
Sketchplore: Sketch and Explore Layout Designs with an Optimiser

Kashyap Todi

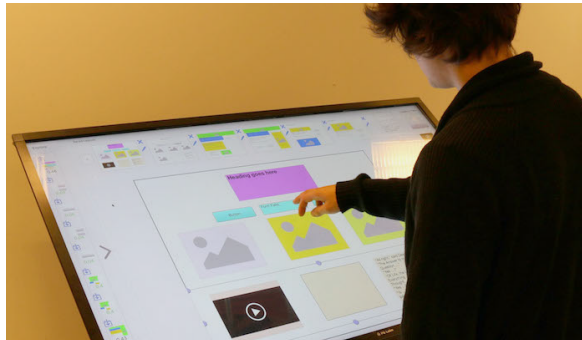
Hasselt University - tUL - iMinds
Diepenbeek, Belgium
kashyap.todi@uhasselt.be

Daryl Weir

Aalto University
Helsinki, Finland
daryl.weir@aalto.fi

Antti Oulasvirta

Aalto University
Helsinki, Finland
antti.oulasvirta@aalto.fi



Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the Owner/Author.
Copyright is held by the owner/author(s).
CHI'16 Extended Abstracts, May 07–12, 2016, San Jose, CA, USA
ACM 978-1-4503-4082-3/16/05.
<http://dx.doi.org/10.1145/2851581.2890236>

Abstract

Sketchploration is a novel concept of integrating real-time design optimisation to sketching tools, to enable interactive exploration of design alternatives. Although traditional optimisation methods can attack very complex design problems, their insistence on precise objectives contradicts sketching. *Sketchplorer* is an interactive sketching tool that uses a real-time layout optimiser. It automatically infers the designer's task and searches for local improvements, and global alternatives. Using several predictive models of user performance and perception, its suggestions steer designers toward more usable and aesthetic layouts. This facilitates the creative and problem-solving aspects of sketching, without requiring extensive intervention from the designer.

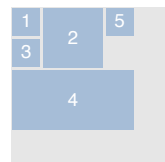
In our presentation, attendees will have the opportunity to interactively sketch and explore layout designs, on a large multitouch surface, with *Sketchplorer*.

Author Keywords

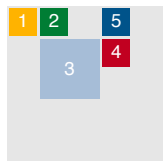
Sketching; Model-based optimisation; Visual Layouts.

ACM Classification Keywords

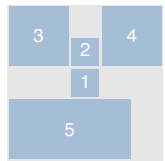
H.5.2. [Information Interfaces and Presentation]: User Interfaces



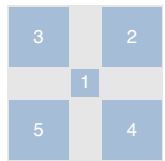
Visual clutter



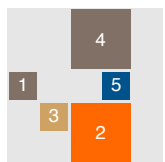
Visual search



Target acquisition



Grid quality



Colour harmony

Figure 2: Outcomes when optimising for a single objective. Sketchplorer combines these into multi-objective search.

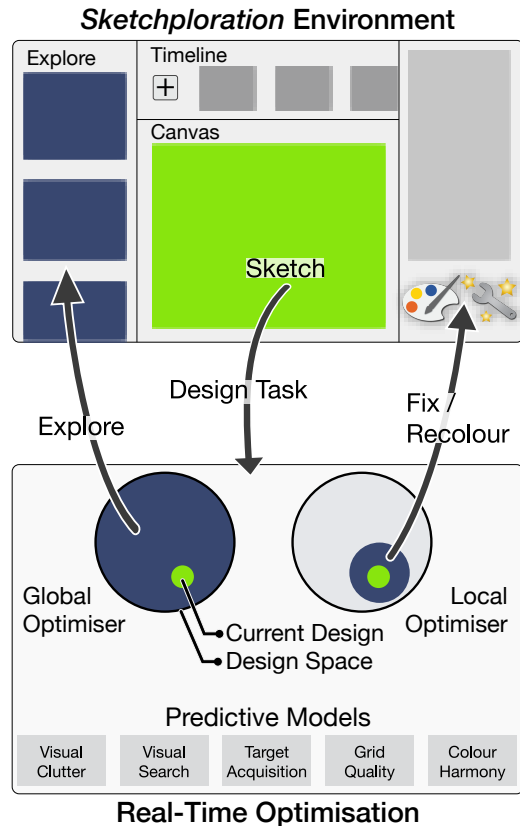


Figure 1: Sketchplorer supports interactive sketching using a novel approach to real-time layout optimisation. While the designer is sketching, the optimiser infers the design task, and searches for local improvements (*recolour*; *fix*) and global alternatives (*explore*). It uses predictive models of performance and perception to generate these suggestions.

Introduction

Sketching is a powerful visual thinking tool, employed by designers to explore different solutions to a given design problem. It is a very fluid process, and tends to be ambiguous and uncertain [2]. Optimisation, on the other hand, can efficiently attack very complex design problems, and systematically improve both usability and aesthetics of designs. Traditionally, optimisation is a rigid process, and demands *precise* input. These contradicting requirements inhibit designers from employing powerful optimisation techniques during sketching and ideation phases of design.

Sketchploration is a novel concept that brings real-time optimisation to visual design tools. It couples a designer and an optimiser in a new “serendipitous” way, supporting the natural flow of sketching and ideation. Technically, sketchploration (Figure 1) extends *model-based interface optimisation* [1, 3] to real-time design exploration under ill-specified and changing design goals. It relaxes the requirements for controlling an optimiser. As the designer sketches, the optimiser infers the design task automatically. It then searches for both local (small changes) and global (large changes) alternatives. Importantly, our optimiser deploys several predictive models of user performance and perception, adapted from literature. This allows it to make informed suggestions that “pull” designers toward *usable* and *aesthetic* designs.

Sketchplorer is a design tool that showcases the first investigation of this concept, focusing on *interactive layouts* familiar from GUIs, web pages, menus, and dialogs. It provides designers with a multitouch environment for sketching and exploring interactive layout designs.

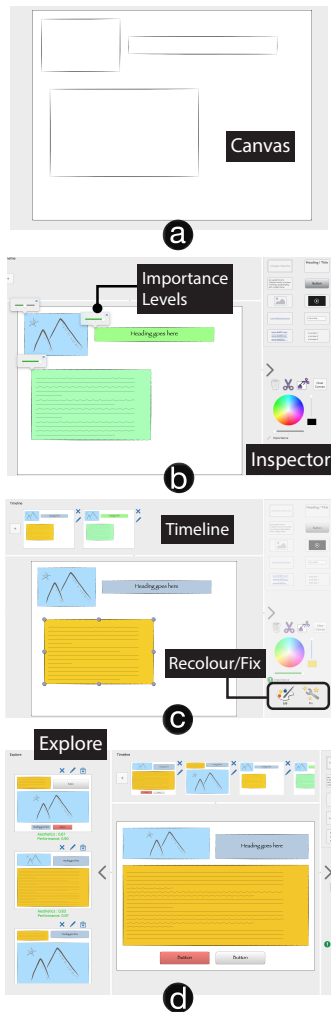


Figure 3: A Walkthrough. The designer (a) ambiguously sketches layout designs, and (b) refines and add details. They have immediate access to a timeline of saved designs. The optimiser suggests (c) **local** (fix and recolour) changes, and (d) **global** (explore) design alternatives.

Walkthrough: Designing a Blog Page

This walkthrough illustrates the use of Sketchplorer, our design tool based on the sketchploration concept, from a designer's perspective.

Sketching the initial layout: Sketchplorer initially presents the designer with an empty canvas. The designers starts sketching by creating a structure for her design (Figure 3a). Drawn elements can be moved around, or resized, at any time. She ambiguously sketches out boxes, serving as proxies for the elements of her blog page. Sketchplorer infers the design task, and the optimiser starts computing both local and global suggestions in the background.

Refining and adding details: An *inspector* panel sits on the right-edge of the display, and can be pulled out at any time. This can be used to specify details such as element type, colour, and importance (Figure 3b). The designer now takes a moment to add these details to some of the elements. For instance, she indicates that her page has a heading and a paragraph element, and marks them as being of high importance. Satisfied with the first version of the design, she taps the save ('+') button. This adds the current design to the designer's *timeline*, and provides a preview (Figure 3c). Saved designs can be retrieved, and edited, at any later time.

Making local changes (fix and recolour): While the designer sketches and refines, the system continuously streams the description of the current design to the optimiser. The local optimiser uses the current design as a starting point to suggest *fixes* and *recoloured* designs. Pulsating icons in the inspector panel indicate the availability of these suggestions (Figure 3c). The designer now refers to these suggestions, and realises that by using a recolouring suggestion, she can make the paragraph of text stand out. She selects this, and continues working on the sketch.

Exploring global alternatives: By abstracting from the current design, the global optimiser retrieves unique designs, and returns them to the designer. An *explore* panel, residing on the left edge of the display (Figure 3d), is periodically updated with these designs. The designer browses through the list of alternatives, and finds two interesting alternatives. She adds the first to her list of saved designs, and drags the second onto the canvas, to continue working on it.

In a short duration, the designer's collection of *saved layouts* is populated with several feasible alternatives—some sketched by the designer, and the others with the aid of the optimiser. While the above phases appear to be linear, in practice, sketching and exploration phases are intertwined.

Overview of Interactions

Sketchplorer is designed for a multitouch environment using a large display, and uses touch gestures for all controls.

Sketching and Refinement: Sketchplorer allows designers to either sketch ambiguous bounding boxes for layout elements, or pick out a specific element type and place it on the canvas. It takes care to accurately order every element on the canvas, without designer intervention. Each time an element is changed, the hierarchy is inferred, and the ordering dynamically adapted. Hierarchical groups of elements can be selected and manipulated at the same time. This inferred hierarchy is also essential for the internal representation of a layout in the optimiser. A colour picker allows designers to select the hue-saturation combination, and adjust the brightness. Double-tapping on an element reveals an in-place pop-over, and allows adding details without having to move to the inspector. An element's importance can be adjusted in the inspector panel. Alternatively, an overlay can be enabled, that displays the importance of every element, and allows batch adjustments (Figure 3b).

Saved versions and timeline: In Sketchplorer, the current design can be added to the timeline at any time. The timeline provides an overview of all saved designs, and any alternative design, or intermediate sketch, can be dragged back to canvas. This allows comparing designs, have an overview of the evolution of their designs, and iterate over sketches.

Integration with the optimiser: To the designer, the optimisation appears as a two-pronged approach, consisting of local and global optimisation. The local optimiser observes every change in the design. It suggests fixes that maintain the overall composition of the original sketch, but improve certain aspects. It also provides harmonious recolouring suggestions, which improve aesthetics and performance. In contrast, the global optimiser listens exclusively to changes in design tasks, and acts upon them. It abstracts away from exact details, allowing it to explore the entire design space, and generate unique designs. It performs exploration in real-time and periodically returns improved results, displayed to the designer in the expandable *explore* panel. The optimiser focuses on creating unique and improved solutions, not on refining or polishing a solution to make them perfect.

Predictive Models and Objective Function

Whereas previous work has used heuristics and focused on visual search and motor performance [1, 3], Sketchplorer addresses spatial and colour aspects of layouts. To achieve this, we use the following predictive models:

1. **Visual Clutter:** Rosenholtz Model
2. **Visual Search:** Kieras-Hornof Model
3. **Target Acquisition:** Fitts' Law
4. **Grid Quality:** Balinsky's Symmetry Measure
5. **Colour Harmony:** Cohen's Harmonic Colour Sets

Figure 2 illustrates optimisation results for individual objectives, for a sample task with five elements. We define a multi-objective task where we seek to minimise a weighted combination of the outputs of these models:

$$U = \sum_{i=1}^5 w_i S_i, \quad (1)$$

where the weights w_i sum to 1 and the individual objectives S_i are normalised to the range $[0, 1]$.

Conclusion

By integrating real-time optimisation into a sketching tool, *sketchploration* opens up opportunities for designers to efficiently explore alternative designs, and improve the usability and aesthetics of outcome designs. We have conducted user studies to validate our optimisation technique, and the interactions supported by Sketchplorer. The results from these are positive, but a full discussion is outside the scope of this abstract.

References

- [1] Gilles Bailly, Antti Oulasvirta, Timo Kötzing, and Sabrina Hoppe. 2013. MenuOptimizer: Interactive Optimization of Menu Systems. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology (UIST '13)*. ACM, New York, NY, USA, 331–342. DOI: <http://dx.doi.org/10.1145/2501988.2502024>
- [2] Bill Buxton. 2007. *Sketching User Experiences: Getting the Design Right and the Right Design*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA.
- [3] Krzysztof Gajos and Daniel S. Weld. 2004. SUPPLE: Automatically Generating User Interfaces. In *Proceedings of the 9th International Conference on Intelligent User Interfaces (IUI '04)*. ACM, New York, NY, USA, 93–100. DOI: <http://dx.doi.org/10.1145/964442.964461>