proper usage of the term *creativity* involves endless debate about its proper usage by reference to the multitude of volumes written about improving, managing and assessing creativity in people, organisations and society. As a society, we are not meant to agree on what creativity means, in the sense that the disputes we have about this are an engine for change and progress in society, and it would surely be stultifying if we all suddenly agreed on this most important of concepts. Therefore, while it is problematic for various areas of study – not least Computational Creativity – that creativity is an essentially contested quality of any person, it is something we need to embrace and even celebrate. For more in-depth discussion of creativity in this context, see the work of Jordanous [12, chapter 3].

Austin informally introduced the linguistic notion of an *illocutionary act* as one which somehow performs an action [13], and Searle further categorised such speech acts into: *assertives*, *directives*, *commissives*, *expressives* and *declarations* [14]. Of these, *declarations* is of most interest here; these acts are defined as changing reality in accordance with the proposition stated. A good example of such a speech act is: "I pronounce you husband and wife". We believe that – in certain circumstances – people can bestow the reality of a person being creative simply by stating it. To see this, we recall the contested nature of creativity, thus it is tacitly assumed that there is no consensus about this. It therefore follows that people who are not particularly interested either way about the creativity (or lack thereof) of someone else, can be easily swayed by the declarative speech act of a third party in a position of authority. When Nicholas Serota, long time director of the Tate art museums and galleries, says that a piece is a great work of art, that work becomes (at least temporarily) a great work. When he states that a particular artist is unusually creative, who are we to argue? More than this, given that the sentence 'X is creative' is shorthand for: 'Most people agree that they perceive X to be creative', by speaking for the general public, such authorities can essentially bring into being the creativity of X, regardless of whether X perceives him/herself as such.

3 Observer Issues in Computational Creativity

To the best of our knowledge, no-one has yet debated the question of whether it is appropriate to use terms related to creativity to describe behaviours of software in similar ways to how they are used to describe behaviours of people. One could argue that, given the particularly human-centric notion of creativity, and that a human connection is paramount in much of the arts, it is simply inappropriate to use the term 'creative' to describe software. Naturally, there have been many commentators (including ourselves) who have stated their opinions about whether software can or can't be creative in general, but a set of such opinions doesn't comprise a debate about the proper usage – if there is one – of the word 'creativity' in a computational context.

Possibly given the essentially contested nature of the notion of creativity in people, and possibly due to a general lack of interest, the status quo is that we currently haphazardly apply human terminology related to creativity to software. This often requires the projection of other human qualities onto software (for instance, we recently had a discussion about poetry generating software appearing *juvenile*, without once questioning what juvenile means in the context of software). Given this status quo, we must project the secondary nature of creativity as a human quality into the computational sphere. Clearly then, the perception of creativity in software needs to be managed [15], and studying the role of people as both observer of algorithmic process and consumer of automatically generated artefacts is of much importance in Computational Creativity research.

To begin to address various observer issues, we compare the interpretation of computer-generated and human-produced artefacts in a rather extreme situation, where knowledge about the personality of an artist and their practice is entirely missing. There have been a number of movements where art practitioners are adamant that the aesthetic value of artworks is the only thing that matters, and that all other socio-political or conceptual considerations are irrelevant. For instance, addressing an inherent *Intentional Fallacy*, Wimsatt and Beardsley asserted that "the design or intention of the author is neither available nor desirable as a standard for judging the success of a work of literary art". They reject the stance that: "In order to judge the poet's performance, we must know what he intended" [16, pp3].

[3] We are grateful to an anonymous reviewer for pointing this out.

As an earlier variation, the *Aesthetic Movement* [17] advocated that art should have no didactic purposes, existing only to enhance one's life through its beauty (and that life should imitate art in this sense). In such situations, process, personality and context are irrelevant to the viewer/reader/listener of such aesthetic pieces. A modern take on this is that of the *Stuckists* [18], who emphasise the expression of emotion and personal experience, but eschew conceptual art and the discussion of conceptual back-stories to art, while hailing the amateur artist. In addition to art movements, many individual leaders in cultural fields have expressed a desire for their work to be taken at face value, e.g., David Lynch, film-maker: "I refuse to give explanations of any film I make"; Andy Warhol, artist: "If you want to know all about Andy Warhol, just look at the surface: of my paintings and films and me, and there I am. There's nothing behind it"; and David Lang, composer: "I don't want people to hear the process . . . I don't want people to know about the mechanics of [my process]".

We argue that in modern culture, a curious thing can happen when artists attempt to remove all reference to themselves and their process from discussions about the artistic (and commercial) value of their work. That is, in the absence of such information, people may tend to fill in the gaps about personality and process, and may do so in ways which bolster the credibility of an artist and increase the perceived value of his/her works. Indeed, one could argue that – in the same way that artists invite people to interpret the images/imagery in artworks in their own way by not prescribing what people should see/read/hear – when they refuse to provide such meta-level details, artists and writers are actually (purposefully or not) inviting art lovers to invent back-stories about process and personality themselves. Given how highly creativity and the arts are valued in society, such inventions tend to be favourable and can be romantic, e.g., we project the romantic view of an artist working in a Parisian garret, selling paintings for meals, etc. It is not true that we know nothing about an anonymous artist or his/her practice. In fact, we know thousands of things, we invent more to fit the context, and we romanticise about their life. Above all, we project creativity onto them as a default. Such projections are very much part of art appreciation and should be cherished. They enable us to have a dialogue with the artist and their artwork, whether they are there in person or not.

In such a context of non-disclosure, the comparison of the situation for computer-generated artefacts with the situation for human-produced artefacts is not particularly favourable. The vast majority of people have little or no idea about programming or programs, and may even harbour a desire not to find out about these things. Thus, when invited to assess a painting or poem, say, without background knowledge, they are denied any opportunity to invent a back-story, cannot project personality traits or romantic situations onto the computer, and cannot enter into any dialogues. More importantly, this situation can lead to people realising how much they value the human connection, whether actual or imagined, in such situations.

In the context of creative writing, we have previously stated that:

> Mainstream poetry is a particularly human endeavour: written by people, to be read by people, and often about people. Therefore – while there are some exceptions – audiences expect the opportunity to connect on an intellectual and/or emotional level with a person, which is often the author. [19]

Generalising from this individual case, there is, we posit, a *humanity gap* to be faced by Computational Creativity researchers who truly want their software to enhance society by being creative for artistic and utilitarian purposes. Unfortunately, as a community, our own scientific practice can exacerbate the situation. This is because researchers regularly undertake Turing-style comparison tests, where computer and human-produced artworks are mixed up and their origins are hidden to the participants in the experiment. Often, the participants have a technological background of some sort, but equally often have little such experience. The situation is not currently bad, but as more members of the general public are exposed to computer generated artefacts with zero additional information, the more things could get worse, as people realise that the lack of context and opportunity for dialogue is inherent (as things currently stand) when computers participate in cultural activities.

The situation is worse: authoritative people in cultural domains often have similarly little technological background and do hear of experiments where computer generated artefacts are presented without context. It is not unheard of for such authorities to make proclamations about the uncreative nature of software and the lack of potential for this in their domain. Such proclamations can often be declarative speech acts in the sense of Searle above: they can *make software uncreative* by being uttered, and there is usually little that Computational Creativity researchers can do about this in the short term.

Of course, situations with non-disclosure are intended to reduce variables so that a scientific study of the value of computer generated artefacts can be undertaken. One could argue that these contexts are intended to help people realise how much they value the aesthetic appeal of art, literature and music, regardless of other factors. In fact, they can often help people realise how little they can relate to the computational origin of such work. In [20], we raise other issues with Turing-style comparison studies, such as encouraging naïvety and pastiche generation in creative software. We can go further, by stating that this methodology might be a case where scientific practice has impeded cultural progress: we raise the question of whether Turing-style comparison tests, and the presentation of computer-generated artefacts with no supporting information in general, while valuable for short-term scientific progress, is actually detrimental to the long-term goal of embedding creative software in society.

4 Addressing the Situation

In Searle's Chinese Room thought experiment [21], discussion involves whether a computer can be said to have a mind or consciousness in the same sense that people do. Questioning whether a computer in the Chinese Room can be perceived as creative in the same way that people are, has more saliency for Computational Creativity research. Computers aren't human, and are unlikely to be even remotely similar in the near future. Hence, like most AI researchers, we personally are not interested in whether software will one day be perceived as being creative in the same sense that people perceive other humans to be creative (although there are researchers in the community for whom simulation of human-like creativity is the aim). We argue in [22, 15] that people take into account how a person or software operates when they assess the value of the output. Hence, unlike most AI researchers, we have to address the humanity gap described above, and to do so, our software cannot just be artificially intelligent, it has to *be seen to be AI*. We address ways in which this can be achieved in the first subsection below. In the second subsection, we introduce and justify the notion that computer-generated artefacts should be seen as fundamentally different objects to human-produced ones, and we discuss the positive ramifications of this.

4.1 Software Accounting for its Actions

One aspect of the Chinese Room experiment is the lack of information coming from the room about the processes taking place. It may be unlikely, but it is possible that people's perception of a mind in the computer would be increased if the text translation process were explained somewhat, especially if that process involved decision making, the production of new information, the invention of new rules, and so on. Similarly, the perception of creativity in software may be enhanced by the software explaining what it has done and why, and describing the artefacts it has produced. We argue for software to be accountable in this way by *framing* its work, as discussed in [22]. Framing is a term borrowed from the visual arts, referring not just to the physical framing of a picture to best present it, but also giving the piece a title, writing wall text, penning essays and generally discussing the piece in a context designed to increase its value.

We advocate a development path that should be followed when building creative software: (i) the software is given the ability to provide additional, meta-level, information about its process and output, e.g., giving a painting or poem a title (ii) the software is given the ability to write commentaries about its process and its products (iii) the software is given the ability to write stories – which may involve fictions – about its processes and products, and (iv) the software is given the ability to engage in dialogues with people about what it has produced, how and why. This mirrors, to some extent, Turing's original proposal for an intelligence test [23]. As an example, as described in [19], we demonstrated a poetry generation system which is able to provide commentaries about its poetry, and how and why it produced a particular poem. In the case studies below, we present generative projects with much emphasis on such management of observers.

In a human context, such accountability might seem counterproductive to the aim of achieving a perception of creativity in the minds of observers. People cherish the unexpected in the arts and sciences, both in the surprising nature of the finished results, and in the process: people who take advantage of chance encounters and serendipitous situations have always been celebrated in society. As we discuss in [24], in a computational setting, there are advantages to software being immersed in environments where serendipity might occur. Accounting for lucky events that trigger creative acts might lessen the celebration and hence the impact that the acts have. People tend not to describe their processes and products in such an explicit way as we advocate for software. Indeed, as proposed above, keeping quiet about such matters can be beneficial to human creators. Such silence can preserve the mystery surrounding creative acts and creative people, and can invite speculation and romanticisation, which adds value. Explanation of the process can lead to demystifying it, even making the creative acts seem ordinary and uninspiring.

In this light, it might seem like a bad idea to get software to account for itself. However, software is not human, and societal acceptance of the creative potential of software is much lower than it is for, say, small children. Given the largely universal encroachment of software into society and the exposure over many years that most people have had, while it is expedient to think of software in human terms, it is possible that people will increasingly think of software in its own rights. In our experience of observers of creative systems, software is not viewed as a mini-human (with 'mini' clarified as less intelligent, less human, less interesting perhaps), but rather as a different intelligent entity in the world. At the very least, people know that software isn't human, and they account for this when consuming the artefacts that software produces. People are aware of the humanity gap.

We believe that, at this stage in the history of computationally creative systems, it is more important to address the humanity gap than worrying about maintaining mystery in the creative process, and that getting software to frame its work is a good proposal to begin with. Framing serves to highlight that intelligent processing was used to produce artefacts, which is an important first step. To illustrate this, as mentioned in [19], there is an automated poetry generation system available at a website[4] which states that: "A great deal of poetry mystifies its readers: It may sound pretty, but it leaves you wondering 'what the hell was that supposed to mean?"' Justified by this observation, the software available at this website randomly produces poems with little or no intelligent processing, which look like they may have been written by a person as a particularly difficult to interpret piece.

Of course, people have an amazing capacity to find meaning in texts written with no communicative purpose, and this can be entertaining and even enlightening, regardless of where the text came from. However, this misses the point that when we decode the difficult poetry of an intelligent person, as is was constructed intelligently – perhaps purposefully to be abstruse and obfuscated – there is at least some chance we will hit upon what the poet had in mind with their communicative act. There is, therefore, a human connection and comprehension purpose to decoding the poetry, in addition to the benefits of interpretation and reflection. This is not true of poems constructed without intelligence, and when software does this, it highlights again the humanity gap. Fortunately, there have been many intelligent automated poetry generators developed. However, with the exception of the system described in [19], these do not account for their process or product. In such circumstances, we have to hope that poetry lovers will also read the technical papers describing the system, or that human-written contextual material is provided and read (which is unlikely).

Randomness in the above online poetry example was probably achieved with a random number generator. It is our contention that unexpectedness achieved through the usage of such random number generations is generally to be avoided in creative systems. Random number generators are not easily explained, and any explanation of how they work is likely to be dull and raise distracting issues about whether it acts truly randomly (or as random as a person could be). Moreover, in our experience, any unexpected behaviour or output by software is cherished by observers. However, the bottom drops out of this experience when they realise that – rather than some inspired choices – a random number generator was largely responsible for the novelty. The dialogue, if it had started at all, stops abruptly.

Unpredictable behaviour in software and the production of novel artefacts does not have to rely on random number generators. Instead, it can be achieved via the handing over of creative responsibility to software, which inevitably raises the complexity of its processing, and this can reach a level where predicting in advance what it will produce is impossible. This can be coupled with a search space large enough to contain results that will surprise people, and one way to achieve that is through the downloading of materials from dynamically changing web sites (such as social media and news outlets), as demonstrated in [25, 26, 27]. As in the case studies below, for many people, the fact that the software has accounted for its actions will not detract from the value of the artefact, but will actually add to the experience through increased dialogue.

4.2 The Fundamental Nature of Artefacts

In advocating full disclosure of the computational origin of artefacts via the software accounting for itself, we expect that people will consume computer generated artefacts knowing what they are getting, and what they are not getting: a human connection with the creator. One of the main results of the Computational Creativity odyssey

[4] www.cutnmix.com/robopoem

could, therefore, be a realisation that we don't want or need computer poets, musicians or painters, and our research has served to highlight how much we value creative people, that software should know its place as a mere tool, and that creativity and humanity are too closely linked for a meaningful separation. This would, we believe, be a wasted opportunity, as we have seen many times how computational creators can enhance our culture. One way to mitigate this waste would be to accept that what software does and what people do are always going to be different, and hence we should stop using words such as 'creativity' to describe algorithmic processing, as this makes little sense in a human-free context. With this option, we would rename the field, introduce new terminology, and drop the aim of enhancing the study of creativity through computation. Given the history of the field, that many people in the field are interested in the simulation of human-like behaviours, and that the majority of creative systems have been modelled at some level on how people create particular artefacts, this option seems unlikely to gain a footing.

Another reason to not separate computer and human creativity too far is that the artefacts produced are seemingly very similar. Indeed, Turing-style tests have shown that, under the right conditions in certain domains, people cannot tell the difference between computer and human produced poems, painting, compositions, etc. At first glance, we could conclude that poem-shaped collections of words, for example, that are produced by machine, should simply be taken as poems without further discussion. In well received work on evaluating creative software via its output [28], Ritchie advocates that we first consider whether output from software is *typical*, i.e., the artefacts are of the right form for the context. The assumption after this stage is that we should refer to any typical objects as a poem, sonata, painting, etc., and treat them as if they were produced by a person. However, it is worth challenging whether computer-generated artefacts are indeed fundamentally the same as human-produced counterparts.

Tear drops and acid drops look exactly the same, but are fundamentally different. At a chemical level, the difference is structural: the arrangement of atoms into molecules. However, at another level of abstraction, tears and acid drops are really constituted of exactly the same things: protons, neutrons and electrons. The difference is not, therefore, in terms of what they contain or what they look like. This abstraction is, of course, a slight of hand, but it does highlight a more important fundamental difference between tear drops and acid drops: the effect they have on people at a physical and an emotional level. Acid drops burn flesh and raise anger, tear drops don't sting, but are equally potent at raising sadness, compassion or joy.

An analogy between the comparison of teardrops and acid drops and the comparison of computer-generated and human-produced artefacts presents itself. While, say, a computer generated poem may contain exactly the same letters in exactly the same order as one penned by a person, we should consider the two as fundamentally different to each other because of their potential effect on readers. It may be disturbing, but we must accept that when placed in a mind through reading and interpretation, a computer generated poem will not have the same effect as one penned by a poet. This is true of poems by children, but here, as all readers were once children, and many have children, a different, valuable, connection can be made. It seems more appropriate to think of including juvenilia in the poetry canon than computer generated poems. Computer generated artefacts are fundamentally different objects in the world to their human produced counterparts, even if there is a strong surface level similarity. If we accept this, then there can be benefits both for Computational Creativity researchers, and the consumers of computer creations.

On a practical level, we can embrace the lack of humanity in computers, and start thinking about how to fill the gap. One particular instantiation of this would be to differ the form of computer generated artefacts from their human counterparts. For instance, for all the reasons given above, it seems perfectly sensible to think of a computer generated poem as a doublet containing a commentary and a poem-shaped piece of text, as we did in [19]. As another example described in the first case study below, we re-imagined a portrait as a triplet constituting a computer generated portrait, a commentary and an experience sat in front of a laptop computer. There may also be benefits in better understanding our field if we accept the fundamental differences between automatically generated and hand produced artefacts. *Strong* and *weak computational creativity* mirrors so-called strong and weak Artificial Intelligence [29]. In the former, the focus is far more on the simulation of creativity than the generation of artefacts of value. In the latter, the emphasis is far more on the production of artefacts of real value to society than any perception of creativity. Many of us engage in projects that straddle these two outlooks. In the weak sense, getting software to produce artefacts as close to those produced by people as possible (in terms of their look *and* their effect on people) is paramount. In the strong sense, however, it would be perfectly acceptable to re-imagine artefact composition in a computational setting, if it allowed a scientific study of whether people perceive software as creative or not.

5 Observer Management in Automated Portraiture

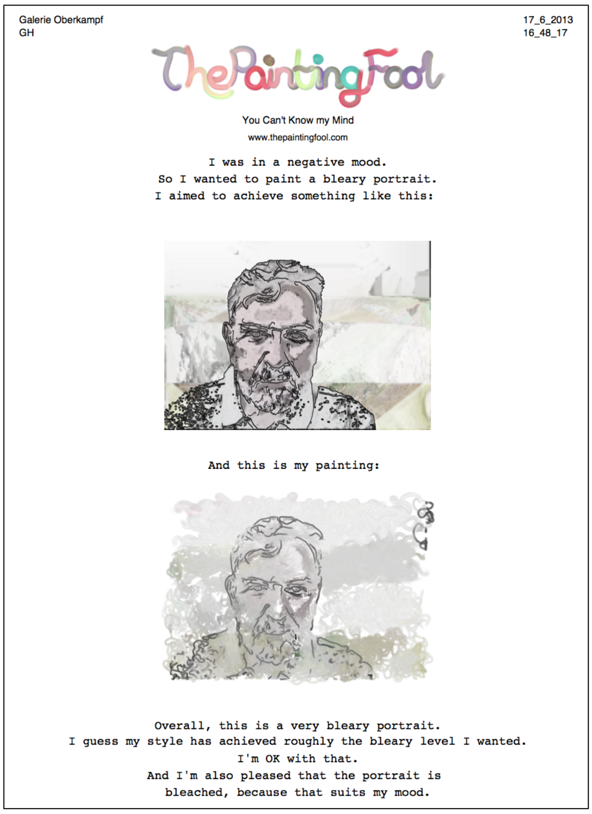
As part of an exhibition with The Painting Fool[5] system [30] in 2013, we enabled the software to produce portraits for people live in a gallery. Managing the expectations and perceptions of the observers, who in many cases were also the sitters for the portraits, was a key aspect of this project, which was done both by us and by The Painting Fool. In particular, we hung posters describing the behaviour of the software as exhibiting aspects of intentionality, imagination, skill, appreciation, reflection and learning. Moreover, the software's actions and output were tailored to support the perception of these behaviours and an impression of creativity in the software by observers present in the exhibition, especially those sitting for a portrait.

Portraits were painted with people sitting in front of a laptop. It was immediately made clear that (i) the software was modelling a 'mood' to direct its painting, and (ii) the sitter was very much a tool for the software, not the other way around. This was achieved by opening remarks from the software of the form: "Thank you for being my model. I'm in a negative mood right now, so I would like you to express a sad emotion." This was followed by The Painting Fool explicitly directing the sitter, while video recording them. A still image was then extracted where the sitter was expressing an emotion. Machine vision techniques were applied to remove the background, into which was substituted one of 1,000 abstract art images, to which one of 1,000 image filters was applied. The same filter was applied to the face of the sitter placed in the foreground, producing in a few seconds an image conception, or sketch for the portrait, such as the first image of figure 1. Following this, a canvas appeared on screen, and a hand holding either a pencil, paint brush or pastel made virtual marks on the canvas leading to a non-photorealistic rendering of the background and foreground of the portrait, taking between 2 and 10 minutes, depending on the style. An example portrait is given at the bottom of figure 1, which was printed and given to the sitter, along with the commentary (the whole of figure 1).

The title of the exhibition was: *You Can't Know my Mind*.[6] This

[5] Online presence: www.thepaintingfool.com

[6] Gallery pages: www.tinyurl.com/yckmm



Galerie Oberkampf
GH

17_6_2013
16_48_17

ThePaintingFool

You Can't Know my Mind
www.thepaintingfool.com

I was in a negative mood.
So I wanted to paint a bleary portrait.
I aimed to achieve something like this:

And this is my painting:

Overall, this is a very bleary portrait.
I guess my style has achieved roughly the bleary level I wanted.
I'm OK with that.
And I'm also pleased that the portrait is
bleached, because that suits my mood.

Figure 1. Example commentary by The Painting Fool, from the *You Can't Know my Mind* exhibition, Paris, June 2013.

was carefully chosen to present to the observers the idea that people in Computational Creativity research are writing software to be beyond their control, in order to increase surprise and (perception of) creativity. The mood model implemented in The Painting Fool was largely responsible for the unpredictability of its actions. For this, the software continually read articles in a large corpus from the Guardian newspaper. An initial article was chosen arbitrarily but not randomly (based on time of day) when the software was initiated each morning. Key-phrases from this article were extracted and used to find other articles of a similar nature. On inspection, we noted that, while the nature of the next article was predictable to some extent, what the software would be reading after ten minutes was impossible to predict, e.g., it would start reading about the war in Syria, and after a short while it would be reading a piece about football. Each article was assessed using sentiment analysis to determine if it was a high/medium/low positive or high/medium/low negative piece. An average over the sentiment of the previous 10 articles was calculated and used to direct the software into one of six 'moods'.

In this manner, The Painting Fool could always account for its mood by describing what it had read and including some particularly pointed key-phrases. As it was impossible to predict what mood it would be in for a particular portrait (as mentioned above), we call the kind of algorithm used here *accountably unpredictable*. The mood changed the way in which the software attempted to produce a portrait: via the mood-inspired choice of an adjective such as 'bleary' in figure 1, decisions about the background, image filter, rendering method and post-hoc visual analysis were made (details omitted). Also, by using a two stage process where a conception and a painterly rendering of it were produced, and exposing this in the commentary provided, the software expressed a level of intentionality with which observers could relate. To highlight this, using a neural network machine vision approach inherited from the DARCI system [31] (details also omitted), the software determined whether it had achieved its aim or not, and included that in the commentary, as per figure 1.

The purpose of the exhibition was cultural, not scientific, and no experimentation was undertaken. From our experience, however, we contend that the behaviours exhibited by the software and explained in poster form enabled people to be surprised by the resulting portrait (and many of the 100 or so sitters in the exhibition were very surprised), while still projecting creativity onto the software. This upheld the aim of the *You Can't Know my Mind* exhibition: as it used some intelligence, and could explain its actions, it was somewhat appropriate to employ the word 'mind' with reference to The Painting Fool. However, as the process was unpredictable, it was impossible to know this mind, and people realised that some software is written not to be a tool but to be a creative individual. In fact, when in the most negative of moods, for roughly one in every 6 portrait attempts, The Painting Fool refused to paint a portrait and sent the (often shocked) sitter away, citing a particularly depressing key-phrase in a particularly distressing article. In these cases, it pointed out explicitly: "No random numbers were used in coming to this decision".

6 Players of Automatically Designed Videogames

A *game jam* is a contest where entrants attempt to make a videogame from scratch in a short period of time, normally with the added restriction of a theme which developers must incorporate into their game somehow. Ludum Dare is one of the largest regularly occurring game jams in the game development community, taking place three times a year and garnering over 2,000 entries in December 2013, where developers were given the theme 'You Only Get One'. ANGELINA is an automated videogame designer developed to investigate issues surrounding Computational Creativity in a ludic and interactive context. Many different versions of ANGELINA have been developed, working with various different kinds of game, technologies and user guidance. The most recent iteration, *ANGELINA-5*, was designed to enter game jams, by allowing it to be given just a theme in plain text as a starting point. This theme is then interpreted by ANGELINA-5 and used to influence the design of the game.

ANGELINA-5 entered Ludum Dare in December 2013. One of the objectives was to investigate the reactions of various groups of people to a piece of computationally creative software entering such a contest. To gain more insight into these groups, we entered two games designed by ANGELINA-5 to Ludum Dare. In the first submission,[7] we included a commentary generated by ANGELINA-5 to illustrate the actions of the system, as well as multiple paragraphs describing the research behind ANGELINA-5 and identifying the game as the creation of a piece of software. In the second submission,[8] we edited ANGELINA-5's commentary to remove references to it being software-based, edited it for grammar, and added no supplementary explanation about the software, the origin of the game, or anything to connect the game with a digital author. The ratings process for Ludum Dare takes place in the 22 days following the contest, and is conducted as a peer review system, where each entrant is asked to rate and review games by other entrants. Ratings are given as marks out of five for eight categories: Audio, Graphics, Mood, Theme, Humour, Fun, Innovation and Overall.

The results for the two entries by ANGELINA-5 can be seen in table 1. While we were unable to get specific vote data, we do know

[7] *To That Sect* game: `www.tinyurl.com/tothatsect`

[8] *Stretch Bouquet Point* game: `www.tinyurl.com/stretchpoint`

	To That Sect	Stretch Bouquet Point
Overall	500	551
Fun	515	543
Audio	211	444
Graphics	441	520
Mood	180	479
Innovation	282	525
Theme	533	545
Humour	403	318

Table 1. Placings for ANGELINA-5's two games entered into Ludum Dare 28. There were 780 submissions to this track.

that 70 people rated *To That Sect*, the non-anonymised submission, while 26 people rated *Stretch Bouquet Point*.[9] While it is impossible to calculate the confidence of these ratings without the vote data, we can see that they differ by hundreds of positions for some categories such as Mood and Audio. We can also see a noticeable difference in the comments left by some of the reviewers underneath both submissions, in terms of their tone and attitude when dealing with each game. Many commentators indirectly criticise anonymised games, such as "You made me feel something there. Don't make me put it into words though". Other commentators made more obvious statements of criticism or praise, such as "This was a rather annoying experience" or "This game feels dreamy. The audio is intense." Only one comment included both praise and criticism. We attribute the indirect or sarcastic comments to an unwillingness to potentially criticise a human for performing poorly, even though other reviewers were less tactful. Ludum Dare is often used as a learning experience for amateur developers, and many children enter using simple game creation tools. We believe many reviewers felt uncomfortable with direct criticism for this reason.

By contrast, comments on *To That Sect* were more balanced in nature, often offering both praise and criticism in equal amounts, e.g., "Angelina seems really good at creating an atmosphere with both sound and visuals. But the game part of it seems a bit lacking still". In the description of the game, we asked people to rate it as they would any other Ludum Dare entry, hoping to dissuade people from reviewing the *concept* of ANGELINA-5 rather than the game itself. Nevertheless, many reviewers suggest that their scores were influenced by their appraisal of ANGELINA-5 as a novel system, rather than what it was capable of creating, e.g., "creating a program to create your game . . . [is] certainly not something you see every day. On that front alone, this gets a lot of points for innovation". These results suggest that reviewers were unable to separate the creator from the artefact, and were incapable of reviewing the game as if created by a person. For instance, *To That Sect* rated 282nd of 780 for Innovation. These ratings are subjective, and it is hard for us to objectively assess them. However, we do not believe there is anything particularly innovative about *To That Sect*. As such, we must attribute this high ranking to reviewers assessing the game as a product of ANGELINA-5. It seems that reviewers projected (human) innovation in the ANGELINA project onto the game it produced.

We can also examine reactions to particular elements of ANGELINA-5's work and compare it to critiques of similar games. One comment on *To That Sect* states "If it [had] added shooting at the statues that you must avoid and a goal how much ships you have to collect, it would have been better. It felt like playing [an] 'art-message' type of game". *LITH*[10] is a game entered into the competition by a human designer, where the player navigates a maze and collect bags of gold coins, while avoiding patrolling robots. They can escape to an exit at any stage, with their score being the amount of gold collected. While not an exact duplicate, the rules of *LITH* bear much resemblance to those of *To That Sect*, i.e., search for as many objects of a certain type as possible, while avoiding another object, then exit. *LITH* was entered in the same track as ANGELINA-5's games, and ranked 95th Overall, 125th for Fun, and 274th for Theme. None of the comments on *LITH* reference the game's rulesets in a critical way. Notably, *LITH* ranks 259 places above To That Sect for Theme. This is significant, as the *LITH* designer justifies its theme in a fairly thin way, by saying simply that the player only has one opportunity to save their score (which they do by ending the game, as in To That Sect). The games are by no means identical: *LITH*'s level is more closed in to accentuate a feeling of claustrophobia, but the similarities are many. This analysis suggests a fundamental difference in how people evaluate a game when they have knowledge and when they have no knowledge of its designer and design process.

7 Related Work

Searle's Chinese Room argument is useful for framing many of the controversies in Computational Creativity. The discrimination test between Chinese-speaking Hao and the English-speaking Searle is directly analogous to discrimination tests between computer and human produced artefacts which are often performed in Computational Creativity. Here, we agree with Searle's focus on *process* over *behaviour*: identical input/output pairs can merit attribution of different properties depending on the processes. This amounts to an argument against such discrimination tests as a way of evaluating machine creativity. Whether computers will ever be seen as "really creative" by society will depend on changing notions of creativity and Computational Creativity developers and their systems managing the public *perception* of creativity in their software, in the same way as human artists manage public *perception* of their own creativity. Here, Searle's distinction between intrinsic and observer-relative properties can help us to understand the rarely discussed cultural and social aspects of the field. We hold that intrinsic properties of a poem generated by a creative system include such aspects as length and metre, while the property of "being creative" is observer-relative. Searle points out that observer-relative social facts depend on human institutions for their existence, thus pointing to sociology and the study of relationships between technological innovation and scientific research and social, political and cultural values.

Social perspectives on the perception of creativity provide an essential counterpart to the traditional AI focus and will be necessary if Computational Creativity is to become an accepted part of mainstream society. The sociology of scientific knowledge can provide relevant ideas here. For instance, via his *Actor Network Theory* [32], Latour holds that in order to understand processes of innovation and knowledge-creation in science and technology, we need to study the relationship between actors, including material objects and diverse social groups. In the case of Computational Creativity, relevant groups include researchers, the wider AI community, funding bodies, experts in the psychology of human creativity, artists, art critics, philosophers and so on. Each group has accompanying visions, beliefs and goals, in which they have, to a varying degree, invested (and which, to a varying degree, define them as a group). Understanding such different perspectives and their interactions is essential if output from creative software is ever to be deemed creative by mainstream consumers of cultural artefacts.

[9] This is due in part to ANGELINA-5's small following on the internet, which promoted the non-anonymised submission more than normal.

[10] *LITH* game: `www.tinyurl.com/lith-ludum`

8 Conclusions

A piece of software could create very similar looking pieces to a person, or could produce pieces of startling novelty and beauty. The same piece of software could produce work in very human-like ways, and enter into dialogues with people about its process, its products and the cultural contexts of the day. But the bare fact is that, at this stage in humanity's technological development, people will not look on the artefacts produced in the same way as they would those produced by people, nor would they celebrate the creativity in the machine as they would in a human being. This is not a problem – the only issue for Computational Creativity researchers here is thinking that this is a problem. And nor is this a moot point. We are at the beginning of an time where computer creativity will go from being a novelty to being commonplace, and addressing the kinds of observer issues discussed here will be very valuable in helping this transition.

As a recent controversial example, online retailer Amazon recently briefly sold T-shirts with slogans such as *"Keep Calm and Rape a Lot"* [33]. The T-shirt company responsible posted an apology on its website, and insisted that the offending articles were "automatically generated using a scripted computer process running against hundreds of thousands of dictionary words". This may be the first example of computer generated artefacts causing such offence and a company – while taking responsibility – blaming generative software for poor quality artefacts, while tacitly acknowledging that the software had taken on unsupervised creative responsibilities in their workplace. The slogan-generating script was unlikely to be exhibiting particularly intelligent behaviour if it couldn't avoid using words like rape – and this highlights the need for AI researchers and practitioners to engage in more Computational Creativity projects. Engineering into software higher intelligence, more innovative skills and greater accountability, so it produces more interesting artefacts will, we hope, lead to greater acceptance of creative systems in society.

Ultimately, we expect to get to the situation where computational creators need not produce commentaries and other material, as their intelligence and creativity is not in doubt – the artefacts they produce will speak for themselves. However, to get to this stage, we believe that at least the following things are necessary. Firstly, software should provide more information about the process and contexts behind its creations; software should provide commentaries and stories, and enter into dialogues. Secondly, we should re-imagine artefacts such as poems as digitally-created counterparts, as this will clarify the differences in emotional connection between human-authored and computer-generated artefacts. If we can get used to the idea of books and e-books being similar but different, in order to manage expectations of physicality, then surely we can adopt the idea of poems and automatically produced poems (*a-poems*, say) being fundamentally different, in order to manage expectations of humanity.

Acknowledgements

Many thanks to the anonymous reviewers for their helpful comments, and to Dan Ventura for his collaboration on the *You Can't Know my Mind* software and exhibition, and for discussions about accountable unpredictability. This work was funded by EPSRC grants EP/J004049 and EP/L00206X, and EC funding for project WHIM (611560) supported by the FP7 ICT theme, and the FET program.

REFERENCES

[1] S Colton and G Wiggins, 'Computational Creativity: The final frontier?', in *Proceedings of the 20th ECAI*, 2012.

[2] A Eigenfeldt, A Burnett, and P Pasquier, 'Evaluating musical metacreation in a live performance context', in *Proceedings of the 3rd Int. Conf. on Computational Creativity*, 2012.

[3] D Moffat and M Kelly, 'An investigation into people's bias against computational creativity in music composition', in *Proceedings of the Third Joint Workshop on Computational Creativity*, 2006.

[4] J Locke, *An Essay Concerning Human Understanding*, OUP, 1975.

[5] D Dennett, 'Three Kinds of Intentional Psychology', in *D Dennett, The Intentional Stance*, 43–68, MIT Press, 1987.

[6] W Gallie, 'Essentially contested concepts', *Proceedings of the Aristotelian Society*, **56**, 1956.

[7] J Gray, 'On the contestability of social and political concepts', *Political Theory*, **5(3)**, 1977.

[8] K Smith, 'Mutually contested concepts and their standard general use', *Journal of Classical Sociology*, **2(3)**, 2002.

[9] J A Plucker, and M C Makel, 'Assessment of creativity', in *The Cambridge Handbook of Creativity*, Cambridge University Press, J C Kaufman, and R J Sternberg (Eds), 2010.

[10] J Jones, 'Santa bought me a Playstation. But it's still not art', *Guardian*, 7th January 2014.

[11] K Stuart, 'Video games and art: why does the media get it so wrong?', *Guardian*, 8th January 2014.

[12] A Jordanous, *Evaluating Computational Creativity*, Ph.D. dissertation, Department of Informatics, University of Sussex, 2012.

[13] J Austin, *How to do Things with Words: The William James Lectures delivered at Harvard University in 1955*, Clarendon Press, 1965.

[14] J Searle, 'A taxonomy of illocutionary acts', in *Language, Mind and Knowledge, Vol. 7*, ed., K Gunderson, 1975.

[15] S Colton, 'Creativity versus the perception of creativity in computational systems', in *Proceedings of the AAAI Spring Symposium on Creative Systems*, 2008.

[16] W Wimsatt and M Beardsley, *The Verbal Icon: Studies in the Meaning of Poetry*, University of Kentucky Press, 1954.

[17] L Lambourne, *The Aesthetic Movement*, Phaidon Press, 1996.

[18] *The Stuckists*, ed., K Evans, Victoria Press, 2000.

[19] S Colton, J Goodwin, and T Veale, 'Full-FACE poetry generation', in *Proceedings of the 3rd Int. Conf. on Computational Creativity*, 2012.

[20] A Pease and S Colton, 'On impact and evaluation in Computational Creativity: A discussion of the Turing test and an alternative proposal', in *Proceedings of the AISB symposium on AI and Philosophy*, 2012.

[21] J Searle, 'Minds, brains and programs', *Behavioural and Brain Sciences*, **3(3)**, 1980.

[22] J Charnley, A Pease, and S Colton, 'On the notion of framing in computational creativity', in *Proceedings of the Third International Conference on Computational Creativity*, 2012.

[23] A Turing, 'Computing machinery and intelligence', *Mind*, **59(236)**, 1950.

[24] A Pease, S Colton, R Ramezani, J Charnley, and K Reed, 'A discussion on serendipity in creative systems', in *Proceedings of the Fourth International Conference on Computational Creativity*, 2013.

[25] M Cook and S Colton, 'Automated collage generation – with more intent', in *Proc. of the 2nd Int. Conf. on Computational Creativity*, 2011.

[26] M Cook, S Colton, and A Pease, 'Aesthetic considerations for automated platformer design', in *Proc. of the 8th Annual AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment*, 2012.

[27] A Krzeczkowska, J El-Hage, S Colton, and S Clark, 'Automated collage generation – with intent', in *Proceedings of the First International Conference on Computational Creativity*, 2010.

[28] G Ritchie, 'Some empirical criteria for attributing creativity to a computer program', *Minds and Machines*, **17**, 2007.

[29] J Searle, *Mind, language and society*, Basic Books, 1999.

[30] S Colton, 'The Painting Fool: Stories from building an automated painter', in *Computers and Creativity*, eds., J McCormack and M d'Inverno. Springer, 2012.

[31] D Norton, D Heath, and D Ventura, 'Finding creativity in an artificial artist', *Journal of Creative Behavior*, **47(2)**, 2013.

[32] B Latour, *Science in Action: How to Follow Scientists and Engineers through Society*, Harvard University Press, 1987.

[33] T McVeigh, 'Amazon acts to halt sales of 'keep calm and rape' t-shirts', *Guardian*, 2nd March, 2013.