Designing and Evaluating a Responsivity Tool for Flex Studio

Kai Zheng
CS, ETH Zürich

Cashen Adkins
CS, ETH Zürich

Tim Kluser
CS, ETH Zürich

Karl Robert Kristenprun
CS, ETH Zürich

Alexander Zank
CS, ETH Zürich

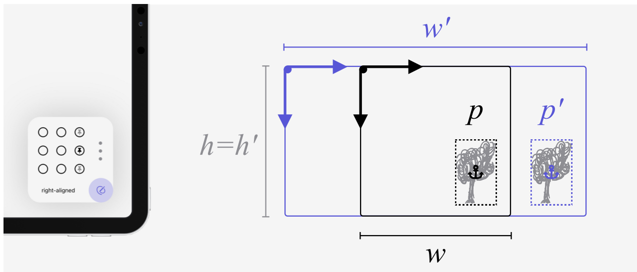

Figure 1: Layer behavior when selecting any of the right-aligned pins.

1 Introduction

Flex Studio is a concept application streamlining the workflow of cartoon and webcomic artists. Instead of focusing on full-page layouts, artists develop one panel at a time, letting their imagination run free while hinting to the system how their drawings should be positioned at a wide range of aspect ratios. Our auto-paneling technology uses this information to export single-panel, print, and full-page versions of their work tailored to each platform, alleviating the need for intensive manual composition work and increasing the reach of their art.

Given how essential sensible layout constraints are to *Flex Studio*'s core offering, setting and validating them is a primary focus of the panel editor. Artists can flexibly create and preview their panels at aspect ratios ranging from 2:1 (wide) to 1:2 (tall). By default, every layer is positioned absolutely around the center. Tapping the magic lasso icon reveals an interface that allows the artist to set positional constraints: the **responsivity tool**.

1.1 The Interface Variants

In this study, we examine the performance, accuracy, and perceived quality of two responsivity tool variants:

1.1.1 Indirect — Pinning

For each layer, the artist can optionally select one of nine pin locations on a corner sheet. A pin neighboring an edge specifies absolute positioning to it. A centered pin preserves the relative position of the layer within the corresponding dimension.

More precisely: With $p = (p_x, p_y)$ and p' denoting the center of the layer's bounding box and w, w', h, and h' denoting the width and height of the panel at two different aspect ratios, we ensure

- $p'_x = p_x$ for left aligned pins,
- $w' - p'_x = w - p_x$ for right aligned pins,
- $p'_x/w' = p_x/w$ for horizontally centered pins,
- $p'_y = p_y$ for top aligned pins,
- $h' - p'_y = h - p_y$ for bottom aligned pins, and
- $p'_y/h' = p_y/h$ for vertically centered pins.

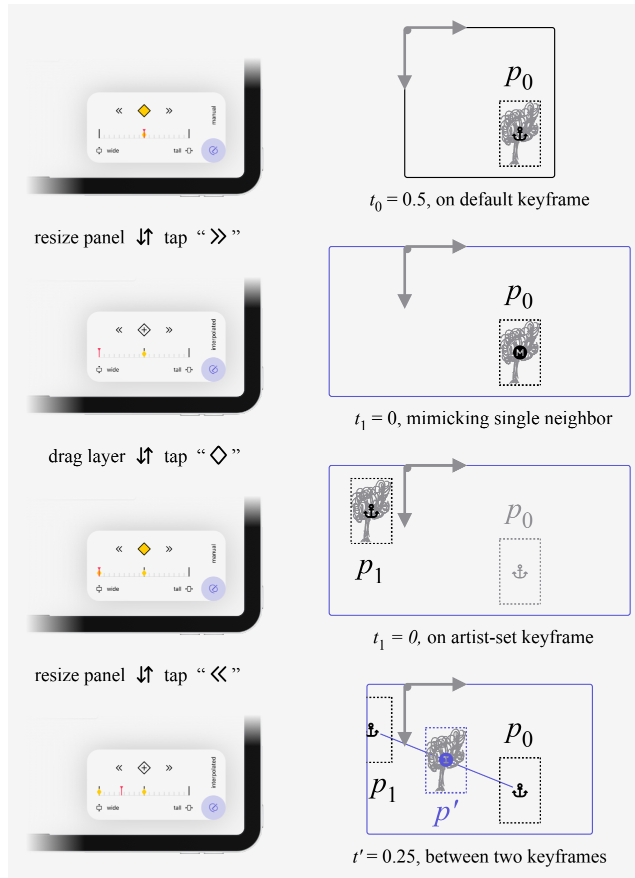

Figure 2: A full keyframe workflow.

1.1.2 Direct — Keyframing

At any aspect ratio, the artist can select and directly drag layers to their desired position, implicitly setting keyframes in the process. Flex Studio linearly interpolates the position of every layer between keyframes with respect to the currently set aspect progression (a value $\in [0,1]$ linearizing the aspect ratio).

More precisely: For a fixed maximum aspect ratio $b > 1$, define the function mapping aspect ratio to aspect progression as

$$prog : [b^{-1}, b] \to [0,1]$$
$$r \mapsto \begin{cases} \frac{b-r}{2(b-1)} & \text{for } r \in [1,b] \\ \frac{(b-2)r+1}{2(b-1)r} & \text{for } r \in [b^{-1},1[\end{cases}$$

Note that by design, $prog(b) = 0$, $prog(b^{-1}) = 1$, and $prog(1) = 0.5$. In other words: an aspect progression of 0 corresponds to the widest, 1 to the tallest, and 0.5 to the square aspect ratio.

Given a progression target t' and its neighboring keyframes as progression-position pairs (t_0, p_0) and (t_1, p_1) with $t_0 \le t' < t_1$, we interpolate the target position

$$p' = p_0 + \frac{t' - t_0}{t_1 - t_0}(p_1 - p_0)$$

If only one neighbor $(\circ, p_n)$ exists, we set $p' = p_n$.

1.1.3 An A Priori Comparison

In both interface variants, artists resize the panel directly by dragging a frame outlet across the screen, review per-layer responsivity data at a glance in the corner sheet, and unset constraints by tapping a button in the corner sheet.

What differs is how artists set constraints. The indirect variant uses buttons with intuitive semantics in the corner sheet while the direct variant has the artist directly drag the layer at their desired position in the panel.

Notably, the direct interface is more expressive since the artist isn't limited to a discrete set of possible constraints covering the aspect ratio extremas but rather is free to set as many as desired on a continuous spectrum.

2 Study Design

We want the responsivity tool to

1. work quickly and easily, i.e., require minimal manual input,
2. behave predicatably, i.e., extrapolate from the user's guidance like another human would, and
3. be flexible enough to achieve a wide range of results, i.e., allow the artist to express many layer alignments.

Therefore, in designing the user study, we chose a hybrid *within-subjects* approach: We record 1. quantitative data in a comparative closed task and 2. qualitative data about the overall usage experience (including the open training).

2.1 Apparatus

Participants work on a 12.9-inch iPad Pro running a tailored build of Flex Studio on iPadOS 16.1.1 that emits function- or coordinate-annotated log messages for every responsivity-related interaction and "touches began" event.

During the open training, participants are handed an Apple Pencil, allowing them to draw on custom layers. Before entering a closed task, this tool is taken away to alleviate input device-related performance differences.

During the closed task, participants receive a color printout of their target, showing the desired panel behavior at the 2:1, 1:1, and 1:2 aspect ratios.

The researcher records the screen and collects logs on a tethered laptop computer. When tasked to fill out sections in the questionnaire, the laptop is turned around, and the participant is presented with an online form.

We collect necessary consent in paper form and leave out identifying information in the questionnaire, making a posteriori linking impossible.

2.2 Independent Variables

Per participant, we test two conditions, described as tuples $(v_1 :: V1, v_2 :: V2)$ of independent variables

V1 (Tested Interface Variant) $\in \{$
  I (indirect – pinning),
  D (direct – keyframing)
$\}$

V2 (Task Type) $\in \{$
  $C1$ (making "Cozy Campfire" responsive given a target),
  $C2$ (making "Plumber" responsive given a target)
$\}$

2.3 Dependent Variables

We measure

T (Task Completion Time, sec) $\in \mathbb{R}$ interval
timer operated by participant

N (Number of Drags and Gestures) $\in \mathbb{N}$ interval
"touches began" events

E (Number of Erroneous Inputs) $\in \mathbb{N}$ interval
"touches began" events not directly followed by a functional event

X (Modified NASA TLX Raw Score) $\in \{10, 11, \ldots, 100\}$ interval
unweighted, skipping temporal demand, with individual scales from 1 *to* 10

R (Perceived Restrictivity) $\in \{1, 2, \ldots, 10\}$ Likert, ordinal
"Considering the five minutes you were given at the beginning and the task you just completed, how restrictive is the tool, or how little do you think the tool allows you to express your creativity? Is it restrictive or extensive, disabling or enabling?"

C (Perceived Complexity) $\in \{1, 2, \ldots, 10\}$ Likert, ordinal
"Considering your time with the tool so far, how hard is it to use? How complex, difficult, unintuitive versus simple, effortless, straightforward was the use of the tool when accomplishing your goal?"

Additionally, we record qualitative assessments in form of full-text replies at the end of the study.

2.4 Null Hypotheses

(HT) There will be no effect of interface variant (V1) on the time (T) required to fulfill a closed task (V2).

(HN) There will be no effect of interface variant (V1) on the number of gestures (N) required to fulfill a closed task (V2).

(HE) There will be no effect of interface variant (V1) on the input accuracy (E) of the responsivity process in a closed task (V2).

(HX) There will be no effect of interface variant (V1) on the Raw TLX score (X) awarded by the users.

(HR) There will be no effect of interface variant (V1) on the restrictivity (R) perceived by the user.

(HC) There will be no effect of interface variant (V1) on the complexity (C) perceived by the user.

2.5 Procedure

After introducing participants to the goal of Flex Studio and receiving the necessary consent, the researcher explains the procedure:

"We'll run two rounds of tests, one for every interface variant. In every round, we'll first introduce you to it by providing a simple demo, and will grant you some time to play around with it in your own drawing. Once you're familiar with it, we'll ask you to make an artist-provided panel responsive according to a reference."

The researcher's demo has them draw a scene consisting of a tree, a person, and a sun and make it responsive in some predefined way.

Before a closed task is initiated, the researcher explains:

"Here's what you are tasked to recreate (*pointing to the reference printout*). We'll start our stopwatch once you press this button, and will stay silent until you press the stop button. Note that it is possible to exactly recreate the reference behavior but not required. **Press stop when you have achieved a reasonably accurate result.**"

Denoting open training with $O_{v::V1}$, the training-condition sequence of the n-th participant is given by

$$n = 0 \mod 4 \implies O_I \to (I, C1) \to O_D \to (D, C1)$$
$$n = 1 \mod 4 \implies O_I \to (I, C2) \to O_D \to (D, C2)$$
$$n = 2 \mod 4 \implies O_D \to (D, C1) \to O_I \to (I, C1)$$
$$n = 3 \mod 4 \implies O_D \to (D, C2) \to O_I \to (I, C2)$$

2.6 Participants

With our setup, the study requires $4k, k \in \mathbb{N}^+$ participants. The pool whose results we present in the next section consists of academics between the age of 20 and 24, one identifying as a woman and seven identifying as men.

3 out of 8 participants indicated having had prior experience with keyframe-based tools (e.g., video editing software) and 3 out of 8 participants said the same for pin-based tools (e.g., mid-fi prototyping tools).

3 Results

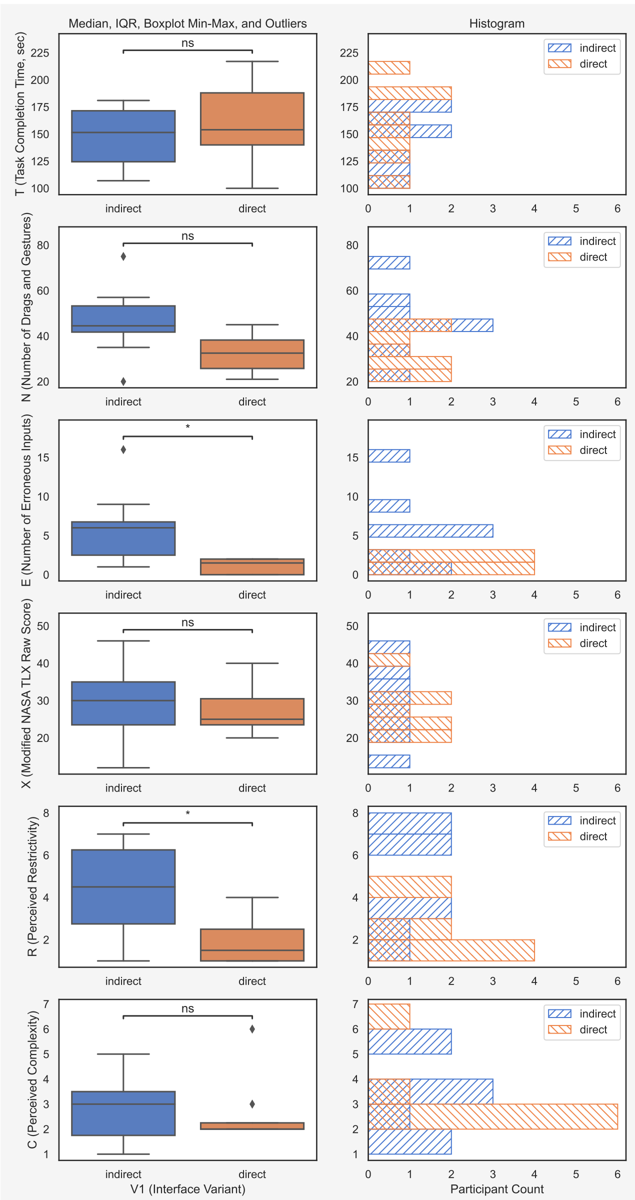

Figure 3: A comprehensive statistical overview.

3.1 Quantitative

3.1.1 Completion Time

To compare the effect of the interface variant (V1) on completion time (T), we conducted a paired samples t-test (Shapiro-wilk $p = 0.96 > 0.05$ and Levene's $p = 0.51 > 0.05, \alpha = 0.05$).

With the direct interface, participants on average needed 159 seconds to complete the task. With the indirect interface, participants on average needed 148 seconds to complete the task.

The mean difference between the two groups was not statistically significant; $t(7) = 2.37, p = 0.54$. Therefore we failed to reject HT and we cannot conclude that one interface variant is faster to use than the other.

3.1.2 Number of Gestures

To compare the effect of the interface variant (V1) on the number of gestures (N), we conducted a paired samples t-test (Shapiro-wilk $p = 0.34 > 0.05$ and Levene's $p = 0.44 > 0.05, \alpha = 0.05$).

With the direct interface, participants on average required 32 gestures to complete the task. With the indirect interface, participants on average required 47 gestures to complete the task.

The mean difference between the two groups was not statistically significant; $t(7) = 2.37, p = 0.074$. Therefore we failed to reject HN and we cannot conclude that one interface variant takes fewer gestures than the other.

3.1.3 Erroneous Inputs

To compare the effect of the interface variant (V1) on the number of erroneous inputs (E), we conducted a Wilcoxon signed-rank test (Shapiro-wilk $p = 0.001 \leq 0.05$ and Levene's $p = 0.07 > 0.05, \alpha = 0.05$).

With the direct interface, participants on average had 1.125 erroneous inputs before completing the task. With the indirect interface, participants on average had 6 erroneous inputs before completing the task.

The mean difference between the two groups was ***statistically significant***; $p = 0.0078$.

These results indicate that the direct interface was less prone to erroneous inputs. HE is rejected.

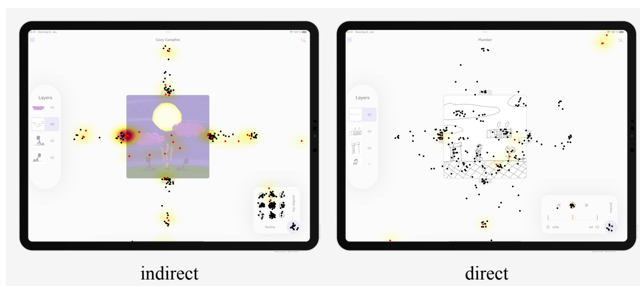

Figure 4: Cumulative touches by interface variant with a heatmap of erroneous touches (red) overlaid.

3.1.4 Raw TLX Score

To compare the effect of the interface variant V1 on the workload, measured through the raw TLX score X, we conducted a paired samples t-test (Shapiro-wilk $p = 0.92 > 0.05$ and Levene's $p = 0.25 > 0.05, \alpha = 0.05$).

With the direct interface, participants on average report a raw TLX score of 27.25 out of 100 after completing the task. With the indirect interface, participants on average report a raw TLX score of 29.5 out of 100 after completing the task.

The mean difference between the two groups was not statistically significant; $t(7) = 2.37, p = 0.50$. Therefore we failed to reject HX and we cannot conclude that one interface variant results in a lower raw TLX score, i.e., less workload.

3.1.5 Restrictivity

To compare the effect of the interface variant (V1) on the restrictivity (R), we conducted a Wilcoxon signed-rank test (Shapiro-wilk $p = 0.013 \leq 0.05$ and Levene's $p = 0.018 \leq 0.05, \alpha = 0.05$).

With the direct interface, participants on average report a perceived restrictivity of 2.0 after completing the task. With the indirect interface, participants on average report a perceived restrictivity of 4.375 after completing the task.

The mean difference between the two groups was ***statistically significant***; $p = 0.034$

These results indicate that the direct interface is perceived as less restrictive. HR can be rejected. This is also supported by the qualitative feedback, which shows that participants prefer having the ability to move the layers wherever they want over only having 9 possible set positions.

3.1.6 Complexity

To compare the effect of the interface variant (V1) on the Complexity (C), we conducted a Wilcoxon signed-rank test (Shapiro-wilk $p = 0.009 \leq 0.05$ and Levene's $p = 0.425 > 0.05, \alpha = 0.05$).

With the direct interface, participants on average report a perceived complexity of 2.625 after completing the task. With the indirect interface, participants on average report a perceived complexity of 2.875 after completing the task.

The mean difference between the two groups was statistically insignificant; $p = 0.034$. Therefore we failed to reject HC and we cannot conclude that one interface variant is more complex to use than the other.

3.2 Qualitative

6 out of 8 participants preferred the direct interface, while the other 2 participants saw advantages in both interfaces, depending on the user's experience.

To quote one participant, "[it] depends on what I am trying to do. If I am tasked to [make someone else's work responsive], the indirect method is simpler. [...] As an artist I'd prefer the direct method since it is a lot less restrictive [when it comes to] where I can place my [layers.]"

4 participants thought that the indirect interface was easier to use and less mentally demanding, one of them mentioned that the difference is negligible.

Surprisingly, one participant noted that the direct interface was easier to use since they didn't need to test its behaviour by changing the aspect ratio beforehand: They "knew how it'd be behaving."

We also received many suggestions on how to improve the design of both methods: grid alignment, layer multi-select, and active markers, to name a few.

4 Discussion

We had initially expected the indirect interface to be significantly faster since participants can rest one hand on the corner sheet, resulting in shorter distances traveled with their fingers and, according to Fitt's law, therefore, shorter movement time. Consequently, we were surprised that there is no statistically significant difference in completion time between both interfaces. Qualitative feedback indicates that this might be due to a "trial and error" approach with the indirect interface. Participants play around with pin settings, testing their behavior by changing the aspect ratio rather than mentally pre-visualizing it, and use one hand to operate the tool.

Testing indicates that erroneous inputs are more common with the indirect interface. Notably, most happen around the frame outlets and not around the bottom sheet, leading us to suspect that they stem from resizing attempts. The direct variant interprets touches within the panel as move events and thus classifies them as correct. Extending the draggable area into the panel could heavily reduce this number in future builds and make the comparison fairer.

Overall, our data suggest that the direct interface is the superior one. Performance and workload do not differ dramatically, as seen by the dependent variables T, N, X, and C. However, we can confidently conclude that the higher expressiveness of the direct interface, or conversely the higher perceived restrictivity R of the indirect interface, are significant and were crucial in the participant's choice of preference.

5 Limitations

Following the previous discussion, the way we test and compare both input methods for erroneous inputs is flawed. More precisely: E is biased towards the direct input method in its current form because, in the latter, we do not classify touches with conflicting intention as erroneous.

Furthermore, the task design might have made the direct interface perform better since the task goal was a clear image of where every drawing should be. Therefore, dragging the objects to the proper positions might be intuitively more straightforward than tapping the pin buttons since one does not need to visualize how things should interact for different aspect ratios. Instead, one only needs to copy positions for the three aspect ratios used in the task.

Considering the age, gender, and occupation of our participants, our samples are pretty limited. Professional cartoon artists in the real world might range from young teens to people of any age. Our participants, all in their young twenties, do represent our primary target audience to some degree. However, a considerable share of the target audience is not included. Furthermore, with only one female participant, people identifying as male are over-represented in our study. Lastly, our study comprises students from academia, mostly studying Computer Science, which might have more experience with Software and App-Interfaces than people who do not study Computer Science. A more random assortment of participants of different ages, genders, and backgrounds would have been desirable.

6 Future Work

As previously mentioned, it would be interesting to see how a combined interface with keyframing and pinning compares to the individual interfaces. This is left for future research.

One could also change the task goal from having to recreate a set reference to asking the participants to make a given cartoon responsive however they wish. Thus, it's not simply about copying positions and trying out different pins with trial and error until the cartoon matches the reference, but about using the tool the like it is supposed to be used in the real world. This might incentive wanting to understand the exact behavior of both tools since intermediate aspect ratios are also considered, not just the three reference aspect ratios.

In a future study, one should also rethink the testing for erroneous inputs. Instead of counting gestures without associated events, one could record the finger position and taps of the participants and manually check the video material for adjusted/repeated gestures due to erroneous inputs.

7 Conclusion

We designed and evaluated pinning- and keyframe-based interface variants for Flex Studio's responsivity tool, allowing artists to make their creations more adaptive.

While our empirical study did not find significant performance differences, it did deem the direct interface more expressive. From qualitative results, we learned that the level of control offered by keyframes was a vital factor in participants' preferences, but the preciseness of pins is also desirable.

As a next step, we suggest exploring the idea of using both interfaces simultaneously to combine their respective strengths.