Reimagining the Role of the Expert: From Interface Design to Interface Curation

Kashyap Todi
Department of Communication and Networking
Aalto University, Helsinki, Finland
kashyap.todi@gmail.com

ABSTRACT

User Interface (UI) design has been a core topic of HCI research for several decades. Equipped with design skills and knowledge, the expert interface designer meticulously analyses a design brief, conceptualises design ideas, and constructs viable solutions. The intended outcome of this tedious process is a usable and aesthetically-pleasing UI. Classical approaches in HCI have relied upon providing designers with guidelines, heuristics, and best practices for realising good designs. In recent years, computational approaches have turned towards formalising and automating parts of the design process. In this provocation, I claim that the future expert will hand over the task of creating design solutions entirely to the machine, and instead take on the role of an *interface curator* who inspects a set of feasible designs and picks out the best possible solutions for a given problem. I discuss the current state of computational interface design, and suggest a path forward towards realising this vision.

CCS CONCEPTS

• **Human-centered computing → Interactive systems and tools**.

KEYWORDS

ACM Reference Format:
Kashyap Todi. 2019. Reimagining the Role of the Expert: From Interface Design to Interface Curation. In *Proceedings of the Halfway to the Future Symposium 2019 (HTTF 2019), November 19–20, 2019, Nottingham, United Kingdom.* ACM, New York, NY, USA, 3 pages. https://doi.org/10.1145/3363384.3363386

1 BACKGROUND AND INTRODUCTION

The scientific community has invested a lot of effort on the design of user interfaces and interactions. Researchers in fields of HCI and design have proposed tools, techniques, and methods to improve both the design process and the outcomes in terms of the resulting UIs or interactions. Initially, the focus was on better understanding how we could design interfaces to make them usable, or how graphical interfaces could use concepts familiar from the physical world. This led to design concepts such as affordances and natural mappings [19], design heuristics [18], and UX approaches such as skeumorphic design. Further, researchers and practitioners identified the need to involve users during early stages of design, thus leading to design processes such as user-centred [20] and participatory design [22]. Designers often adopted a “fail fast, fail often” ideology, and iterated over designs to improve them. The Design–Implement–Analyse (DIA) cycle is now a well-known approach towards iterative UI design [3].

However, designing using these classical techniques come with their own pitfalls. Design principles are not generalisable, and often hard to keep track of. It seems infeasible to expect a designer to remember every design principle, guideline, or best practice, while they are designing interfaces. Even within user-centred design, it is challenging to include every category of user during the design process. This can lead to biases in the final outcomes, and could result in designs that do not meet the needs of marginalised user populations, or users with particular needs (e.g. accessibility). Further, iterative design is time-consuming, and the trial-and-error process does not lead to a better understanding of what encompasses “good interfaces”.

Another line of HCI research has investigated systematically modelling users, and applying these models towards improving interface design. Initially, the Human-Processor Model [4] explained different cognitive processes that take place when we interact, and provided us with a formal understanding of timing requirements. Fitts’ Law [9] has been one of the most well-investigated HCI models, and has been effectively applied towards evaluating and improving interface designs. Several predictive models of aspects relating to aesthetics, performance, and usability have been proposed and developed. These range from aspects such as visual search [14], which models eye movements, to colour harmony [5], which quantifies how different colour compositions can influence aesthetics of an interface. Recent works (e.g. [21, 26]) have used a combination of models of human perception and performance for generating design solutions. By mathematically formulating a good user interface, we can now overcome several drawbacks of completely manual interface design.

In this provocation, I look towards the future of user interface design, and propose that the future interface design expert will not be as much the interface designer, but rather an *interface curator*. Their fundamental task then will not be to turn a design brief single-handedly (and manually) into a fully-functional UI. Instead, they will now provide systems with concrete design tasks, which will automatically be converted to feasible design solutions by the machine. Much like a curator at an art gallery, they will use their expertise and insight to browse through a large set of feasible designs and select the most desirable solutions for a given problem. When talking about how we perceive machine agents,

HTTF 2019, November 19–20, 2019, Nottingham, United Kingdom

ACM ISBN 978-1-4503-7203-9/19/11…$15.00
https://doi.org/10.1145/3363384.3363386

Suchman [24] discusses the distinction between the physical and social. Traditionally, the design tool has been seen as just an artefact that responds to a designer's actions by providing them with expected outcomes. In [17], Brad Myers hinted that *"tools might enforce or at least encourage user interfaces that were highly usable, rather than today's stance that tools should be neutral and leave the design mostly to the human designer"*. With these as inspiration, I suggest that future tools will take this one step further, and act as social agents that take charge of fully generating design outcomes that are objectively good and usable. By doing so, the human expert and the artificial designer could find common ground to open up new lines of communication and collaboration.

2 STATE-OF-THE-ART: WHERE ARE WE NOW?

Over the previous years, several approaches have been proposed for machine-assisted interface design. There has been an extensive amount of research on methods, tools, and techniques towards computational design. While a full discussion of all works in this area is beyond the scope of this paper, I discuss some key aspects and evolution in the state-of-the-art.

Initially, the focus of computational user interface design was on completely automating the generation of interfaces. Model-based methods typically accept a design-specification as input, and used this to construct full layouts (e.g. [2, 7, 10]). They could generate alternative layouts for different user needs [11], or for different screen resolutions [8], thus addressing issues such as accessibility and device compatibility. Automatic approaches could retarget the content of an interface to match an example design [16], or could adapt the interface based on a collection of previously seen interface designs [25]. However, these do not fit well with traditional design approaches. In retrospect, it seems like a majority of these techniques and tools were targeted towards engineers who lacked the skills or knowledge to create good user interfaces. This resulted in the necessity to over-specify the design problem, which seems to be incompatible with interface designers' workflows.

Mixed-initiative computing [12] suggests that humans and machine work on a common task collaboratively, while distributing tasks that best suit them. In this spirit, recent interactive design tools (e.g. DesignScape [21], Sketchplore [26]) attempt to provide computational support without burdening the designer with task specification or problem engineering activities. They fit in with designer activities, and actively suggest design improvements and alternatives based on what the system can automatically infer about the design task. These tools apply different computational approaches that can, for example, generate entirely new designs using predictive models or design heuristics, or mimic existing designs through example-based retargeting [16]. Optimisation techniques have been proposed for keyboard layouts [13], menu designs [1], and for gestural interactions [23] too. These enable the system to find objectively good solutions from a large design space within a feasible timespan. Further, machine learning approaches such as multi-armed bandits can support exploration and exploitation of a large number of design alternatives.

3 CREATING DESIGN TOOLS FOR INTERFACE CURATION

To support future interface curators, interactive design tools will have to organically integrate with the design curation process. There are some key components that would be required towards realising these goals;

(1) **Specifying the design task:** The first step towards creating a design solution is to analyse the design brief, and concretise the problem at hand. Interactive tools will need to allow designers to flexibly provide concrete design requirements for the task at hand. This design task provides the starting point (input) for a system or algorithm that can find design solutions. Some possible techniques for specifying the input could be through sketching, textual descriptions, or conversational interfaces. The system should avoid requiring over-specification of the problem, which although beneficial for finding solutions, can be cumbersome or infeasible for the design expert.

(2) **Deriving semantics of the target interface:** Semantics provide meaning to an interface. They cover aspects such as relationship between elements and flow of user interactions. Current computational techniques that automatically generate design solutions often ignore the semantics of the interface. This can result in designs that might in theory perform well, but are not functional in practice. It will be beneficial for future tools to explore techniques that enable the designers to conveniently explicate the semantics of a desired interface, or to (semi-)automatically infer the intended semantics, for example, through data mining approaches [6, 15].

(3) **Finding design solutions:** The key responsibility of a system that supports interface curators is to find feasible design solutions for a given design task. In this step, the machine takes the design task and desired semantics as input, and detects suitable designs from a large design space. To do so, approaches such as generative design, retargeting, and design optimisation could be adopted and improved upon. An important consideration for an interactive tool would be to construct or find these solutions rapidly in real-time.

(4) **Presenting the viable solutions:** Another important aspect for a tool that enables curation is the actual presentation of found design solutions to the expert. Automated design approaches can often lead to the generation of several design solutions, which although theoretically unique, are practically quite similar in nature. The tool should not overwhelm the curator with a surmounting number of choices, especially several similar solutions. Instead, they should aid the curator in the process of discovering several diverse solutions that meet the design requirements. This could be achieved by systematically measuring the diversity between designs, applying approaches such as exploration-and-exploitation, and by using intelligent visualisation techniques.

(5) **Iterative improvement of solutions:** As designers select desirable alternatives, they might detect the need to further

refine the designs. Tools should thus support the iterative design approach, enabling quick improvements once an initial set of viable designs is found.

(6) **Providing explanations and design rationale:** To make the results interpretable to the curator, and to support the design choices made while finding solutions, tools should also strive towards providing explanations and rationale as to the selected designs are objectively good. By doing so, they can also highlight differences between several solutions. For instance, two designs might be quite different as they are intended towards two distinct user populations. It would be desirable to highlight such aspects to the curator, thus enabling them to make informed choices.

3.1 Discussion

In this provocation, I have proposed a new role for the UI design expert within the interface design process: one of the interface curator. While interacting with future design tools, these experts will now, instead of spending valuable time constructing new designs from scratch, utilise their skills towards recognising the best design solutions from a set of feasible designs that are generated or selected by a machine. The dialogue between the curator and the proactive design tool will enable rapid generation of final designs that are justifiably and objectively usable. Further, they will enable the detection of multiple good solutions, each catering to different needs and requirements of user populations. By reconfiguring the roles of the human and the machine as proposed in the paper, the next generation of interface and interaction design can ensure better use of available resources and better outcomes for the end-user.

REFERENCES

[1] Gilles Bailly, Antti Oulasvirta, Timo Kötzing, and Sabrina Hoppe. MenuOptimizer: interactive optimization of menu systems. In *Proceedings of the 26th annual ACM symposium on User interface software and technology - UIST '13* (2013). ACM Press, 331–342. DOI: http://dx.doi.org/10.1145/2501988.2502024

[2] C. M. Beshers and S. Feiner. Scope: Automated Generation of Graphical Interfaces. In *Proceedings of the 2Nd Annual ACM SIGGRAPH Symposium on User Interface Software and Technology* (1989) *(UIST '89)*. ACM, 76–85. DOI: http://dx.doi.org/10.1145/73660.73670 event-place: Williamsburg, Virginia, USA.

[3] William Buxton and Richard Sniderman. 1980. Iteration in the Design of the Human-Computer Interface. (1980).

[4] Stuart K. Card, Thomas P. Moran, and Allen Newell. 1986. *The model human processor: An engineering model of human performance.* John Wiley & Sons.

[5] Daniel Cohen-Or, Olga Sorkine, Ran Gal, Tommer Leyvand, and Ying-Qing Xu. Color Harmonization. In *ACM SIGGRAPH 2006 Papers* (2006) *(SIGGRAPH '06)*. ACM, 624–630. DOI: http://dx.doi.org/10.1145/1179352.1141933 event-place: Boston, Massachusetts.

[6] Biplab Deka, Zifeng Huang, Chad Franzen, Joshua Hibschman, Daniel Afergan, Yang Li, Jeffrey Nichols, and Ranjitha Kumar. Rico: A Mobile App Dataset for Building Data-Driven Design Applications. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology* (2017) *(UIST '17)*. ACM, 845–854. DOI: http://dx.doi.org/10.1145/3126594.3126651 event-place: Québec City, QC, Canada.

[7] Jacob Eisenstein, Jean Vanderdonckt, Jean Vanderdonckt, and Angel Puerta. Applying Model-based Techniques to the Development of UIs for Mobile Computers. In *Proceedings of the 6th International Conference on Intelligent User Interfaces* (2001) *(IUI '01)*. ACM, 69–76. DOI: http://dx.doi.org/10.1145/359784.360122 event-place: Santa Fe, New Mexico, USA.

[8] J. Falb, R. Popp, T. Rock, H. Jelinek, E. Arnautovic, and H. Kaindl. Fully-automatic generation of user interfaces for multiple devices from a high-level model based on communicative acts. In *2007 40th Annual Hawaii International Conference on System Sciences (HICSS'07)* (2007-01). 26–26. DOI: http://dx.doi.org/10.1109/HICSS.2007.236

[9] Paul M. Fitts. 1954. The information capacity of the human motor system in controlling the amplitude of movement. 47, 6 (1954), 381–391. DOI: http://dx.doi.org/10.1037/h0055392

[10] Krzysztof Gajos and Daniel S. Weld. SUPPLE: Automatically Generating User Interfaces. In *Proceedings of the 9th International Conference on Intelligent User Interfaces* (2004) *(IUI '04)*. ACM, 93–100. DOI: http://dx.doi.org/10.1145/964442.964461

[11] Krzysztof Z. Gajos, Daniel S. Weld, and Jacob O. Wobbrock. 2010. Automatically generating personalized user interfaces with Supple. 174, 12 (2010), 910–950. DOI: http://dx.doi.org/10.1016/j.artint.2010.05.005

[12] Eric Horvitz. Principles of mixed-initiative user interfaces. In *Proceedings of the SIGCHI conference on Human factors in computing systems the CHI is the limit - CHI '99* (1999). ACM Press, 159–166. DOI: http://dx.doi.org/10.1145/302979.303030

[13] Andreas Karrenbauer and Antti Oulasvirta. Improvements to keyboard optimization with integer programming. In *Proceedings of the 27th annual ACM symposium on User interface software and technology - UIST '14* (2014). ACM Press, 621–626. DOI: http://dx.doi.org/10.1145/2642918.2647382

[14] David E. Kieras and Anthony J. Hornof. Towards Accurate and Practical Predictive Models of Active-vision-based Visual Search. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2014) *(CHI '14)*. ACM, 3875–3884. DOI: http://dx.doi.org/10.1145/2556288.2557324 event-place: Toronto, Ontario, Canada.

[15] Ranjitha Kumar, Arvind Satyanarayan, Cesar Torres, Maxine Lim, Salman Ahmad, Scott R. Klemmer, and Jerry O. Talton. Webzeitgeist: Design Mining the Web. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2013) *(CHI '13)*. ACM, 3083–3092. DOI: http://dx.doi.org/10.1145/2470654.2466420

[16] Ranjitha Kumar, Jerry O. Talton, Salman Ahmad, and Scott R. Klemmer. Bricolage: Example-based Retargeting for Web Design. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (2011) *(CHI '11)*. ACM, 2197–2206. DOI: http://dx.doi.org/10.1145/1978942.1979262 event-place: Vancouver, BC, Canada.

[17] Brad Myers, Scott E. Hudson, Randy Pausch, and Randy Pausch. 2000. Past, Present, and Future of User Interface Software Tools. 7, 1 (2000), 3–28. DOI: http://dx.doi.org/10.1145/344949.344959

[18] Jakob Nielsen. 1994. 10 Heuristics for User Interface Design. (1994). https://www.nngroup.com/articles/ten-usability-heuristics/

[19] Donald A. Norman. *The Design of Everyday Things.* Basic Books, Inc.

[20] Donald A. Norman and Stephen W. Draper. *User Centered System Design; New Perspectives on Human-Computer Interaction.* L. Erlbaum Associates Inc.

[21] Peter O'Donovan, Aseem Agarwala, and Aaron Hertzmann. DesignScape: Design with Interactive Layout Suggestions. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems - CHI '15* (2015). ACM Press, 1221–1224. DOI: http://dx.doi.org/10.1145/2702123.2702149

[22] Douglas Schuler and Aki Namioka. 1993. Participatory Design: Principles and Practices. (1993).

[23] Srinath Sridhar, Anna Maria Feit, Christian Theobalt, and Antti Oulasvirta. Investigating the Dexterity of Multi-Finger Input for Mid-Air Text Entry. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems* (2015) *(CHI '15)*. ACM, 3643–3652. DOI: http://dx.doi.org/10.1145/2702123.2702136 event-place: Seoul, Republic of Korea.

[24] Lucy A. Suchman. 2006. *Human-Machine Reconfigurations: Plans and Situated Actions.* Cambridge University Press, New York, NY, USA.

[25] Kashyap Todi, Jussi Jokinen, Kris Luyten, and Antti Oulasvirta. Familiarisation: Restructuring Layouts with Visual Learning Models. In *23rd International Conference on Intelligent User Interfaces* (2018) *(IUI '18)*. ACM, 547–558. DOI: http://dx.doi.org/10.1145/3172944.3172949

[26] Kashyap Todi, Daryl Weir, and Antti Oulasvirta. Sketchplore: Sketch and Explore with a Layout Optimiser. In *Proceedings of the 2016 ACM Conference on Designing Interactive Systems* (2016) *(DIS '16)*. ACM, 543–555. DOI: http://dx.doi.org/10.1145/2901790.2901817