Artistic resizing : a technique for rich scale-sensitive vector graphics

Pierre Dragicevic, Stéphane Chatty, David Thevenin, Jean-Luc Vinot

Artistic Resizing: A Technique for Rich Scale-Sensitive Vector Graphics

Pierre Dragicevic
University of Toronto
dragice@dgp.toronto.edu

Stéphane Chatty
IntuiLab
chatty@intuilab.com

David Thevenin
IntuiLab
thevenin@intuilab.com

Jean-Luc Vinot
DSNA / DTI / R&D
vinot@cena.fr

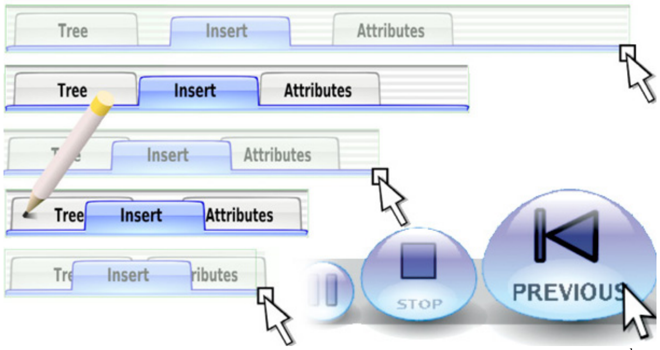

Figure 1: On the left, vector graphics with non-uniform resizing: the 2nd and 4th variants have been drawn in Illustrator, the other ones are interpolated. On the bottom right, a dock using Artistic Resizing.

Visual quality is given more and more importance in graphical user interfaces. This points out the need for new methods and tools to effectively involve graphic designers into GUI design and development teams. As a first step in this direction, IntuiKit [Chatty et al. 2004] showed how rich vector graphics could be exploited to combine the visual expressivity of authoring tools such as Adobe Illustrator with the behavioral expressivity of programming toolkits.

Converting static visuals into fully dynamic user interfaces still raises a number of issues. One of them is resizing. The only known technique for resizing vector graphics is uniform scaling. Such a method does not guarantee visual readability for all sizes and aspect ratios. Besides, user interfaces commonly show non-uniform scaling behaviors. When a button is resized for example, invariants are maintained such as label size and centering. Such behavior is traditionally modeled as a set of constraints which can be either coded by a programmer (as it is the case in most toolkits) or inputted to a constraint-solving system. But specifying constraints is hard, if not impossible, on rich and arbitrarily-structured vector graphics, even with declarative or visual approaches.

In spite of their familiarity with media more stable than interactive displays, graphic designers are nevertheless acquainted with the notion of adaptation. For example, a logotype can be used on different media, at different sizes, or on different color backgrounds. Designers often provide several variants of their work to explain how they should be adapted to different contexts. Artistic Resizing [Dragicevic et al. 2005] builds on that understanding by allowing graphic designers to provide examples of graphics at different key sizes and let the system infer their general resizing behavior.

In a typical scenario, the designer first uses her drawing tool to provide the static graphics of a GUI object or icon at an arbitrary size. She then specifies its resizing behavior by adding visual variants. This is done by duplicating a variant on the same document then moving and scaling its constituents (Figure 1, left). The graphic document can be loaded into the IntuiKit interpreter and tested at any time, allowing it to be iteratively enriched until the designer is satisfied by the visual appearance at intermediate and extreme sizes and aspect ratios. Animated examples can be found at `http://www.intuilab.com/artresize`

Artistic Resizing's inference algorithm is based on a simple bivariate geometry interpolation technique we call *orthogonal interpolation*. It requires a set of graphic groups (variants) that share the same structure and have a bounding box. Each local affine transformation is extracted and interpolated independently from the others. The *first line* of the matrix is linearly interpolated along the *width* of the bounding boxes, whereas the *second line* is interpolated along their *height*. The assumption is that horizontal (resp. vertical) resizing only results in horizontal (resp. vertical) motions, i.e. translations, scales and shears. On more than two examples, two monovariate piecewise linear interpolations can thus be applied, eliminating the need for multivariate techniques.

Although simple, orthogonal interpolation has a number of useful properties. First, its results are independent from the graphics structure, provided that each tree path contains at most one varying transformation (graphics can be normalized to conform to this rule). Second, interesting geometrical properties are preserved on interpolated graphics, such as algebraic measures (allowing the specification of fixed margins and alignments), relative ratios (allowing centering), contact and parallelism.

Whereas advanced image interpolation techniques have been proposed in a variety of domains, Artistic Resizing shows that a minimalist approach can successfully serve the purposes of GUI resizing. Its properties differ from those of rigid interpolation schemes used in 2D and 3D computer animation [Shoemake et al. 1992], reflecting different requirements: GUI resizing is bivariate, non-rigid, axis-dependant and rarely involves rotations.

Artistic Resizing builds upon a reasonable trade-off between power and simplicity. Because it does not rely on an extensive search for invariants [Kurlander et al. 1993], it is not subject to combinatorial explosion and is efficient even on extremely complex graphics. In contrast with most by-example systems, it is predictable and does not require the user to prune unintended rules. Artistic Resizing additionally allows the expression of more subtle, non-linear resizing behaviors and can be easily combined with higher-level layout models (Figure 1, right).

Acknowledgements

Thanks to Michel Beaudouin-Lafon, Sylvie Athènes, Yves Rinato, Stéphane Conversy, Sandra Basnyat, Frédéric Jourdan, Céline Schlienger and Alexandre Lemort for their help.

References

CHATTY, S., SIRE, S., VINOT, J., LECOANET, P., LEMORT, A., MERTZ, C. Revisiting visual interface programming: creating GUI tools for designers and programmers. In *Proc.of UIST '04*. ACM Press, 267-276.

DRAGICEVIC, P., CHATTY, S., THEVENIN, D., VINOT, J. Artistic resizing: a technique for rich scale-sensitive vector graphics. In *Proc. of UIST '05*. ACM Press, 201-210.

KURLANDER, D., FEINER, S. Inferring constraints from multiple snapshots. *ACM Trans. Graph.* 12, 4 (Oct. 1993), 277-304.

SHOEMAKE, K. DUFF, T. Matrix animation and polar decomposition. *In Proc. Graphics interface '92*. Morgan Kaufmann Publishers, 258-264.