DRS 2018 Design Research Society 2018 Catalyst

University of Limerick
25th–28th June 2018

An Ontology of Computational Tools for Design Activities

MOTHERSILL Philippa* and BOVE V. Michael

MIT Media Lab
* Corresponding author e-mail: pip@mit.edu
doi: 10.21606/dma.2018.456

While able to automatically generate and optimise designs for variables provided by a designer, today's computational design tools do not specialise in the earlier, more tacit tasks such as gathering and sorting disparate information or generating hypotheses and identifying novel directions. This paper presents a review of computational technologies that could potentially play a role in these early stage design activities. Using a framework that deconstructs design activities into underlying tasks, an ontology that reviews the various computational tools that could be applied in these activities was created. Computational technologies such as neural networks and stochastic algorithms were found to provide features that could potentially allow for discovering and linking new information together in order to provoke the – often unexpected – inspiration that can guide designs in the latter phases of development.

computational design tools; creativity support; early design process

1 Introduction

Since the mid-twentieth century, computation has become increasingly intertwined with design, from abstracting the craft of the design process into models that use a more algorithmic logic (Alexander, 1966), to the development of automated Computer Aided Design (CAD) software that explores and optimises the range of different values a set of design variables could have (Papanikolaou, 2012). The paradigms of computation used in the design process have changed dramatically throughout the development of CAD technologies. Early tools such as Pro/ENGINEER allowed engineers to set clear parameters and relationships between a database of features, requiring designers to explicitly plan and describe their 'design intent'. In comparison, newer direct modeling CAD systems such as Autodesk Fusion 360 allow forms to be 'sculpted', enabling designers to integrate more of their implicit intuition into their creations (Tornincasa & Di Monaco, 2010).

Despite these advances, CAD tools are still more applicable to the latter, rational stages of the design process and less useful early on, where intuition is used to re-interpret a design situation, build analogies and look for emergent ideas (Bernal, Haymaker & Eastman, 2015). Emerging today are advanced computation techniques that could contribute to some of these more human-centered problem solving activities; design solutions can be generated and optimised to a set of input

variables using genetic algorithms, and evaluative systems can derive inferences and insights from data using statistical models (Sjoberg, Beorkrem & Ellinger, 2017).

How far is it possible for these new computational techniques to play a role in the tools used in the design process? This paper will review a range of computing technologies and suggest how they might relate to the design process now and in the future. Establishing the opportunities and challenges of integrating computational tools into the early stages of the design process, we describe an approach through which to identify potential computational technologies relevant to design activities. We then offer an ontology of computational tools for design, considering the capabilities of these tools by reviewing several case studies that offer new technical approaches as part of their creative process. Highlighting a few of these computational tools, we conclude by considering how they might be applied to tools fitting the early phases of the creative process.

2 Computing the Design Process

Computation—both as an epistemological framework and a digital technology—can be a powerful tool. Despite the increasing use of computational tools in design, their limitations at engaging with tacit knowledge and abstract definitions mean they are only sparsely used in the early stages of the creative process. This section reviews the various phases in the design process to consider what types of activities are carried out in the early stages, and presents an approach to deconstruct these less rational activities to understand if it is possible to map new computational technologies to them.

2.1 A brief review of the literature

There has been much research into defining the many phases and activities of the design process; Dubberly (2004) collected a staggering 88 of them. This is partly due to the fact that design can have many meanings; no longer just focusing on the aesthetics of an industrially produced artifact (McCullough, 1998), the design methods movement expanded the definition of the design process to include the activities of design research and idea generation (Michel, 2007).

However, as this plethora of different approaches shows and as commented on by Wynn and Clarkson (2005): “there is no single model which is agreed to provide a satisfactory description of the design process [and] no ‘silver bullet’ method which can be universally applied to achieve process improvement.” Despite this lack of agreement, the many attempts to review and synthesise the different models into an overarching taxonomy (Mendel, 2012; Wynn & Clarkson, 2005; Design Council, 2007) generally divide the overall design process into four phases—discover, reframe/define, envision/develop, and create/deliver—that are often concurrent and cyclical (Lawson, 2006; Schön, 1983; Blessing, 1994). In the discovery phase, designers build on initial hunches to collect diverse information and intuitively structure the often disparate data to reveal patterns and gather insights. In the reframe/define phase, designers use their imagination to juxtapose the information in non-obvious ways to “reveal new salience, relationships, and meanings” (Mendel, 2012). These opportunity areas are the focal points for envisioning new designs, i.e. the creative brief to guide the next phases. Potential solutions or concepts are generated and evaluated in the next envision/develop phases, converging from many extreme envisionings to a few more concrete forms and final solutions in the final create/deliver phase.

Throughout these phases, designers change from considering concrete information to more abstract interpretations then back (Fulton Suri, 2008). Especially in the early phases that focus on design research and idea generation, designers bridge “the space in-between research and concept” (Robinson in Dubberly & Evenson, 2008). Moving between analysis and synthesis, designers use abductive reasoning to translate models about what the current situation is into a preferred future of ‘what could be’ through creating and playing with abstract concepts (Steinfeld, 2017).

This focus on abstract interpretations may explain why computational tools are rarely used by designers in the early phases. Taking Gero’s (1990) definition that design “can be modeled using variables and decisions made about what values should be taken by these variables”, we suggest

that it is in these first two phases—where intuition and playful exploration guide the creative leaps that synthesise information in new ways and "liberate thinking from old habits so as to break through to the Aha! moment of inspiration" (Schneiderman, 2007)—that the 'variables' that guide the rest of the design process are defined (Fulton Suri, 2008; Pahl & Beitz, 1996, in Wynn & Clarkson, 2005).

The latter stages which assign values to these variables involve a more well-bounded deductive process that is much better suited to current computational tools that can iteratively test huge numbers of different values for those variables (Steinfeld, 2017; Papanikolaou, 2012). In comparison, the early phases contain more tacit problem-solving activities, such as collecting diverse information and reframing it in novel ways, that are not served by many computational tools. Figure 1 shows this dearth of computational tools in the activities in the early phases of the design process, underlining our premise and the need for this review of technologies that could inform future CAD tools.

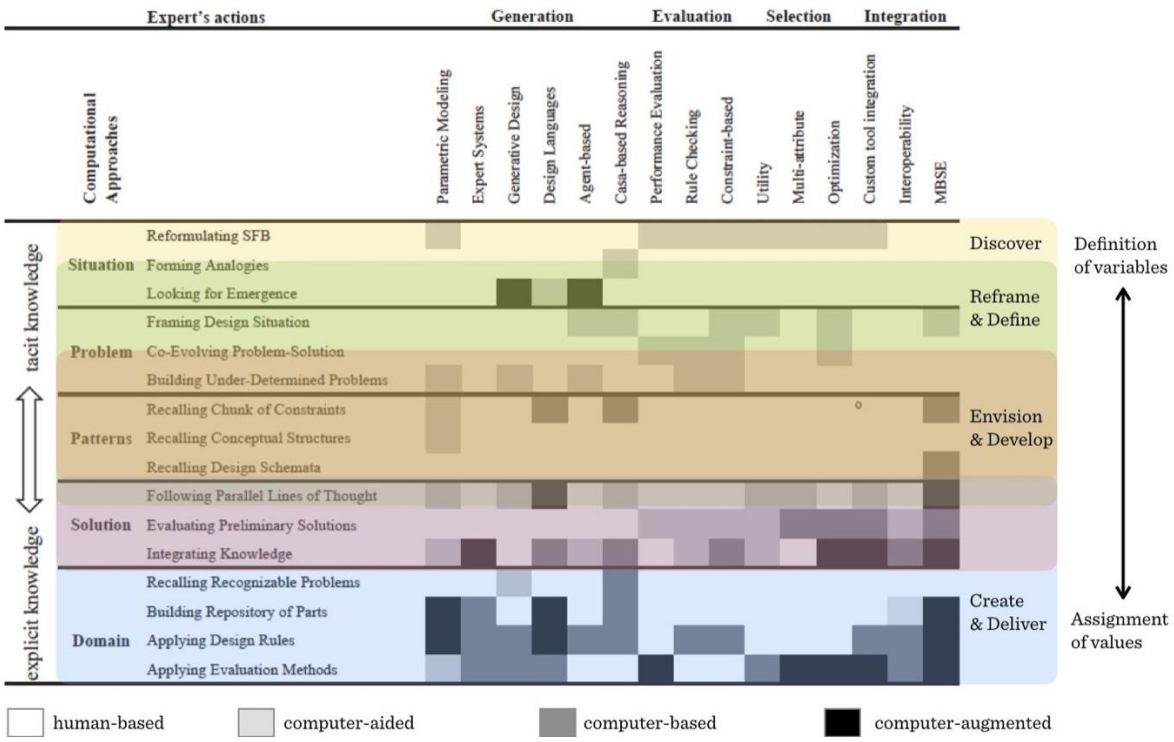


Figure 1. Bernal et al's (2015) diagram of human and computational tools available for the 'actions' in the design process with additional coloured bands added to show the four overlapping design phases

2.2 An epistemological framework for understanding the role of computational technologies in the design process

Considering the lack of computational tools that exist in the early phases of discover and reframe/define, what approach might help us understand how to meaningfully utilise these new tools? CAD tools today specialize in the automatic generation and optimisation of the values for a set of variables defined by the designer and related through an explicitly understood and code-able algorithmic logic (Loukissas, 2012). In comparison to this very structured affordance, the meandering ad hoc experiments carried out in the early creative process appear abstract and loosely defined (Mitchell, 1993; Schön, 1983). These discrepancies highlight the challenges—and opportunities—of applying computation in the early stage of the design process.

However, some of the fundamental attributes of computation and design tools are closer than we might first imagine. A tool is not merely a utilitarian instrument; it can be any physical, digital, or co-nceptual mechanism that enhance our design abilities (McCullough, 1998). And while many modern computational design tools are indeed digital, computation can be more broadly considered as a

process to "reckon things together" (Yalınay-Çinici, 2012) using 'algorithms' that are simply sets of instructions (Algorithm, n.d.). In our view, a computational design tool is therefore an aid that uses a somewhat defined set of instructions to guide the process of designing something, and hence can include anything from the rules of brainstorming to a complex optimizing CAD program.

This somewhat rationalist approach has been taken further by other researchers such as Simon (1969) who strove to integrate cybernetics into the design process. We agree with Margolin (2002) that this very positivist view of the design process is "too remote from actual design situations" and overly mechanistic models of prescribed activities can actually be restrictive to creativity due to their lack of generality (Wynn & Clarkson, 2005; Finke, Ward & Smith, 1992; Schneiderman, 2007). However, we also believe that computation can be a powerful tool—especially new technologies such as machine learning that use a more systems-based approach—and developing strategies to reveal patterns of logic within even the most tacit design activities can help identify areas where these more rational functionalities can potentially enhance our creativity.

Analysing how tools can be used in the design process, Spier (1970) writes: "The use of an artifact is direct and immediate and may be profitably distinguishable from function. The use of a pencil is to make marks on suitable surfaces; its function is to communicate ideas and sentiments." This breakdown is not dissimilar to the 'three levels of analysis' models used to describe perception in cognitive psychology (McClamrock, 1991), where a much larger goal, e.g. communication, is contributed to by smaller tasks, e.g. mark making, and low-level tools, e.g. a pencil. Although this approach breaks down a design activity into the underlying elements, we do not consider this an overly cybernetic model; the affordances of a particular tool may have a certain functionality, but it's output and application can be flexible depending on how it is used by the designer, thus incorporating the more ad hoc principle of bricolage that is more readily used in early phases.

Taking inspiration from this balanced approach to modeling the design process, we propose the following as an epistemological framework for designers and researchers to more easily understand how computational tools might be applied in various activities in the design process:

- **Design activities** i.e. 'what' is being carried out in the design process. Identifies the higher-level activities in which the overall problem or goal is described but not the underlying structures for how it might be achieved. As a designer, you might consider: "The goal of this [design activity e.g. mood board development] is to use [inputs e.g. extreme design themes] to generate [outputs e.g. extreme concept mood boards]..."
- **Design tasks** i.e. 'how' the design activities will be achieved by breaking down the activities into a series of specific tasks, e.g. an algorithm. These tasks describe actions that can be carried out but do not detail the exact tools that will be used. A designer might add to the above sentence by considering: "... using [design knowledge, e.g. contextual understanding] and [specific processes, e.g. image search]…"
- **Design tools** i.e. 'what' will be used to execute the design tasks. The tools (physical or digital) that can be used to many different design tasks and therefore contribute to a range of design activities. A designer might add to the above sentence by considering: "... with [specific media, e.g. fashion magazines, and tools, e.g. Pintrest]"

We believe this could be an instructive and generative framework for considering the potential of computational tools throughout the design process. The following sections use this framework to review the design activities and tasks present in the creative process, and identify a range of computational tools that can be used in these activities.

3 Defining the design activities in the early creative process

A collection of the activities and tasks within the discover and reframe/define phases of the design process as referred to in the literature is shown below in Figure 2.

Author	Discovery phase design activities and tasks	Reframe/define phase design activities and tasks
Alexander (1961, in Dubberly, 2004)	Understand context from actual world *Create mental model of context*	Connect mental models to visual stimuli *Create and connect visualisations of contextual mental models to visual stimuli*
Banathy (1996, in Dubberly, 2004)	Create divergence of information and ideas from an initial genesis *Create alternative images*	Converge information by envisioning possible futures *Create alternative images* *Synthesize and hypothesise image of future system*
Bernal (2015)	Frame the focus of interest Rapidly identify relevant aspects of a problem *Forming analogies* *Looking for emergence*	Shift the direction of design development *Analogy* *Trigger unpredictable inferences* *Reformulation* Frame the design situation
Cross (1990, in Dubberly, 2004)	Decompose the existing situation *Break existing information into constituent parts*	Recompose into a new situation *Reassemble the parts in a new way*
Darke (1978, in Dubberly, 2004)	Collect and generate information	Conjecture new ideas from that information
Doblin (1987, in Dubberly, 2004)	Gather information *Carry out interviews, data searches, field research* Structure the information *Create lists and matrices of data*	
Dubberly & Evenson (2008)		Devise stories about what could happen *Create hypotheses* Model alternatives *Create imagined speculative alternatives*
Finke (1992)	Generate diverse and novel information *Find associations* *Find attributes and infer functions* *Reduce information into categories and exemplars* Find novel interpretations *Shift contexts to reframe information* *Find incongruous info to inspire new understanding* *Find what won't work by finding limitations*	Allow new and unexpected features to emerge *Use analogical transfer, contextual shifting and conceptual interpretation to find new meanings* *Keep ambiguity in the information to allow for reinterpretation* Synthesise and transform information into new ideas *Create conceptual or verbal recombinations*
Fulton Suri (2008)	Collect information from many interpretations *Consider information from empathic, speculative, and interpretive views as well as descriptive and factual.* *Reference analogous situations* *Find extremes and boundary conditions* Learn from subjective experiences and interaction *Integrate personal perspectives from yourself and the team as well as externally* *Challenge interpretations* *Build on information responsively*	
Gero & Maher (1993)	Consider idea from first principles Reframe ideas *Consider information analogies several levels of abstraction away from the original context*	Reinterpreting the existing design *Mutating the features of the original information* Recombine ideas in surprising new ways
IDEO (2004, in Dubberly, 2004)	Gather information through observation *Use shadowing, behavioural mapping, consumer journey, extreme user interviews, story telling to gather and represent information about the project*	Use brainstorming to generate and reframe ideas *Create a large quantity of ideas* *Build on ideas and make them wild* *Represent the ideas in a visual way*

*Figure 2. The design activities and tasks in the discovery and reframe/define phases (activities in **bold**, tasks in italics)*

Author	Discovery phase design activities and tasks	Reframe/define phase design activities and tasks
Jones (1970, in Dubberly, 2004)	Explore the design situation	Perceive or transform the problem structure *Consider alternatives (combine with other elements, new concepts, partial substitution, reduction)*
Lawson (1980, in Dubberly, 2004)	Identify the first insight Prepare for new ideas by exploring that initial insight	Allow for incubation of that information Provide tools to provoke and highlight the moment of illumination
Mendel (2012)	Gather disparate sets of data *Collect information in a semiotic framework/database (labeling and tagging, etc)* *Create structural schemes and frameworks for organising and juxtaposing (bi-polar axes, dimensions, grids, matrices, persona models, etc)* Create questions about the data	Understand relationships and gaps Consider data from multiple perspectives *Deconstruct data and relationships and recombine* *Compare data to similar and dissimilar aspects* *Visually map information in ways to reveal new salience, relationships, and meanings*
Polya (1945, in Dubberly, 2004)	Find and sort the unknown data *Introduce suitable notation* *Separate the various parts of the information*	Find the connection between data and the unknown *Find related problems* *Restate the problem differently*
Schneiderman (2007)	Gather information *Exploratory search of previous and related work* *Create mechanisms for organizing search results* *Use tools for annotation, tagging, and marking* *Find distributions, gaps, and outliers* Draw on knowledge from other designers Rapidly generate multiple alternatives	Explore implications Generate hypotheses Produce some initial ideas Draw on opinions from other designers

Figure 2 cont.

A summary of the main design activities and tasks related to the discover and reframe/define phases are described in Table 1. This list is not proposed to be exhaustive; they are merely 'primary generators' (Darke, 1979) to act as a guiding structure for analysing which computational tools may have potential in the early phases of the design process.

Table 1. Summary of the main design activities and tasks related to the discover and reframe/define phases

Design phase	Design activity	Design tasks
Discover	Gather disparate information	• Use initial insights to find related information • Think about initial insights and information in different contexts • Create divergence using associations, abstractions and analogies
	Sort information	• Collect information in a way that allows easy analysis and comparison, e.g. annotating, tagging and database structures • Decompose information into related attributes/categories • Use structure and categories to look for patterns and questions
Reframe/ define	Generate hypotheses	• Present and recompose information in many representations (word/image) to create stories for possible design alternatives • Allow for ambiguity in these hypotheses to encourage multiple interpretations
	Identify novel directions	• Use analogy or different contexts to interpret information in new ways • Recombine/mutate/substitute the information in new ways to create wildly unexpected inferences and moments of illumination

4 Computational technologies relevant to discovery phase activities

Drawing inspiration from several real world design projects, this section reviews the computational tools that could be applied to execute the tasks in the design activities described in Table 1.

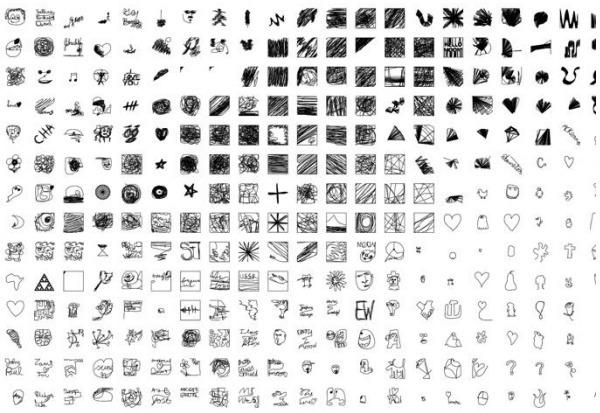
4.1 Design activity: Gather disparate information

4.1.1 Design task: Use initial insights to find related information

The discover phase involves searching for and organising the information related to a design situation in unexpected ways; tasks that even advanced optimising parametric CAD tools such as SolidWorks or Autodesk Dreamcatcher do not provide extensive support for (Bernal et al, 2015). The computational tool that designers often use to help them find information related to their initial prompt is the now ubiquitous semantic search engine such as Google. In this technology, the machine learning technique of dimensionality reduction abstracts a large database that uses many dimensions to connect all of the information into a smaller, more manageable set of key features using linear and non-linear mapping (Barysevich, 2017); not dissimilar to how designers navigate the information related to their projects to learn from related fields (Finke et al, 1992; Mendel, 2012).

A tool that can execute these operations on a corpus of text, and one that forms the basis of many Natural Language Processing tools, is word2vec (www.tensorflow.org/tutorials/ word2vec) (Mikolov et al, 2013). Words are assigned a number based on their connection to others, forming a vector that can be used to compare words in different contexts and find similarities through it's direction and location. A similar strategy can be used to compare images, with a popular algorithm being t-SNE (Maaten & Hinton, 2008); Figure 3 shows how sketches from Golan Levin and David Newbury's (2018) Moon Drawings project can be sorted into similar styles (McDonald, 2016).

Taking this further, Yossarian (www.yossarian.co) adds a 'metaphorical distance' to this vector to return connected words and images with a more diverse interpretation of the initial word and image input by the designer (Figure 4). The details of the technology are not public, but we postulate it does this by adding a factor to change the distance or direction in the vector mathematics connecting the entities in the database. Working with poet Helen Mort to help provide inspiration to write a poem a day ("Helen Mort's poetry challenge with Yossarian", 2015), Yossarian allowed Mort to more quickly connect diverse themes, a crucial part of the early creative process (Minissale, 2013). This computational tool of dimensionality reduction with a vectorising factor to extend the metaphorical search capabilities could therefore potentially help designers find unexpected information in their search activities, leading to more novel design solutions.



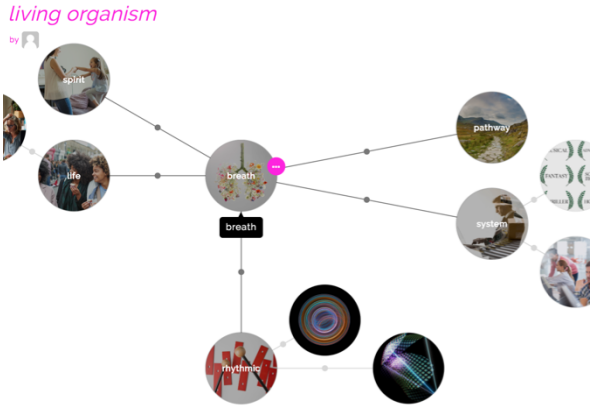


Figure 3 (left). MacDonald's (2016) sorting of Levin and Newbury's (2018) Moon Drawings project sketches
Figure 4 (right). Yossarian metaphorical search engine

4.1.2 Design tasks: Think about initial insights and information in different contexts & Create divergence using associations, abstractions and analogies

Traditional CAD tools often use very structured procedural knowledge and pre-defined geometric relationships to automate certain actions (Bernal et al, 2015), e.g. automatic patterning of shapes in SolidWorks or Adobe Illustrator. This limits the ability of these tools to integrate analogical information into their operations; an important feature to allow for divergent thought and idea generation (Gero & Maher, 1993).

Computational tools with this ability are the machine learning techniques such as convolutional (CNN) and recurrent (RNN) neural networks that are prevalent in image and language processing tools such as IBM's cognitive system Watson. CNNs are useful for image recognition as, after 'learning' patterns from a large training set of tagged images, they can distinguish parts of images related to different categories. RNNs use feedback systems to help them continually learn about the information they are training on and modify the patterns they are seeing, making them very good at parsing and generating new text.

'Living Sculpture' by SOFTLab is a project that used these tools to broaden the perspective of the designers while exploring and identifying trends in the materials, shapes and colours that Gaudi used in his work to influence development of a new sculpture (Lewis, 2017a). Feeding hundreds of tagged images of Gaudi's work, Barcelona and its culture into Watson's Visual Recognition tool taught the system how to recognise the components of those images that 'looked' Gaudi-esque and those that didn't. The system could then compare them to other unrelated images in the database to see if there were any similarities, e.g. it recognised that many of the Gaudi images had depictions of spiders in them. Similarly, Watson's AlchemyLanguage tool analysed various documents about Gaudi and his work as well as Catalan culture, nature and design to identify the most prevalent keywords and concepts. The concepts highlighted using these tools included objects such as 'waves', 'arches', and 'spiders' which were very obvious to the designers familiar with Gaudi, but Watson also helped identify less immediately apparent but very inspiring connections such as the forms, materiality and colours of 'crabs', 'shells' and 'candy' (Wiltz, 2017). The similarity of SOFTLab's work to these elements in Gaudi's designs can be seen in Figure 5 below.

SOFTLab designer Michael Szivos described how Watson's cognitive tools helped them to carry out the tasks they normally do without computers in the early conceptual design stage of a project such as "look at references and try to extract fundamental ideas that we then re-translate into a specific project" (Lewis, 2017b). Integrating these computational tools of CNNs and RNNs into design tools could help designers to not only expand the initial information they were exploring but also quickly parse it to identify both expected and unexpected findings.

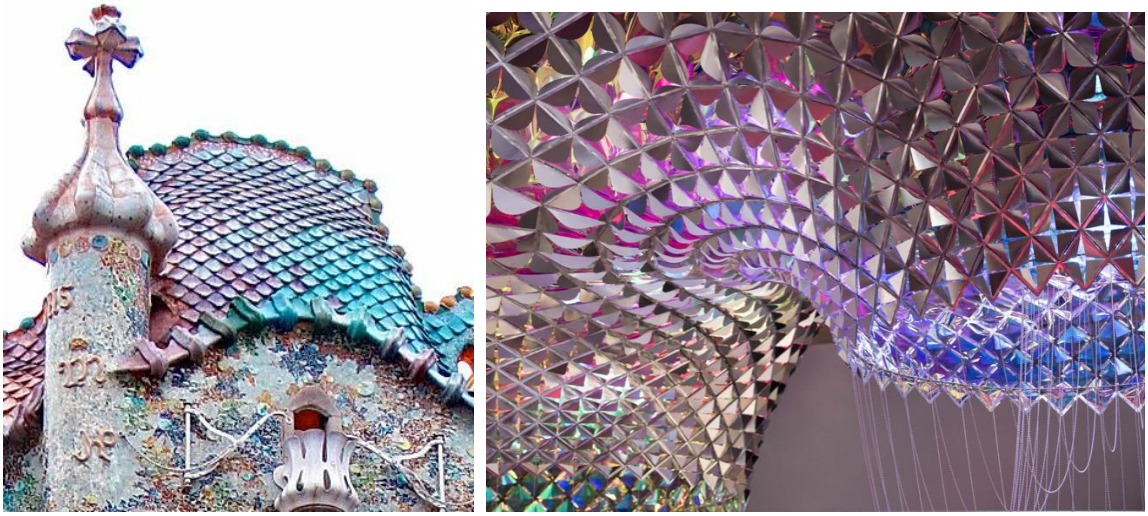


Figure 5. Gaudi's Casa Batlló (left) by Amadalvarez (CC) and SOFTLab & IBM's Living Sculpture (right) showing similar iridescent patterns (SOFTLab, 2017)

4.2 Design activity: Sort information

4.2.1 Design task: Collect information in a way that allows easy analysis and comparison, e.g. annotating, tagging and database structures

Computer assisted qualitative data analysis (CAQDAS) tools to aid the tagging (or coding), sorting and analysis of information collected during research in a design project, such as ATLAS.ti and NVivo, allow researchers to search and pull out common themes from their data, but also require a very

manual coding process (Saldana, 2009); the computational tools described above, on the other hand, only require some of the data to be tagged. A subset of these techniques called unsupervised learning algorithms help automate this process; tools that use CNNs and RNNs, such as the Clarifai application (www.clarifai.com), can learn from a training set of data to automatically tag a wider corpus of images or video and understand the categories present.

Overlapping coding with other stages of the research process can help generate new hypotheses (Eisenhardt, 1989). What if the tools that helped us code the research could also inspire new ideas? An interesting development of this technique created by Fito Segrera is The Treachery of [Soft] Images (2016); a homage to Magritte's painting of similar name where an image of a pipe is described as not being a pipe. Here, images found on the internet are put through a neural network that labels them with humorous—and potentially very inspiring—misinterpretations.

4.2.2 Design tasks: Decompose information into related attributes/categories & Use structure and categories to look for patterns and questions

Traditional CAD systems such as SolidWorks often require a design to be decomposed into its underlying attributes, often limited to geometric properties, in order to store and relate them to each other parametrically, e.g. specifying the points which make up a curve (Tornincasa & Di Monaco, 2010); the decomposition required in the early design process requires different computational tools to identify attributes, find patterns and inspire related categories.

As with the projects described above, it is the combination of CNNs with a dimensionality reduction algorithm that can help decompose and structure the text and image-based information used early in the design process. One such tool is t-SNE (Olah, 2014) which allows an image to be assigned a numerical description associated to the different categories that it is related to. This numerical description can then be compared to others and the images grouped on their visual and contextual similarities (as Karpathy (n.d.) has shown in Figure 6). Taking this further, McDonald (2016) has used t-SNE to place all of the underlying categories of the images in a database next to each other, showing what objects are often found in similar contexts despite being visually different, e.g. pill bottle, band aid and lipstick are grouped closely due to them being found in bathrooms (Figure 7). These tasks—understanding the underlying attributes of images and sorting them based on their classifications—are currently very human-based; integrating this computational technology into a design tool could help designers more quickly structure their research into constructive categories.

Figure 6 (left). Karpathy's (n.d.) visualisation of the similarity of images
Figure 7 (right). McDonald's (2016) representation of the similarities of image categories

5 Computational technologies relevant to reframe/define phase activities

5.1 Design activity: Generate hypotheses

5.1.1 Design tasks: Present and recompose information in many representations (word/image) to create stories for possible design alternatives & Allow for ambiguity in these hypotheses to encourage multiple interpretations

In the early phases where ideas are being defined, designers often imagine how the information collected in the discovery phase could be considered and recombined in new ways to inform future design solutions. Creative writers and artists have often used tools that incorporate chance to provoke ambiguity or absurdity and help them to generate new possibilities for their work (Gaver & Dunne, 1999; Dorin, 2013). Accessing the higher powers through the I Ching, the ancient Chinese method of interpreting a divination text through the random throwing of sticks or dice, has also been used to inspire creative paths for artists such as John Cage and Philip K Dick (Mountfort, 2016).

Computational tools that integrate these chance processes to provoke new design ideas include story generator algorithms (Gervás, 2012) where a predefined structure of a short story or letter or plot is randomly assigns nouns, verbs, adjectives etc. provided by the user into appropriate places (https://www.plot-generator.org.uk/). Despite being so simply structured and often generating ridiculous, unrefined compositions, the ambiguity of the output creates very unexpected and inspiring juxtapositions of concepts and themes. Taking this further, the short film Sunspring used a RNN machine learning algorithm to learn the structure and style of sentences used in dozens of sci-fi screenplays and then generate the content of the script from scratch (Newitz, 2016).

Applying these tools to the design process, these combinatorial technologies could also be used to "trigger unpredictable inferences" in the early phases of the design process (Bernal et al, 2015). Inspired by similar tools that use chance such as Eno and Schmidt's (1975) Oblique Strategies, we developed a website (designhumandesign.media.mit.edu) that uses a stochastic algorithm to recombine variables related to the designer's research into a creative prompt sentence, e.g. "Design [an object, a website, an image, etc.] inspired by [cameras, fashion, healthcare, etc.] that is [approachable, contrasting, responsive etc.] through [personas, layouts, textures, etc.] using [foam, paint, collage, etc.]" (Mothersill & Bove, 2017).

Considering how we might recompose information related to images, much can be learned from the field of data visualisation (Tufte & Robins, 1997). CAQDAS systems integrate some simple visualisation features but are limited in the creative explorations that designers require in these early phases (Bhowmick, 2006). Data visualization artists such as Jared Tarbell have created tools that explore more creative ways of representing data using computational processes that randomize the fonts, sizes and positions of text and images (Figures 8, 9 and 10). These computational tools could help designers juxtapose unexpected concepts from their research by allowing them to intuitively 'find' the elements that inspire them, like gazing at Leonardo's paint stained wall that inspired deliberate accidents (Turner, 2011) but with more purposeful information embedded in it. These visualisations could even become an immersive experience as CAD systems that integrate virtual and augmented reality technologies become more readily available (Arnowitz, Morse & Greenberg, 2017).

Figure 8. Cylinder Image Display by Jared Tarbell (http://www.levitated.net/daily/levCylinderImageDisplay.html)

Figure 9 (left). Text Space by Jared Tarbell (http://www.levitated.net/daily/levTextSpace.html)
Figure 10 (right). Emotion Fractal by Jared Tarbell (http://www.levitated.net/daily/levEmotionFractal.html)

5.2 Design activity: Identify novel directions

5.2.1 Design tasks: Use analogy or different contexts to interpret information in new ways & Recombine/mutate/substitute the information in new ways to create wildly unexpected inferences and moments of illumination

Once the diverse information related to a designer's initial ideas has been collected, and categories have been identified and presented in novel ways, it must all be synthesised into original ideas that can guide the design as it is developed. These new ideas often come from reframing, recombining or mutating the original information and categories into new contexts or interpretations (Gero & Maher, 1993). Despite the real-time manipulation and generation that direct modelling and generative CAD tools such as Autodesk Fusion 360 and Dreamcatcher respectively offer, they merely present a range of options that hope to provoke the 'Aha' moment of inspiration; the human designer is still needed when engaging with these tools to think critically about what is being designed and 'nudge' the algorithm in the preferred direction (Bernal et al, 2015; Bruner, 2016).

The lack of accuracy in predictions generated by the computational tools discussed above can actually help provoke a more inspiring range of design ideas related to the information collected in the discovery phase. Google's Quick, Draw! App (https://quickdraw.withgoogle.com/) is a tool that runs a CNN in real time while the user is sketching a picture and offers many speculative guesses as to what is being drawn (Figure 11); like a game of Pictionary. As the system continually provides guesses of incomplete images, the user is presented with a range of interpretations not associated to the initial intent of the drawing. This creative misinterpretation is not an unfamiliar activity in the design process; a designer's colleagues may see a half drawn sketch and interpret it as something different to the designer's original intent, often inspiring a new idea for their design (Stacey, Eckert & McFadzean, 1999).

Taking this idea further, the AutoDraw app (https://www.autodraw.com/) guesses what the user might be drawing and then uses CNN to find many different illustrations of a similar context from a

database (Figure 12). Again, this offers the designer an interesting real time interpretation of the information they are inputting into the system. Adding RNN to this tool, as in Magenta's sketch-rnn demo (https://magenta.tensorflow.org/sketch-rnn-demo), allows these alternative illustrations to be generated from the actual sketch that the user draws (Figure 13).

Figure 11. Google's Quick, Draw! app showing interpretation of a cat sketch also as a spider, airplane, campfire, etc.

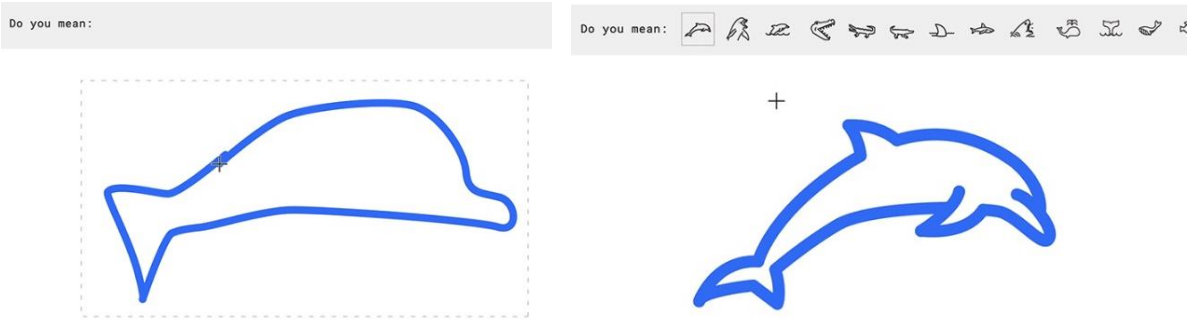


Figure 12. AutoDraw suggesting alternative illustrations for a sketch of a dolphin

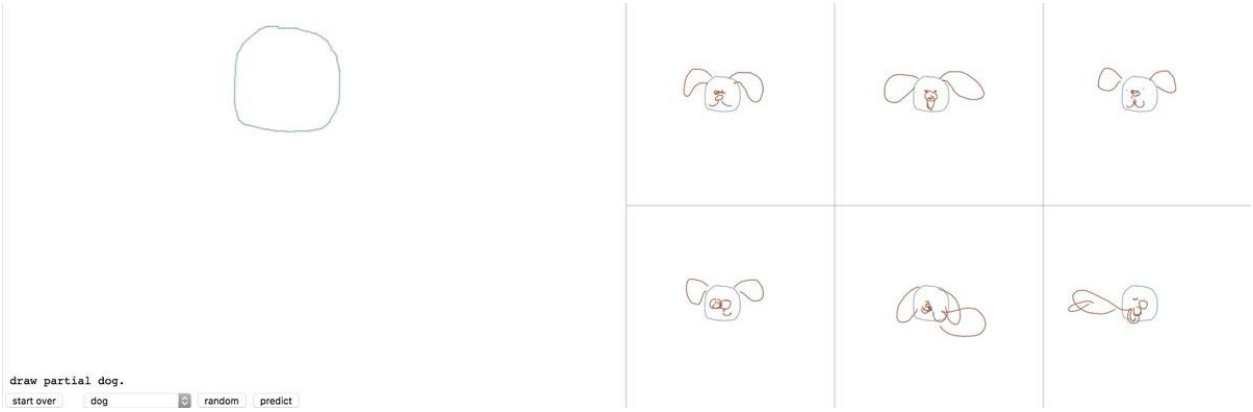
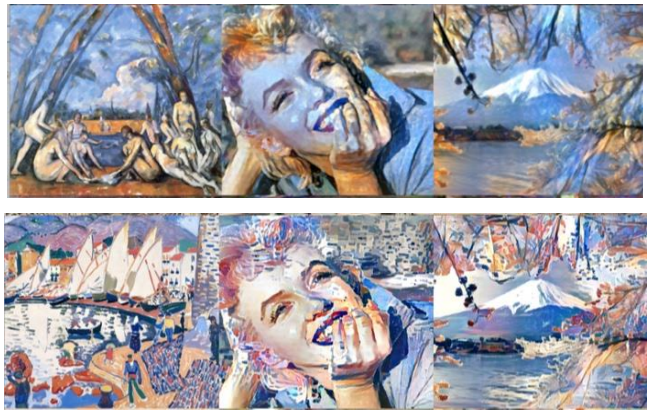


Figure 13. Magenta's sketch-rnn generating sketches of a dog from an initial basic sketch

A more advanced version of these sketching tools are the style transfer algorithms like Google's DeepDream that have become popular in the last few years (Steinfeld, 2017). In these "design by example" tools, CNNs are used to detect the set of context and style features in different images and a feedback technique is used to slowly change the style features of one so that the difference between the two images is reduced (Tejani, 2016). McDonald (2016) has explored this technique extensively, transforming an image of Marylin Monroe and Mount Fuji into versions that could have been painted by all of the artists throughout history (Figure 14). Refining this technology, Korsten

and Flores (2016) 'learned' the style of 17th century master painted Rembrandt and generated a completely new artwork in his style. Integrating more of the user's input as to which areas should be 'transferred' between images, Champandard (2016) uses the idea of analogy to demark areas that have certain categories in the style image, e.g. marking a tree with brown pixels. The user then 'paints' a sketch of a new composition using the same colour scheme, and the CNN transfers the style learned from that section of the style image to only those areas of the new composition (Figure 15).

What is exciting about these computational tools is that these techniques are not unfamiliar to artists, who have been learning, integrating and modifying other artist's styles for centuries. While not achieving the standards of a professional artist, these algorithms provide enough of an idea of what one image in another style would be like—similar to the analogies that designers often apply in their early experiments (Hey et al., 2008)—to inspire the aesthetics and 'feeling' of the design that they will develop.



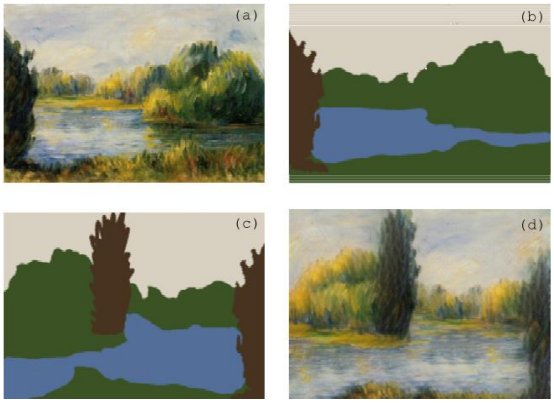


Figure 14 (left). McDonald's (2016) style transfer studies (see more at www.kylemcdonald.net/stylestudies/)
Figure 15 (right). Champandard's (2016) analogy style transfer examples, (a) Original painting by Renoir, (b) semantic annotations, (c) desired layout, (d) generated output.

6 Discussion

This framework helped us review the field of computational tools so as to suggest through examples how these technologies could be implemented in and further developed for the activities in the early phases of the design process.

From this ontology, we can suggest some key computational technologies that could contribute to the development of computational tools applicable to the early phases of the design process. In the discover phase, the activities involved gathering and sorting disparate information. Machine learning algorithms such as CNNs, RNNs and dimensionality reduction techniques are excellent computational tools to parse and categorise the initial information that a designer inputs into a design tool, such as their design research notes, interview transcripts or even inspirational images. Integrating factors that allow for a looser connection between the classification of the data can help the system to search for more analogous information, extending the range of material that the designer can be inspired by. In the reframe/define phase, the activities focused on generating hypotheses and identifying novel directions. Here we suggest that computational tools using stochastic processes to juxtapose the information from the discover phase in new ways, e.g. using visualisation tools that play with the position, size and style of the text and images, could help designers to imagine unfamiliar concepts and novel design ideas. CNNs and RNNs used in story generators and style transfer algorithms can also be used to generate new design 'prompts' for designers to consider and hopefully be inspired by.

While these computational tools offer the potential to enhance our abilities in the early activities, we must also be aware of the limitations of these technologies. Many of the technologies described

above often use symbolic categories within a narrow problem-solving paradigm which are very powerful at analysing text and images from a mathematical point of view, but may be limited when applied to the tacit behavioural thought processes that guide a human designer during the more exploratory and generative activities of the design process (Pfeifer, 1996; Colton & Wiggins, 2012).

Given this critique of these potentially very powerful computational tools for the design activities in the early stages, we would like to propose a few design principles for the use of these technologies and the development of future CAD tools. While machine learning allows for the analysis of much larger collections of information than a designer might be able to when discovering and linking new information together, creating interfaces that are transparent in their computational processes and allow information to be presented as unexpected inspirations, rather than design solution prescriptions, can make these tools more useful in the more exploratory phases of the design process (Colton & Wiggins, 2012; Mothersill & Bove, 2017). Considering the ambiguity present in the ad hoc bricolage nature of the early design phases, these tools should also integrate a "margin of error" in their representations to allow for creative misinterpretation (Bernes, 2017) and thus generate new approaches—or variables—for the design.

As well as applying these principles in our own work developing new computationally-enabled design tools for early phases of the design process, we are currently evaluating the wider potential of the framework described above. Initial feedback from workshops at the Royal College of Art in London (UK) and IDEO design consultancy in Boston (USA) highlighted that more guidance is needed to help other designers and researchers identify appropriately scoped design activities and break them down into the underlying tasks in order to connect to specific tools. Showing a large range of examples of computational technologies helped provide analogies for how computational tools have been applied in unexpected situations and therefore could contribute to very different activities. Building on this feedback, a 'bottom up' application of this framework—where the multiple affordances of tools are described and then applied to other tasks and activities—could be a more generative approach; an approach which actually maps more seamlessly to the historically bricolage tool use that Spier (1970) describes. We hope to keep developing this framework as a new epistemology for understanding the role of computational tools in the design process so as to further empower designers and researchers to impact the development of these future technologies.

7 Conclusion

The CAD tools available today specialise in manipulating and automatically generating optimised designs for a set of pre-defined variables, i.e. they are proficient at the latter phases of the design process where concrete forms and final solutions are envisioned and developed. These tools require very explicit descriptions of a design and as such are not suited to the more abstract, tacit activities present in the early discovery and reframe phases of the design process. This paper considered how new computational technologies could potentially play a role in these early stage design activities.

Using a framework that deconstructed design activities into underlying tasks, we presented a range of computational tools with features appropriate for the more tacit activities present in the early phases of the design process. Such tools included machine learning algorithms such as CNNs, RNNs and dimensionality reduction techniques to help sort information related to a design in the discover phase and stochastic algorithms to help juxtapose the information in new ways in the reframe/define phase. Designing the interfaces of these tools to allow for a more transparent and ambiguous representation of the information can ensure that they are not overly mechanistic or prescriptive and allow for the creative misinterpretations and bricolage nature of the early design process.

Early feedback on this framework as an epistemology for understanding the potential use of computational tools in the early design process has shown that a 'bottom up' approach that demonstrates many computational tools in different design tasks and activities can help provide intuitive knowledge of how the tools work but also inspiration for alternative applications. Developing this work further through workshops and application to the development of new design

tools, we hope this framework and review can help other designers and researchers understand the potential for these computational tools in even the earliest phases of the design process, and offer suggestions for how we might develop future CAD tools that are more appropriate and considerate of the tacit and ambiguous nature of creativity.

8 References

Alexander, C. (1966). From a Set of Forces to a Form. In Kepes, G. (Ed.). The man-made object (pp. 96-107), New York: G. Braziller.

Algorithm (n.d.) In *Oxford Dictionaries,* Retrieved from https://en.oxforddictionaries.com/definition/algorithm

Arnowitz, E., Morse, C. & Greenberg, D.P. (2017, November). vSpline: Physical Design and the Perception of Scale in Virtual Reality. In *ACADIA 2017: DISCIPLINES & DISRUPTION (Proceedings of the 37th Annual Conference of the Association for Computer Aided Design in Architecture)*, pp. 552- 561

Barysevich, A. "Your Keywords Are Not What You Think They Are" SEO Powersuite,
February 28th, 2017. https://www.link-assistant.com/news/keyword-refinements.html

Bernal, M., Haymaker, J. R., & Eastman, C. (2015). On the role of computational support for designers in action. Design Studies, 41, 163-182.

Bernes, J. (2017) The Poetry of Feedback *e-flux Journal, 8*2 Retrieved from http://www.e-flux.com/journal/82/127862/the-poetry-of-feedback/

Blessing, LTM (1994) A process-based approach to computer-supported engineering design. PhD thesis,
University of Twente, The Netherlands

Bhowmick, T. (2006). Building an exploratory visual analysis tool for qualitative researchers. *Proceedings of AutoCarto, Vancouver, WA*.

Bruner, J. (2016, September 12th) "Artificial intelligence and the future of design" Retrieved from https://www.oreilly.com/ideas/artificial-intelligence-and-the-future-of-design

Champandard, A. J. (2016). Semantic style transfer and turning two-bit doodles into fine artworks. *arXiv preprint arXiv:1603.01768*.

Colton, S., & Wiggins, G. A. (2012, August). Computational creativity: The final frontier? In *ECAI* (Vol. 12, pp. 21-26).

Darke J (1979) The primary generator and the design process. Design Studies, 1(1): 36–44

Council, D. (2007). Eleven lessons: Managing design in eleven global companies-desk research report. *Design Council*.

Dorin, A. (2013, July). Aesthetic selection and the stochastic basis of art, design and interactive evolutionary computation. In *Proceedings of the 15th annual conference on Genetic and evolutionary computation* (pp. 311-318). ACM.

Dubberly, H. (2004). How do you design. A Compendium of Models.

Dubberly, H., & Evenson, S. (2008). On modeling The analysis-systhesis bridge model. *interactions*, *15*(2), 57-61.

Eisenhardt, K. M. (1989). Building theories from case study research. *Academy of management review*, *14*(4), 532-550.

Eno, B., & Schmidt, P. (1975). Oblique strategies: Over one hundred worthwhile dilemmas.

Finke, R. A., Ward, T. B., & Smith, S. M. (1992). Creative cognition: Theory, research, and applications.

Fulton Suri, J. (2008). Informing our intuition: Design research for radical innovation. *Rotman Magazine*, 52-57.

Gaver, W., & Dunne, A. (1999, May). Projected realities: conceptual design for cultural effect. In Proceedings of the SIGCHI conference on Human Factors in Computing Systems (pp. 600-607). ACM.

Gero, J. S. (1990). Design prototypes: a knowledge representation schema for design. AI magazine, 11(4), 26.

Gero, J. S., & Maher, M. L. (Eds.). (1993). *Modeling creativity and knowledge-based creative design*. Psychology Press.

Gervás, P. (2012). Story generator algorithms. *The Living Handbook of Narratology. Hamburg: Universidade de Hamburgo. Disponível em: http://www.lhn.uni-hamburg.de/article/story-generator-algorithms Acesso em*, *19*.

"Helen Mort's poetry challenge with Yossarian", Yossarian Lives. June 1st, 2015. http://blog.yossarianlives.com/post/120437413305/helen-morts-poetry-challenge-with-yossarian

Hey, J., Linsey, J., Agogino, A. M., & Wood, K. L. (2008). Analogies and metaphors in creative design. *International Journal of Engineering Education*, *24*(2), 283.

Karpathy, A. (n.d.) "t-SNE visualization of CNN codes" Retrieved from http://cs.stanford.edu/people/karpathy/cnnembed/

Korsten, B. & Flores, E. (2016) "The Next Rembrandt." J. Walter Thompson Amsterdam. Retrieved from www.nextrembrandt.com

Levin, G. & Newbury, D. (2018) Moon Drawings project. Retrieved from http://www.moondrawings.org/

Lawson, B. (2006). How designers think: The design process demystified. Routledge.
Lewis, K. “The First Thinking Sculpture: Inspired by Gaudi, created with Watson” IBM Internet of Things. February 28th, 2017a. https://www.ibm.com/blogs/internet-of-things/first-thinking-sculpture/
Lewis, K. “Using creativity to frame new technologies in a positive way” IBM Internet of Things. March 1st, 2017b. https://www.ibm.com/blogs/internet-of-things/creativity-watson/
Loukissas, Y. A. (2012). *Co-designers: cultures of computer simulation in architecture*. Routledge.
Maaten, L. V. D., & Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, *9*(Nov), 2579-2605.
Margolin, V. (2002). *The politics of the artificial: Essays on design and design studies*. University of Chicago press.
McDonald, K. (2016, October 7th) “A Return to Machine Learning.” Retrieved from https://medium.com/@kcimc/a-return-to-machine-learning-2de3728558eb
McClamrock, R. (1991). Marr's three levels: A re-evaluation. Minds and Machines, 1(2)
McCullough, M. (1998). *Abstracting craft: The practiced digital hand*. MIT press.
Mendel, J. (2012). A taxonomy of models used in the design process. interactions, 19(1), 81-85.
Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems* (pp. 3111-3119).
Michel, R. (2007). Design research now. *Essays and Selected Projects, London*.
Minissale, G. (2013). The psychology of contemporary art. Cambridge University Press.
Mitchell, W. J. (1993). A computational view of design creativity. In Gero, J. S., & Maher, M. L. (Eds.). Modeling creativity and knowledge-based creative design, (pp. 25-42), Psychology Press.
Mothersill, P. & Bove, V.M. (2017) Humans, Machines and the Design Process. Exploring the Role of Computation in the Early Phases of Creation, The Design Journal, 20:sup1, S3899-S3913
Mountfort, P. (2016). The I Ching and Philip K. Dick's The Man in the High Castle. *Science Fiction Studies*, *43*(2), 287-309.
Newitz, A. “Movie written by algorithm turns out to be hilarious and intense” Ars Technica. June 9th, 2016. https://arstechnica.com/gaming/2016/06/an-ai-wrote-this-movie-and-its-strangely-moving/
Olah, C. (2014, October 9th) “Visualizing MNIST: An Exploration of Dimensionality Reduction” Retrieved from http://colah.github.io/posts/2014-10-Visualizing-MNIST/
Papanikolaou, D., (2012). Changing Forms, Changing Processes. In Kara, H. et al. (Eds.), Interdisciplinary Design: New Lessons from Architecture and Engineering, (pp. 106-114), ACTAR Press
Pfeifer, R. (1996). Symbols patterns, and behavior: beyond the information-processing metaphor. *Encyclopedia of Microcomputers*, *17*, 253-275.
Saldana, J. (2009). An introduction to codes and coding. *The coding manual for qualitative researchers*, 1-31.
Schneiderman, B. (2007). Creativity support tools: Accelerating discovery and innovation. *Communications of the ACM*, *50*(12), 20-32.
Schön, D. A. (1983). *The reflective practitioner: How professionals think in action* (Vol. 5126). Basic books.
Segrera, F. (2016) “The Trechery of [Soft] Images” Retrieved from http://designsociety.cn/en/category/person-list/detail!Fito-Segrera
Simon, H. A. (1969). The sciences of the artificial. *Cambridge, MA*.
Sjoberg, C., Beorkrem, C., Ellinger, J. (2017, November). Emergent Syntax: Machine Learning for the Curation of Design Solution Space. In *ACADIA 2017: DISCIPLINES & DISRUPTION (Proceedings of the 37th Annual Conference of the Association for Computer Aided Design in Architecture)*, pp. 552- 561
SOFTLab (2017) IBM Mobile World Congress, Barcelona. Retrieved from http://softlabnyc.com/portfolio/ibm/
Spier, R. F. (1970). From the hand of man: primitive and preindustrial technologies. Houghton Mifflin Company, Boston
Stacey, M. K., Eckert, C. M., & McFadzean, J. (1999, August). Sketch interpretation in design communication. In *Proceedings of the 12th International Conference on Engineering Design*(Vol. 2, pp. 923-928).
Steinfeld, K. (2017, November). Dreams May Come. In *ACADIA 2017: DISCIPLINES & DISRUPTION (Proceedings of the 37th Annual Conference of the Association for Computer Aided Design in Architecture)*, pp. 590- 599
Tejani, S. “Artistic Style Transfer with Deep Neural Networks” From Bits to Brains. December 27th, 2016. https://shafeentejani.github.io/2016-12-27/style-transfer/
Tornincasa, S., & Di Monaco, F. (2010, September). The future and the evolution of CAD. In *Proceedings of the 14th international research/expert conference: trends in the development of machinery and associated technology* (pp. 11-18).
Tufte, E. R., & Robins, D. (1997). Visual explanations.
Turner, C. “The deliberate accident in art” Tate. January 1st, 2011. http://www.tate.org.uk/context-comment/articles/deliberate-accident-art

Wiltz, C. “IBM Watson Helps Create Sculpture Inspired by Gaudi” Design News. March 1st, 2017. https://www.designnews.com/design-hardware-software/ibm-watson-helps-create-sculpture-inspired-gaudi/147823998856397

Wynn, D., & Clarkson, J. (2005). Models of designing. In *Design process improvement* (pp. 34-59). Springer London.

Yalınay-Çinici, S. (2012). Computation: Uneasy to Translate And Understand - Language, Thought And Architecture. In Gun, O.Y. (Ed.) Dosya 29: Computational Design (pp. 12-18) UCTEA The Chamber of Architects of Turkey Ankara Branch

About the Authors:

Philippa Mothersill is a PhD student in the MIT Media Lab Object-Based Media group, where she draws from design theory, cognitive psychology and computer science to develop new computational design tools for early stages in the creative process.

V. Michael Bove, Jr. is the head of the Object-Based Media group. He is the author or co-author of over 100 journal or conference papers on digital television systems, video processing hardware/software design, multimedia, scene modeling, user interfaces, visual display technologies, and optics.