From Visual Multiplexing to Zebra Maps

Yiwen Xing*
University of Oxford

Afrah Farea†
Istanbul Technical University

Saiful Khan‡
Science and Technology Facilities Council

Min Chen§
University of Oxford

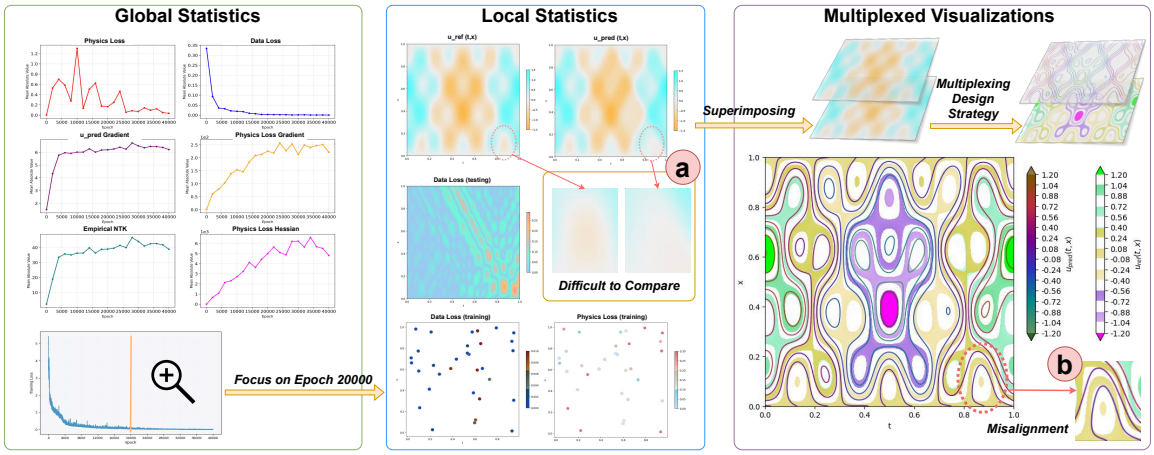


Figure 1: A motivating application of visual multiplexing in PDE machine learning. Domain experts often begin with global statistics (*left*) and then inspect multiple scalar fields from a selected iteration (*center*). Comparing spatially aligned fields through separate heatmaps can be difficult ⓐ, motivating the use of visual multiplexing. The example on the right shows a contour-over-Zebra Map representation, where local boundary mismatches ⓑ reveal discrepancies between the reference and predicted fields.

Abstract

Visual multiplexing refers to the simultaneous presentation of multiple pieces of information associated with the same spatial location [2]. By leveraging human perceptual and cognitive capabilities, viewers can often de-multiplex multiple visual signals from a shared representation. Despite its widespread presence in visualization, the deliberate use of multiplexing as a design strategy remains underexplored. This poster presents an ongoing investigation of multiplexing through a case study in machine learning workflows for solving partial differential equations (PDEs). In this application context, researchers frequently compare multiple spatially aligned scalar fields, making it difficult to identify local relationships using conventional juxtaposed visualizations. Building upon a collection of multiplexed representations, we explore layered designs that combine heatmaps, contours, glyphs, and a novel Zebra Map representation. Zebra Maps introduce contour-like value boundaries into dense scalar fields while preserving heatmap patterns, enabling more effective comparison between aligned fields. Through collaboration with PDE experts, we continue to refine multiplexing strategies and are developing an interactive visualization environment for constructing and exploring multiplexed visualizations.

Index Terms: Visual multiplexing, scalar fields, co-design.

1 Introduction

Humans routinely perceive and interpret multiple pieces of information associated with the same spatial location. Through perceptual grouping, prior knowledge, and other Gestalt-inspired mechanisms, viewers can often recover several visual signals from a shared representation [2, 5]. This capability underpins many common visualization designs, where multiple visual attributes, symbols, or layers coexist within the same visual space.

Visual multiplexing provides a conceptual framework for understanding such phenomena [2]. Rather than treating visual representations as isolated encodings, multiplexing emphasizes how multiple information channels can be integrated into a shared view and subsequently de-multiplexed by human observers. Although multiplexing is widespread in visualization practice, its deliberate use as a design strategy remains relatively underexplored.

In this work, we investigate visual multiplexing through a case study in machine learning workflows for solving partial differential equations (PDEs). Researchers in this domain routinely examine global statistics that summarize training progress across iterations, together with numerous scalar fields defined over the same spatial-temporal domain, including reference solutions, model predictions, residuals, loss terms, sampling patterns, and training signals [3]. Figure 1 (*left* and *center*) illustrates a subset of the global and local statistics derived from real PDE machine learning workflows.

These data fields exhibit different characteristics. Some are relatively sparse, such as losses defined only at sampled locations, whereas others are dense scalar fields with values defined throughout the domain, such as reference and predicted solutions. Multiplexing sparse fields with dense fields is often straightforward, for example by overlaying sampled points on a heatmap. In contrast, multiplexing two dense fields remains challenging. A common task in PDE machine learning is the local comparison of reference and predicted solutions, both of which are typically represented as dense heatmaps. Motivated by this challenge, we worked closely with domain experts to explore different multiplexing strategies. In this poster, we focus on the problem of comparing two dense scalar fields and introduce Zebra Maps, a new representation designed for layered multiplexing. Additional multiplexing strategies can be found in our ongoing work [6].

*e-mail: yiwen.xing@eng.ox.ac.uk
†e-mail: afrah.farea@outlook.com
‡e-mail: saiful.khan@stfc.ac.uk
§e-mail: min.chen@oerc.ox.ac.uk

2 Multiplexing in Practice

Visualization researchers have long explored methods for integrating multiple information sources within a shared visual representation. Composite visualizations provide a broad design space for combining multiple visual representations within a shared visualization, including strategies such as juxtaposition, superposition, and other forms of visual composition [4, 1]. Visual multiplexing provides a complementary perspective on these representations. Rather than viewing them solely as visual compositions, multiplexing emphasizes the human ability to de-multiplex multiple visual signals from a shared representation [2]. Building upon ideas from both composite visualization and visual multiplexing, we investigate layered strategies for integrating multiple scalar fields within a shared spatial domain, enabling direct point-to-point comparison between corresponding locations.

2.1 What Can Be a Layer?

To construct a layered multiplexing visualization, we first need to identify what visual representations can serve as basic layers. Figure 2 ⓐ–ⓖ summarizes seven primitive layer design options, including heatmap, zebra map, contour plot, size glyph, monochromatic dots, color glyph, and predefined multivariate glyph. The zebra map (Figure. 2 ⓑ) will be introduced later as a new representation developed in this work.

These primitives differ in information density, visual abstraction, and perceptual characteristics. Dense representations, such as heatmaps, preserve local scalar-field variation, whereas contours and glyphs provide progressively sparser abstractions. Rather than viewing these representations as isolated visualization techniques, we consider them as reusable building blocks that can be combined into layered multiplexed visualizations.

The examples shown here are intentionally context-free. In practice, each primitive may have many context-dependent variations, including different colormaps and encodings.

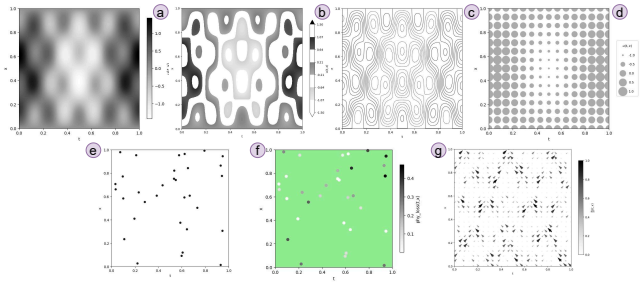


Figure 2: Seven primitives: (a) **heatmap**; (b) **zebra map**; (c) **contour**; (d) **size glyph**; (e) **monochromic dots**; (f) **color glyph**; and (g) **pre-defined glyph**. *(d)∼(f) belong to the **univariate glyph** family and can be applied either on a grid or at sampled locations.*

2.2 From Heatmaps to Zebra Maps

When constructing layered multiplexed visualizations, layer ordering plays an important role. In general, sparse representations are placed above denser representations to reduce occlusion while preserving information from the underlying layers. Human observers can often recover partially occluded structures through perceptual grouping and other Gestalt-inspired mechanisms[2, 5], allowing multiple layers to coexist within a shared visual space.

In PDE machine learning workflows, reference and predicted solutions are typically represented as dense scalar fields. Directly overlaying two heatmaps often results in severe visual interference, making the lower layer difficult to interpret. Replacing one heatmap with contour lines provides a practical multiplexing solution. Contours summarize a scalar field through iso-values and can be overlaid on a heatmap while maintaining the visibility of both fields.

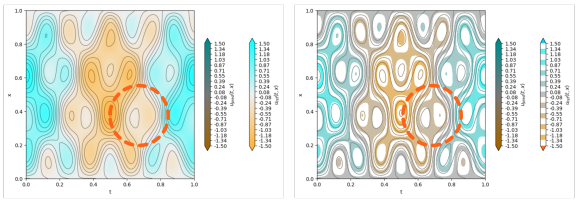


Figure 3: Comparing contour-over-heatmap (left) and contour-over-Zebra Map (right). Zebra Map boundaries share the same iso-value levels as contour lines, making local mismatches easier to identify.

However, discussions with domain experts revealed a remaining challenge. Although contour-over-heatmap representations allow both fields to be viewed simultaneously, contour lines provide explicit value boundaries whereas heatmaps communicate values primarily through color variation. Comparing corresponding values therefore often requires repeatedly referencing the color bar.

To bridge the gap between dense heatmaps and sparse contour representations, we introduce Zebra Maps. As shown in Figure 3, Zebra Maps augment conventional heatmaps by inserting white masks at predefined iso-value levels. The resulting boundaries share the same iso-value levels as contour representations while preserving the original color patterns of the field. When contour lines are overlaid on a Zebra Map, corresponding iso-values can be compared directly, and mismatches between contour lines and Zebra boundaries reveal local discrepancies between the two fields.

3 Ongoing Co-Design and Tool Development

The development of our multiplexing strategies has been guided by continuous collaboration with PDE researchers. Throughout the design process, domain experts provided feedback on analytical tasks, field selection, and comparison requirements. This iterative co-design process enabled us to explore different layer combinations and identify representations that effectively support scalar-field comparison in PDE workflows. Feedback from domain experts also revealed that there is no single optimal multiplexing design. Choices such as field assignments, layer ordering, and colormap configurations often depend on analytical goals and user preferences. While certain combinations provide effective defaults, experts frequently expressed the need to adjust these design parameters when investigating different aspects of model behavior.

Motivated by these observations, we are developing an interactive visualization tool for constructing and exploring multiplexed visualizations. The system provides recommended multiplexing strategies based on our design findings while allowing users to customize layer and color configurations. Future work will focus on evaluating its effectiveness in supporting scientific machine learning workflows.

References

[1] R. Borgo, J. Kehrer, D. H. S. Chung, E. Maguire, R. S. Laramee, et al. Glyph-based Visualization: Foundations, Design Guidelines, Techniques and Applications. In *Eurographics 2013 - State of the Art Reports*. The Eurographics Association, 2013. 2

[2] M. Chen, S. Walton, K. Berger, J. Thiyagalingam, B. Duffy, H. Fang, C. Holloway, and A. E. Trefethen. Visual multiplexing. *Computer Graphics Forum*, 33(3):241–250, 2014. 1, 2

[3] A. Farea and M. S. Celebi. Learnable activation functions in physics-informed neural networks for solving partial differential equations. *Computer Physics Communications*, 315:109753, 2025. 1

[4] W. Javed and N. Elmqvist. Exploring the design space of composite visualization. In *2012 IEEE Pacific Visualization Symposium*, pages 1–8, 2012. 2

[5] K. Koffka. *Principles of Gestalt Psychology*. Harcourt, Brace and Company, New York, NY, 1935. 1, 2

[6] Y. Xing, A. Farea, S. Khan, and M. Chen. A multiplexing design space: Theory, method, and application, 2026. 1