Creating a data visualization typically involves specifying a mapping between attributes of the data and visual encoding channels such as position, shape and color. But the original data is often not structured in a way that enables the direct creation of such visual mappings. Before visual mappings can be specified, the data first has to be transformed, either interactively or by writing scripts. This process, called data wrangling [1], is about manipulating data into a usable form (pivoting, folding, merging, filtering, joining), and is key to the visualization authoring process.

Some recent research projects about visualization authoring have focused on supporting expressive design workflows [2,3,4]. Others have investigated the potential of post-WIMP input modalities such as touch for manipulating visualizations [5,6,7]. But only few projects have investigated how to effectively support data analysts in the data wrangling step. Those that do either require programming skills [8] or rely on classic WIMP user interfaces designed for the desktop [9,10].

The goal of this internship is to design and prototype interaction techniques for post-WIMP data wrangling, that are better adapted to newer computing surfaces such as tabletops and wall displays. These post-WIMP interaction techniques should enable data analysts to interactively manipulate their data using diverse input modalities, giving special attention to extended vocabularies of multitouch gestures [11,12] possibly combined with pen input [13]. This project is the natural continuation of some of our recent investigations, including ink-based interaction [14,15] and expressive visualization authoring [16].

In this internship, the student will:
- do a literature review of interactive data wrangling techniques in the context of visualization authoring;
- design and implement post-WIMP interaction techniques for data wrangling (*);
- validate the proposed designs empirically with a user study (time permitting).

(*) Prototyping will primarily involve software development using Web technologies. To keep software prototyping effort manageable in the internship period, we will rely on existing high-level visualization libraries for the rendering of visualizations, most likely interactive grammars of graphics such as Vega-Lite [17] or ggplot2 [18]. We will also adapt incremental gesture recognizers already developed in the team [11] to recognize input. The student will have access to a Wacom Cintiq Pro 32” [19], which supports both pen and touch input. If the lab cannot be accessed because of covid-19 restrictions, we will fall back to a portable device.

Bibliography