Computational Design Driven by Aesthetic Preference

Yuki Koyama
The University of Tokyo
Tokyo, Japan
koyama@is.s.u-tokyo.ac.jp

ABSTRACT

Tweaking design parameters is one of the most fundamental tasks in many design domains. In this paper, we describe three computational design methods for parameter tweaking tasks in which *aesthetic preference*—how aesthetically preferable the design looks—is used as a criterion to be maximized. The first method estimates a preference distribution in the target parameter space using crowdsourced human computation. The estimated preference distribution is then used in a design interface to facilitate interactive design exploration. The second method also estimates a preference distribution and uses it in an interface, but the distribution is estimated using the editing history of the target user. In contrast to these two methods, the third method automatically finds the best parameter that maximizes aesthetic preference, without requiring the user of this method to manually tweak parameters. This is enabled by implementing optimization algorithms using crowdsourced human computation. We validated these methods mainly in the scenario of photo color enhancement where parameters, such as brightness and contrast, need to be tweaked.

Author Keywords

Computational design; design exploration; aesthetic preference

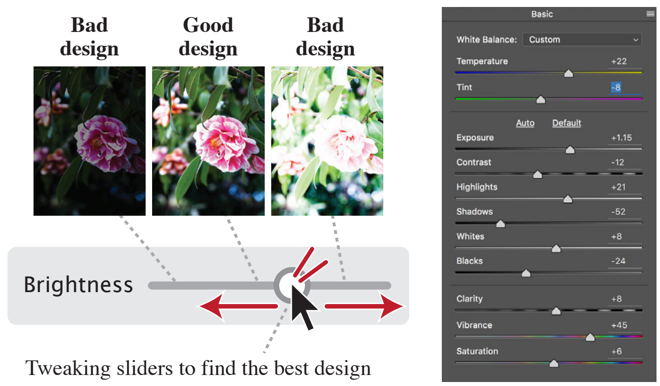
ACM Classification Keywords

H.5.m. Information Interfaces and Presentation (e.g. HCI): Miscellaneous

INTRODUCTION

Background and Goal

Quality of designed objects could be assessed using various criteria according to their usage contexts. Especially in visual graphic designs, *aesthetic preference*—how aesthetically preferable the design looks—is an important criterion. For example, photo color enhancement, or tonal adjustment of photographs, is one of such design scenarios where aesthetic preference is used as a criterion to be maximized (see Figure 1 (a)). When designers enhance color of photographs, they tweak design parameters such as "brightness" or "contrast" via sliders to find aesthetically the best enhancement. When the number of design parameters is small, this design task might not be very tedious; however, as the number of design parameters increases, the possibilities of design expand in an exponential manner, which makes this task difficult. Note that, in actual photo color enhancement, there are tens of sliders that have to be tweaked (see Figure 1 (b)).



(a) (b)

Figure 1. Photo color enhancement is an example of design scenarios where aesthetic preference is used as a criterion. (a) An illustration of exploration for photo color enhancement. Designers tweak sliders such as "brightness" so that they can find aesthetically the best preferable photo enhancement. (b) A real example of sliders for photo color enhancement in Adobe Photoshop CC. In practice, designers have to tweak tens of variables to find the best color enhancement.

Similar parameter tweaking tasks can be observed in many design software. For example, in Unity (a game authoring tool) or Maya (an animation authoring tool), there are many sliders in the control panes, which should be tweaked to adjust the visual of contents. We aim to support (or possibly automate) this general design task of preference-based parameter tweaking.

Problem Formulation

We investigate computational design methods for facilitating design exploration in which aesthetic preference is used as a criterion. Specifically, we consider the design exploration where several design parameters (*e.g.,* "brightness" in Figure 1 (a)) should be tweaked such that the aesthetic preference criterion is maximized. This can be mathematically rephrased as follows. Suppose that there are n design variables $\mathbf{x} = (x_1, \ldots, x_n) \in \mathcal{D}$, where $\mathcal{D}$ represents an n-dimensional design space. We assume that $\mathcal{D} = [0, 1]^n$ for simplicity,

UIST'16 Adjunct, October 16–19, 2016, Tokyo, Japan.
ACM ISBN 978-1-4503-4531-6/16/10.
http://dx.doi.org/10.1145/2984751.2984781

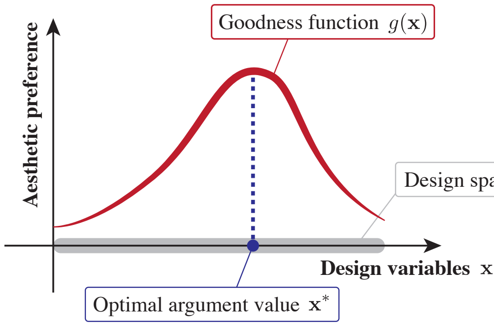


Figure 2. An illustration of our problem setting. We want to provide computational design methods to solve Equation 1, or to find the optimal solution $\mathbf{x}^*$ that maximizes the aesthetic preference of the target design.

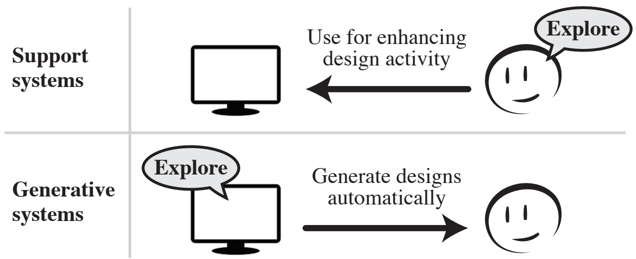


Figure 3. Two approaches of computational design systems. (Top) *Support systems* facilitate users' design exploration by providing computational support functions. (Bottom) *Generative systems* automatically generate designs using computational procedures.

i.e., each variable takes a continuous value and its interval is regularized into $[0, 1]$ in advance. Our goal is to solve the following optimization problem:

$$\mathbf{x}^* = \arg \max_{\mathbf{x} \in \mathcal{D}} g(\mathbf{x}), \quad (1)$$

where the objective function, $g : \mathcal{D} \to \mathbb{R}$, is a *goodness function* that returns a scalar value representing how aesthetically preferable, or good, the design corresponding to the argument design variables is. By solving this optimization problem, we want to find $\mathbf{x}^*$, which is the optimal value that gives the best preferable design. Figure 2 illustrates this problem setting.

Note that the goodness function $g(\cdot)$ is only in designers' minds, and thus usually difficult to represent as a simple equation or rule. Thus, this optimization problem cannot be solved by typical numerical optimization techniques in computer science, which is why the viewpoint of human-computer interaction is important for tackling this problem.

Our Methods

To solve the design problem of Equation 1, there are two possible computational approaches to be investigated. The first approach is to let users explore the design space: developing a *support system* to support users to explore the design space to find the best parameter set $\mathbf{x}^*$ (Figure 3 (Top)). The second one is to let systems explore the design space: developing a *generative system* to automatically generate the optimal design by finding the best parameter set $\mathbf{x}^*$ through the system's algorithmic exploration (Figure 3 (Bottom)).

Based on these two approaches, we present the following three methods:

Method A: This method estimates the goodness function $g(\cdot)$ using crowdsourced human computation, and utilizes the estimated goodness function for facilitating the user's interactive design exploration. See [3] for details.

Method B: This method also estimates the goodness function $g(\cdot)$, but from the editing history of the target user. Then, it utilizes the estimated goodness function for facilitating the user's further design exploration. See [4] for details.

Method C: This method does not estimate the goodness function $g(\cdot)$, but does search the maximum of $g(\cdot)$ directly, using crowdsourced human computation.

Table 1 summarizes the relationship among these three methods. Our methods are data-driven rather than rule-based, and we exploit two different sources for gathering such necessary preference data: *human computation* (Method A and C) and *editing history* (Method B). Using human computation techniques, we can obtain *general* preference data generated by a large number of undefined crowds in an on-demand manner. Using editing history of a single target user, we can obtain preference data that are *personalized* to the user.

RELATED WORK

Computational Design of Functional Objects

In computer graphics community, many computational design methods have been investigated. Various functionality criteria have been formulated, such as the *fly-ability* of paper airplanes [9] and the *connect-ability* of 3D-printed connectors [5]. They provide computational supports for ensuring functional requirements or maximizing functionality. In contrast, our computation is for maximizing aesthetic preference.

Computational Aesthetics

Techniques for quantifying aesthetic preference have been investigated in various design domains. For example, Secord *et al.* [8] showed how the preference of viewing direction for 3D models can be computationally assessed. Some of such techniques have been applied to computational design systems. O'Donovan *et al.* [6] presented a computational design system for graphic layout that takes layout preference into consideration. These methods are based on domain-specific knowledge or heuristic techniques. Our methods also deal with aesthetic preference, but basically do not assume domain-specific properties, which allows our methods to be applicable to various domains.

Creativity Support Tools

Various creativity support tools for visual design (*e.g.,* [1]) have been investigated mainly in human-computer interaction community. Similar to ours, they support design exploration, but the goal is different from ours. In general, creativity support tools are for encouraging users to find unexpected and interesting designs, not aesthetically the best designs.

Table 1. The relationship among the three methods we present in my thesis.

	System Style	Purpose of Computation	Preference Data Origin	Details
Method A	Support system	Estimating $g(\cdot)$	Crowdsourcing	[3]
Method B	Support system	Estimating $g(\cdot)$	Editing history	[4]
Method C	Generative system	Finding $\mathbf{x}^*$	Crowdsourcing	–

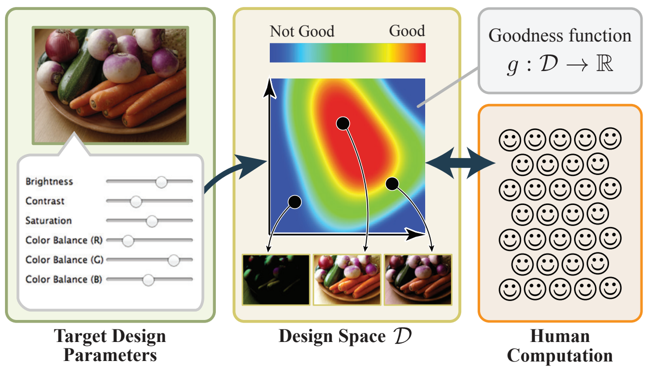


Figure 4. Overview of Method A. Here, we show the case of photo color enhancement as an example design domain. In this case, sliders such as "brightness" and "contrast" forms a high-dimensional design space $\mathcal{D}$. To analyze this space, we utilize crowdsourced human computation and obtain necessary preference data. Based on this data, we estimate the goodness function $g(\cdot)$ for this space.

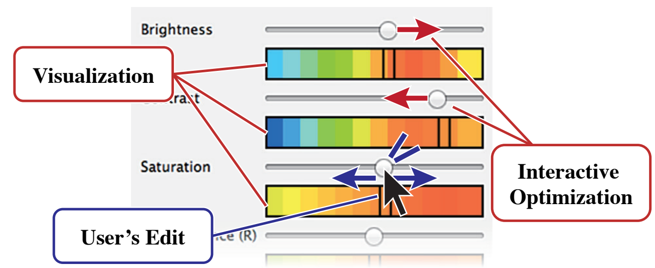


Figure 5. ***VisOpt Slider*****, a new slider interface for design exploration that can be used with our preference estimation techniques. The user can adjust each parameter effectively by the visualization ("Vis") of estimated preference near the slider and the interactive optimization ("Opt") that gently guides the current parameters to the estimated optimal direction.**

Human Computation and Crowdsourcing

Human computation and crowdsourcing (see [7] for detailed discussions of the definitions of these terms) are often used for gathering human-generated data that is difficult for machine to generate (*e.g.,* perceptual or semantic labels). For example, Gingold *et al.* [2] presented an image understanding method where necessary perceptual data for image analysis is gathered by human computation implemented on a crowdsourcing platform. We also utilize crowdsourced human computation to gather data about aesthetic preference in Method A and Method C.

METHOD DETAILS

Method A: Preference Estimation by Crowdsourcing

This method analyzes the target high-dimensional parameter space $\mathcal{D}$ to obtain a distribution of human preference, *i.e.,* the shape of the goodness function $g(\cdot)$. This method uses crowdsourced human computation to gather many pairwise comparisons between various parameter sets, sampled from the design space $\mathcal{D}$. Figure 4 illustrates an overview of this method. This estimated function enables a new user interface for facilitating user's design exploration, called *VisOpt Slider* (see Figure 5). This slider interface visualizes the estimated function values using color map (red means preferable and blue means not preferable), and also interactively optimizes slider values during user's exploration. With these computational guidance, the user can explore the design space efficiently. As for evaluation, we applied this method to four different visual design domains: photo color enhancement ($n = 6$), metallic shader adjustment ($n = 8$), camera and light positioning ($n = 8$), and facial expression modeling ($n = 53$).

Method B: Preference Estimation from Editing History

Method A models the goodness function $g(\cdot)$ using crowd-generated preference data, which means it does not reflect *personal* preference. In contrast, Method B learns the goodness function $g(\cdot)$ from editing history of a single user so that it can reflect personal preference. To use this technique in practical scenarios, we also present a new workflow, where, while the user conducts a repetitive task (*e.g.,* enhancing 50 photographs one by one), the system *implicitly* and *progressively* learns the user's preference, and progressively supports the user's design exploration. This means that the support by the system becomes more and more effective as the user proceeds the repetitive task. This concept was tested on photo color enhancement application. Figure 6 (a) illustrates this workflow (which we call *self-reinforcing color enhancement*), and Figure 6 (b) shows a screen capture of our proof-of-concept system, called *SelPh*. By using SelPh, the user can efficiently find the best parameter value $\mathbf{x}^*$ for each photograph.

Method C: Preference Maximization by Crowdsourcing

In contrast to the above two methods, where users interactively explore the design space with computational supports, Method C investigates how aesthetic preference can be computationally maximized *without* the user's manual interaction. To achieve this goal, we explore algorithms and implementations of *crowd-powered design optimizers*, which are solvers of the optimization of Equation 1 using crowdsourced human computation. While Method A and B estimate the shape of the goodness function $g(\cdot)$, Method C gathers human-generated preference data to proceed optimization algorithms (*i.e.,* the gradient descent method) and finally to find the optimal solution $\mathbf{x}^*$. Note that this research project is still on-going, but we would like to show an initial experimental result in Figure 7, where we applied the gradient descent method for photo color enhancement ($n = 6$). In this gradient descent procedure, the

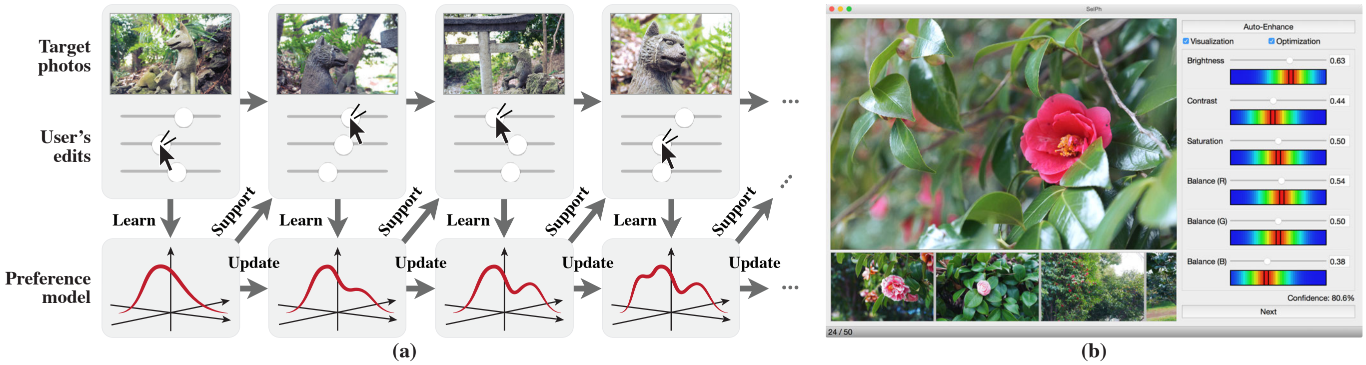


Figure 6. (a) Concept of *self-reinforcing color enhancement* (Method B). As more photos are enhanced by the user, the system *implicitly* and *progressively* learns the user's preferences and, as a result, the system is able to support the user in an increasingly effective manner. (b) A working prototype system, named *SelPh*. It has several user support functions enabled by the self-reinforcement, including VisOpt Slider and confidence-based adaptation.

Figure 7. An initial experimental result of Method C. To maximize aesthetic preference in photo color enhancement, we applied the gradient descent method in this experiment. Here, 15 crowd workers were assigned for each iteration. Note that the color of the photograph gradually improves as the iteration proceeds.

gradient of the goodness function, $\nabla g(\mathbf{x})$, is calculated via human computation.

SUMMARY

We presented three computational design methods for design tasks in which aesthetic preference is used for assessing the quality of design. More specifically, the goal of these methods is to find the optimal design parameter values. Two of the three methods are designed as support systems, where the systems provide computational supports for the user's interactive design exploration. The other one is designed as a generative system, where the system computationally explores the design space and automatically finds the optimal solution without requiring the user of this method to manually explore the design space. The key idea was that, for computationally dealing with aesthetic preference, it was effective to gather preference data by human computation techniques or from the user's editing history. We hope that these methods can provide fundamentals of future researches on computational aesthetic design.

REFERENCES

1. Benedetti, L., Winnemöller, H., Corsini, M., and Scopigno, R. Painting with bob: Assisted creativity for novices. In *Proc. UIST '14* (2014), 419–428.
2. Gingold, Y., Shamir, A., and Cohen-Or, D. Micro perceptual human computation for visual tasks. *ACM Trans. Graph. 31*, 5 (Sept. 2012), 119:1–119:12.
3. Koyama, Y., Sakamoto, D., and Igarashi, T. Crowd-powered parameter analysis for visual design exploration. In *Proc. UIST '14* (2014), 65–74.
4. Koyama, Y., Sakamoto, D., and Igarashi, T. Selph: Progressive learning and support of manual photo color enhancement. In *Proc. CHI '16* (2016), 2520–2532.
5. Koyama, Y., Sueda, S., Steinhardt, E., Igarashi, T., Shamir, A., and Matusik, W. Autoconnect: Computational design of 3d-printable connectors. *ACM Trans. Graph. 34*, 6 (Oct. 2015), 231:1–231:11.
6. O'Donovan, P., Agarwala, A., and Hertzmann, A. Designscape: Design with interactive layout suggestions. In *Proc. CHI '15* (2015), 1221–1224.
7. Quinn, A. J., and Bederson, B. B. Human computation: A survey and taxonomy of a growing field. In *Proc. CHI '11* (2011), 1403–1412.
8. Secord, A., Lu, J., Finkelstein, A., Singh, M., and Nealen, A. Perceptual models of viewpoint preference. *ACM Trans. Graph. 30*, 5 (Oct. 2011), 109:1–109:12.
9. Umetani, N., Koyama, Y., Schmidt, R., and Igarashi, T. Pteromys: Interactive design and optimization of free-formed free-flight model airplanes. *ACM Trans. Graph. 33*, 4 (July 2014), 65:1–65:10.