We present an interactive, photo-editing prototype, 

**1 Abstract**

We present a photo-editing method that enables pixel parameter manipulation of image filtering by means of interactive painting. Predefined as well as custom image filters are exposed to the user, as a parametrizable composition of image operations. Brushes, as a sequences of actions, mapping user input (in terms of brush shape, flow, pressure, etc.) to arbitrary functions or convolution operations, are used to draw within the parameter space. Our prototype flowpaint demonstrates that interactive painting can be used to, e.g., locally tweak inadequate parametrization and, furthermore, provides a blueprint for an open photo-editing platform.

**2 Introduction**

Computational, artistic stylization of images can be defined as a composition of ordered, highly parametrizable image processing steps (Kyprianidis et al. 2013). The often tedious, trial-and-error prone process of manually tweaking an effect's outcome is typically limited to a subset of parameters of global scope. Local effect-parameters, e.g., intermediate computation results, are usually not accessible. Although, most consumer photo-editing suites apply a paint-brush metaphor (imitating real-life painting for, e.g., masking, coloring, and selecting), local parameter manipulation by painting is rudimentary supported (i.e., composing via masking and blending).

**3 Method**

Our method is targeting manipulation of local, per-pixel parametrization to support for (1) freestyle, artistic stylization and (2) modification and correction of (pre-)computed or intermediate results. In contrast to placing dynamic or static rendering primitives [Schwarz et al. 2007; Schmidt et al. 2011] our method further extends the concept of specialized, local computation parametrization [Todo et al. 2007] to a generalized, user-configurable brush-painting within effect parameter-spaces.

We present an interactive, photo-editing prototype, (flowpaint), that uses hardware accelerated image processing and implements customizable, per-pixel parameter painting. For it, effects (image-filters) with exposed parameter layers and brushes as sequences of well-defined layer-action pairs are used. In that way, flowpaint represents a first step towards photo-editing ecosystems where image filters are either user-created at run-time or distributed, shared, and customized through public image-filter repositories.

The GUI of flowpaint as shown above exposes, i.e., dynamic brush shape, brush, action, and effect configuration views.

**4 Results**

The expressiveness of the proposed brush model renders the process of specifying brushes and effect-dependent presets highly complex, requiring image-processing expertise. The following figures depict variations of automated and manually parametrized oil-painterly stylizations, with different brush-painting techniques applied to different areas of the image.