Challenges in User Experience Design of Image Filtering Apps

Mandy Klingbeil
Hasso Plattner Institute
Faculty of Digital Engineering,
University of Potsdam, Germany
Digital Masterpieces GmbH, Germany
mandy.klingbeil@hpi.de

Sebastian Pasewaldt
Digital Masterpieces GmbH, Germany
sebastian.pasewaldt@digitalmasterpieces.com

Amir Semmo
Hasso Plattner Institute
Faculty of Digital Engineering,
University of Potsdam, Germany
amir.semmo@hpi.de

Jürgen Döllner
Hasso Plattner Institute
Faculty of Digital Engineering,
University of Potsdam, Germany
juergen.doellner@hpi.de

ABSTRACT

Photo filtering apps successfully deliver image-based stylization techniques to a broad audience, in particular in the ubiquitous domain (e.g., smartphones, tablet computers). Interacting with these inherently complex techniques has so far mostly been approached in two different ways: (1) by exposing many (technical) parameters to the user, resulting in a professional application that typically requires expert domain knowledge, or (2) by hiding the complexity via presets that only allows the application of filters but prevents creative expression thereon. In this work, we outline challenges of and present approaches for providing interactive image filtering on mobile devices, thereby focusing on how to make them usable for people in their daily life. This is discussed by the example of BeCasso, a user-centric app for assisted image stylization that targets two user groups: mobile artists and users seeking casual creativity. Through user research, qualitative and quantitative user studies, we identify and outline usability issues that showed to prevent both user groups from reaching their objectives when using the app. On the one hand, user-group-targeting has been improved by an optimized user experience design. On the other hand, multiple level of controls have been implemented to ease the interaction and hide the underlying complex technical parameters. Evaluations underline that the presented approach can increase the usability of complex image stylization techniques for mobile apps.

CCS CONCEPTS

• **Computing methodologies** → **Image manipulation**; • **Human-centered computing** → *User centered design*; *Empirical studies in interaction design*;

KEYWORDS

mobile, image filtering, stylization, non-photorealistic rendering, interaction, user experience design, industry case

ACM Reference format:
Mandy Klingbeil, Sebastian Pasewaldt, Amir Semmo, and Jürgen Döllner. 2017. Challenges in User Experience Design of Image Filtering Apps. In *Proceedings of SA '17 Symposium on Mobile Graphics & Interactive Applications, Bangkok, Thailand, November 27-30, 2017,* 6 pages.
https://doi.org/10.1145/3132787.3132803

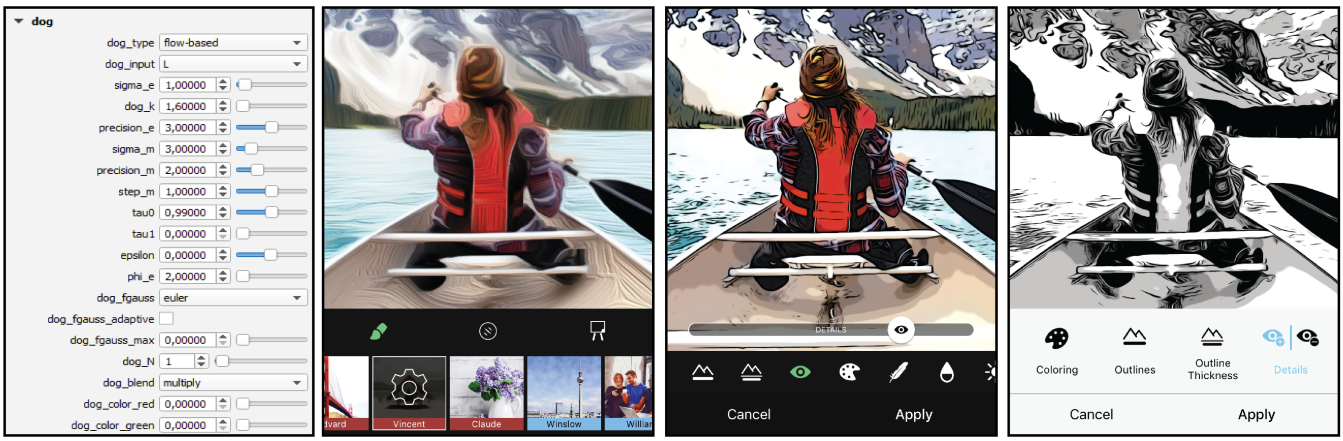
SA '17 Symposium on Mobile Graphics & Interactive Applications, November 27-30, 2017, Bangkok, Thailand

1 INTRODUCTION

Mobile photography enjoys a growing popularity: On the mobile photo-sharing community Flickr, smartphones have taken over professional cameras in their usage.[1] The application of on-camera visual effects, also known as filters, takes an essential part of the mobile photo sharing success, since filtered photos are 21% more likely to be viewed and 45% more likely to be commented on by consumers [Bakhshi et al. 2015]. Filters, such as Instagram's color adjustments or Prisma's style transfers, allow to enhance photos on-the-fly and give photos a stylized look without any prior knowledge of photo processing or editing. Yet, most image-based stylization techniques "are encapsulated algorithms that allow little influence on the final product other than global parameter changes"[2], and their interfaces "are often cluttered with an enormous number of parameters whose influence is difficult to understand"[2].

This paper presents the ongoing development of BeCasso [Pasewaldt et al. 2016; Semmo et al. 2016], a productization of image stylization research into an interactive mobile iOS app. BeCasso transforms photos into artistic renditions that are inspired by real paintings and drawings techniques, such as oil and watercolor paintings. The underlying GPU-based framework provides shader-based filtering that couples higher-level algorithmic support with low-level controls for parameterization [Semmo et al. 2016]. This way, BeCasso is able to provide an increased spectrum of interactivity through parameterizing image filters at three levels of control:

[1] https://www.flickr.com/cameras/ (last accessed 6/2017)
[2] http://tobias.isenberg.cc/Main/Research, last accessed June 27, 2017



Professional Research GUI | BeCasso - Presets | BeCasso - Global Parameters | Clip2Comic - Parameter Painting

Figure 1: Comparison of four different GUI approaches for varying levels of control: The research GUI exposes all global technical parameters, allowing for a low level of control. In contrast, BeCasso combines either all technical parameters into a preset, or a subset into a global parameter. To enable pixel-precise low-level control, Clip2Comic implements parameter painting by giving a user the possibility to locally decrease or increase several technical parameters at once.

(1) parameter presets followed by (2) global parameter adjustments, redefined by (3) complementary on-screen painting that operates within the filters' parameter spaces for local adjustments.

By providing high-level and low-level control [Isenberg 2016], BeCasso is designed to meet the expectations of mobile artists and casual users, and therefore addresses people along two dimensions as presented by Hassenzahl [2005]: Users are able to reach their "do-goals" (e.g., apply a filter on a picture) and are simultaneously empowered to stimulate their creative vein, expressing themselves through digital art, maybe even feel competent in the field of art. This way, they also reach their "be-goals". One year after publishing the app on the Apple App Store, user studies have shown that there still remain three important challenges:

(1) *Parameter Accessibility:* When implementing a filter, only parameters with a significant relevance for further editing, i.e., with a visible effect, should be exposed to users.
(2) *Usability:* The user interface design should empower users to understand and easily use all levels of control. The app is only usable if it is easy to learn, efficient to use, memorable, error tolerant and satisfying to use [Nielsen 1994].
(3) *Targeting:* As each user group has its own skill levels, expectations and work-flows to achieve the goal of creating visual expressions, a focused user experience design is required. This includes all aspects defining the interaction between users and the product.

All three challenges need to be covered to provide an experience that facilitates casual creativity [Winnemöller 2013]. This paper presents industry cases where different approaches have been implemented, tested and verified over the last year by conducting user studies with external participants. From the insights gained through those industry cases and evaluations, we outline usability issues and challenges in the user experience design of mobile image filtering apps.

2 RELATED WORK

PencilFX, ToonPAINT [Winnemöller 2013] and PaintCan [Benedetti et al. 2014] are examples of mobile apps that originate from research prototypes. They typically reduce a large number of technical parameters to a few comprehensive parameters that are exposed to users. State-of-the-art filtering apps, such as Waterlogue[3] and Brushstroke[4], typically further reduce the parameter space, but provide only high-level parameters for effect tuning (e.g., presets, global adjustments). The user experience for most of those apps was designed for people with limited artistic skills, or at least, who are not art professionals, but who would like to take advantage of the development of digital art and painting tools. As example, PaintCan is designed for novices, overcoming technical, artistic and creative challenges [Benedetti et al. 2014]. ToonPAINT is designed to address non-professionals, but provides more than a one-click solution, letting users not only transform a photo into a sketch by selecting a preset, but also color it with touch gestures to engage users and allow a sense of personal achievement [Winnemöller 2013]. To develop user-centric products, it is required to get a deep understanding of the context as well as of the people it is designed for. For this, we have collected details about our target groups in several ways, e.g., qualitative user studies, in-app metrics, and key performance indicators, and finally focused the app on mobile artists and casual users. In BeCasso, we combine high-level presets with low-level painting, e.g., as used in ToonPAINT and PaintCan, and extend the concept of specialized local parameterizations [Anjyo et al. 2006; Todo et al. 2007] to a generalized brush-based painting within effect-parameter spaces, e.g., enabling user-defined stroke orientations. We show that this approach can greatly increase the feasibility of implementing interactive tools that adjust the appearance of filters at run-time, which poses a contemporary

[3] http://www.tinrocket.com/waterlogue
[4] http://www.codeorgana.com/brushstroke

field of research of the non-photorealistic rendering (NPR) community [Isenberg 2016]. Previous works show that this type of "user-centric NPR" [Winnemöller 2013] has potential to assist art creation on desktop systems [Gerl and Isenberg 2013; Salisbury et al. 1997] or on mobile devices [Benedetti et al. 2014].

3 USER-DRIVEN DEVELOPMENT

BeCasso is an example of a knowledge and technology transfer from research to production, and thus to end-users. It's development benefited from experiences made with Clip2Comic, a mobile iOS app that allows to transform photos and videos into cartoons, and that was published in 2014. Both projects started as research prototypes, focusing on maximal low-level control over technical parameters (Figure 1).

In this section we outline steps necessary to develop such research prototypes to consumer products that target specific market groups by optimizing the user interface design and parameter accessibility to hide potentially complex technical parameters and to increase usability.

3.1 Target Groups

While improving BeCasso, we identified two major groups interested in this technology—both non-professionals with limited artistic skills—that are outlined in the following:

3.1.1 Mobile Artists. Mobile artists, also known as phone artists, are a group of serious hobbyists in the field of mobile art. They use a mobile phone or a similar device to take photographs and use apps to edit or manipulate them. The founders of one of the biggest mobile photography communities, Matt Cooper and Roger Gilbert, describe this group as addictives who need to get creative with their mobile devices, regardless of their skill level or their age, producing, sharing and discussing creative imagery. The artistic process as well as sharing the artwork were important for this group.[5]

Mobile artists are interested in high-quality styles, open to individuality, and willingly to invest time into the creative process. For mobile artists, presets are only a starting point to explore the variety of a style, while global and local parameter adjustments gives them the sense of personal achievement.

3.1.2 Casual Users. While mobile artists are not interested in one-click solutions, casual users think more about their social engagement when editing a photo. Thus, casual users apply filters that make their photos look more fun and unique, being always open to new types of filters. As user research showed, the selection of a preset itself is already an important part of the creative process for them. Their goal is to select the right filter and instantly share the result. Casual users want a bigger selection of presets, i.e., intensive high-level interaction instead of laborious and time-intensive low-level controls.

3.2 Parameter Accessibility

Complex image stylization techniques, such as comic effects [Winnemöller et al. 2006], often consist of multiple rendering steps, e.g., computation of flow-fields, bilateral filtering for image smoothing and an extended difference-of-gaussian (xDoG) filter for edge rendering [Winnemöller et al. 2012], that are controlled by numerous technical parameters. Choosing and modifying the right parameters often requires in-depth knowledge of the underlying rendering techniques.

For example, a user can modify edges in the final rendering of BeCasso by controlling the number of contours, their thickness as well as the level-of-detail, i.e., if more details through more, finer edges or less details through less, thicker edges should be rendered. The level-of-detail parameter is controlled by modifying the bilateral filter ($sigma_{dg}$ & $sigma_{rg}$) to reduce high-frequency details in the input image and by modifying the xDoG ($sigma_e$) to be more sensitive to minor gradient changes.

Through examining the dependencies between parameters, it is possible to combine them without loosing too much control over the output, while increasing accessibility and usability for non-expert users. For example, the above described modification of details is represented by only one *detail* parameter slider (Figure 1) to users, but is mapped internally to multiple technical parameters. While the value of the slider seems transparent for users, the available value range through the slider is reduced to achieve only reasonable effects.

Defining and exposing suitable parameter (combinations) for end-users is a highly iterative procedure that requires in-depth-knowledge of the rendering techniques as well as constant user feedback. If too many parameters are exposed, users may be overwhelmed and thus, the creative process may be impeded. In contrast, exposing too few parameters reduces the functionality of the effect, and thus the freedom of design.

3.3 User Interface and Interaction Design

As we started to launch multiple image stylization apps, we developed an iOS framework that allows us to reuse basic user interface elements, e.g., layout components and single views. This allows us to integrate new features or quickly implement new apps, e.g., with different filters and their own theming, having only one app structure in general. By having two image stylization apps (BeCasso and Clip2Comic) in the App Store, we are able to compare user behavior among different app types, e.g., premium vs. freemium apps, but also to do A/B testing with external testers using Testflight[6].

3.3.1 Presets. Instagram[7] introduced post-processing filters for phone cameras in 2010 to a wide range of people. Since then, presets are a common concept to make image stylization available to the general public. BeCasso offers between five and nine presets per filter. Each preset was defined through heuristic evaluation including a set of 20 representative pictures with different motives and characteristics (e.g., from the stylization benchmark image set [Mould and Rosin 2016, 2017]). All presets represent an individual parameter combination to achieve different kind of results, e.g., an oil style that adopts Vincent van Gogh's strong color application and abstract, long brush strokes with only a few details in "Starry Night", and another oil style that allows portraits to be painted as detailed and soft as Leonardo Da Vinci's "Mona Lisa".

[5] https://iphonephotographyschool.com/mobitog/, last accessed June 23, 2017

[6] https://developer.apple.com/testflight/

[7] https://www.instagram.com

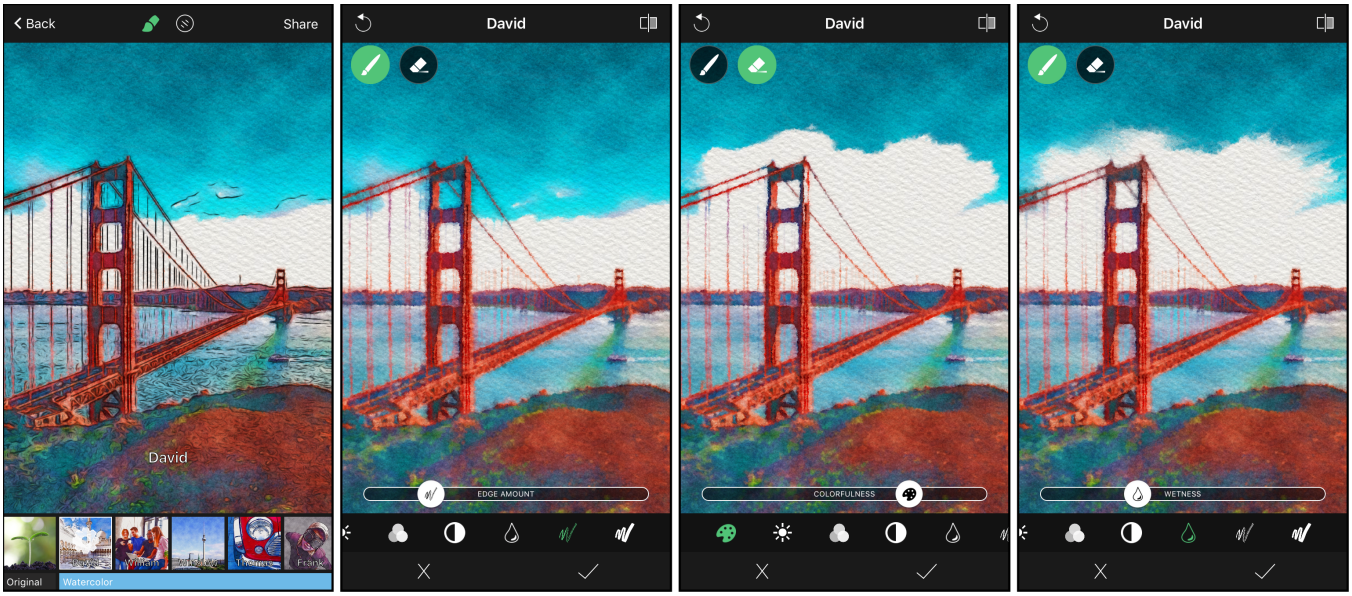


Figure 2: Example of a workflow in BeCasso version 1.0. A user selects a watercolor preset as starting point (1). By tapping the preset again, he opens the edit screen for all available parameters of this style. In the next step, the user decreases the parameter "Edge Amount" globally—for the whole picture—through the slider (2). Afterwards, he selects "Colorfulness" and activates the eraser to locally decrease its value for parts of the sky with his finger (3). In a last step, he selects "Wetness" and activates the brush to soften the transition between white and blue in the sky (4).

Mobile artists are attracted by a style and see presets as a starting point to create new artwork from there, and are open to invest time and effort to achieve the best possible results. Casual users prefer one-click solutions instead, seeking fast results. They are in search of presets that produces a "wow" effect, something that would stick out of the masses. Thus, the selection of a preset itself is already a creative process in which casual users invest time, carefully choosing the right preset to represent their emotions and mood, giving them a personal achievement.

Evaluation showed that if users liked a style, they would request even more presets with different parameter combinations. Therefore, presets already produced great results, without making it necessary to globally adjust parameter values through sliders (see *Global Parameter Adjustments*). Furthermore, App Store reviews give a sense of how well presets work with user-generated content: "I am a photographer and have always liked the impressionist look for images. It takes a lot of skill to make that happen in Photoshop. But this app has made it a child's play to make a painting out of your photo. There are so many styles you can make with mere clicks. Love this app to the core."

If users didn't like a style, they tapped through its presets displeased because of the strong resemblance: Unsurprising, five watercolor presets would show significant similarities—as their feature all the watercolor style—and vary only in their values of available style parameters. Especially casual users are easily frustrated by look-alike filters, and are mainly motivated to keep tapping through all presets by the expectation to find further, completely other styles within the app.

3.3.2 Global Parameter Adjustments. The majority of users is familiar with the usage of sliders to edit an effect globally. Mostly, apps offer the possibility to adjust the strength of the applied filter, i.e., how strong the original photo and the stylized photo are blended. BeCasso offers direct access to the filter itself to edit parameters that have a visible impact on the output. Several iterations were needed to distinguish parameters that are helpful for users, being easily understandable in their influences over the output, from parameters that would cause more confusion and frustration than helping to achieve better results. As an example, six parameters for the oil style were reduced to three, and five parameters for the watercolor style needed to be reduced to four to improve the overall usability. Dependent parameters have been aggregated, and are now accessible through a single slider, while others are set to a constant value and remain hidden. This way, the expressiveness of styles is maintained.

The parameter naming is one of the influencing factors for user satisfaction when exploring filters. In BeCasso, different approaches were tested, e.g., using terms of the original artistic process itself, or terms that would nearly describe what happens on the screen. "Relief" as description for the amount of oil paint that was (digitally) applied to the picture, was accepted immediately by more than 90% of our test users, while technical terms such as "Bumpiness", "Bumpscale" or "Texture" didn't support their understanding of the parameter. Additionally, artistic terms made test users feel "more like a real artist", thus supporting their creativity.

3.3.3 Local Parameter Adjustments. The first version of BeCasso offered drawing tools to locally adjust style parameters (Figure 2).

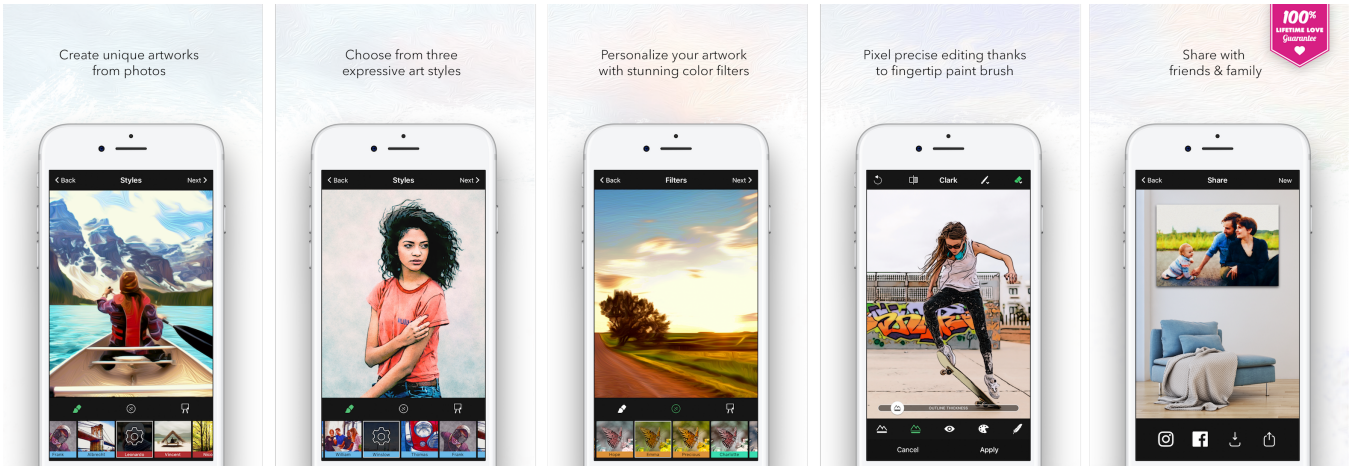


Figure 3: Examples of BeCasso marketing materials for the Apple App Store. The images are representative screenshots depicting the different styles (watercolor, oil paint and cartoon) for various motives (landscapes, peoples, activities). The motives as well as the tonal voice has been optimized towards the target groups.

For this purpose, global and local parameter editing were presented in the same view. After selecting a parameter, e.g., "Relief" when editing an oil style, users were able to set its global value through a slider. To empower users to adjust the value of a parameter for selected areas in their picture, e.g., to reduce the shown relief on a face, but to keep it in the background of the picture, BeCasso provided specific touch gestures. Users could either use a tool named "Brush" to locally increase, or an "Eraser" to locally decrease the value of the parameter. By swiping over the picture, they technically altered a mask that was internally blended with the corresponding global parameters.

3.3.4 Challenges. We discovered multiple usability issues for the initial approach, based on the research prototype. We will outline the most important issues in the following.

(1) Combining two features in a single view needs a lot of visual support, as it is not necessarily expected by users. After exploring the sliders to adjust style parameters globally, more than 90% of our 26 initial test users didn't even try to draw on the picture. As the "Brush" and "Eraser" were shown on the upper part of the screen and not right next to the parameter selection or sliders, users missed this feature very easily.
(2) Discovering and understanding a feature is limited by the visibility of an action. It is easy to learn how to paint a red line onto a white canvas when selecting a red color and touching the screen. With parameters like "Relief" or "Wetness", users have been easily confused of what should have happened. Furthermore, it is clear that after coloring a white canvas into red, it cannot turn any more red. When applying "Wetness" in BeCasso, users were not able to know when they reached the maximum of the parameter value.
(3) BeCasso's interaction concept differed too much from established concepts. In the app, users set the global value of a parameter through a slider, and have been able to adjust this value locally through swiping over the picture. As an example, users were able to set the amount of outlines (contours) to 0 via slider, showing no lines at all. But afterwards, they could have added a few lines with their fingers in desired areas using the "Brush". That means both the slider as well as the mask affected the selected parameter in a specific area. In contrast, well-known apps like FaceTune[8] uses sliders to directly affect one area selected by touch gestures. Other apps offer sliders to edit parameter values only on a global level, but let users also draw with their fingers: Touch gestures would erase or apply the whole effect to parts of the picture, but wouldn't have any connection to the selected parameter within the view.
(4) How to locally influence a parameter value properly through touch gestures required too much technical background knowledge. Even after explaining the conceptual model to test users, only a few have been able to actively use the feature without additional support.

Multiple iterations have led us to an approach that was inspired by beautification apps such as FaceTune: The local parameter adjustments are offered as "Retouch Tools" in an additional distinct view (Figure 1 - Clip2Comic Parameter Painting), thus, clearly separating it from the global parameter adjustments. Further, the brush and eraser icons have been replaced by two icons for each paintable parameter. The amount of paintable parameters was reduced to those with the most impact, making almost every user action visible. The "+" or "-" inside the icon describes the intention to locally increase or decrease the respective parameter. First tests indicate that almost all of our test users, independent of their affiliation with one of the target groups, were able to use this feature for their own creative artwork without any explanation or support.

4 USER-DRIVEN MARKETING

As photo filters are popular among all ages, it seems reasonable to publish an app with a representative name and branding, showing

[8] https://www.facetuneapp.com

screenshots of the app, and describing its features without concentrating on a certain target group. For BeCasso, we wanted to attract a wide range of users, including digital artists—experts or novices—by presenting a professional tool that produces high-quality painterly effects.

All App Store materials for BeCasso, including subtitles, the app description, screenshots (Figure 3), a 30 seconds preview video as well as keywords for the App Store search, have been created by professionals. We altered elements of the App Store materials on a regular basis, including the screenshots and the keywords, and used App Store analytics to identify weak points within the branding and marketing. We were also able to collect data of users who discover apps by chance, e.g., on the App Store home page, as BeCasso was features several times in various countries.

For more accurate conclusions about the impression of the app, we utilized the results of App Store SearchAds[9]. SearchAds allow to define the target groups even more precisely. Additionally, we conducted user tests and user survey to identify the expectations of the revealed groups.

5 CONCLUSION

We showed that parameter accessibility, usability and targeting are three remaining challenges for the development of interactive image stylization apps on mobile devices that allow multiple levels of control. After several iterations focusing on the user interface and marketing aspects, we came to the conclusion that there is not one optimal "one size fits all" solution. Casual users are more interested in new and varying filters, especially those which make their photo look fun and unique, without the need to take care of details. However, mobile artists want to integrate the state-of-the-art rendering techniques into their overall creation process, using the high-resolution exports from BeCasso in conjunction with other apps, whether as starting point or, as said by a test user, "for a painterly look at the end of an edit".

Despite of those different requirements, we have developed an approach to open up the low-level control of filter parameters that works for both groups. We categorized local parameter painting as "Retouch Tools", presenting it separately from all other controls that concern the style. As first user studies show, casual users seem to be more open to use this feature than before. More qualitative and reliable quantitative data will be available in the near future, as Clip2Comic was released with the proposed user interaction design in the App Store in June 2017. Users of Clip2Comic are now able to test those "Retouch Tools" as often as they like, but need to purchase them to apply their changes. More than one third of users have purchased this feature so far, which underlines the proposed interaction concept.

If we want end-users to profit from image stylization research through the productization in form of mobile apps, we need to design the user experience of such apps more specific, getting to know the target groups better and identifying their requirements. This way, we are able to explicitly give users the opportunity to gain control over the developed techniques.

[9]https://searchads.apple.com

ACKNOWLEDGMENTS

We thank Frank Schlegel and Paul Schröder for their contributions to the design and implementation of the proposed system. This work was partly funded by the Federal Ministry of Education and Research (BMBF), Germany, for the AVA project 01IS15041.

REFERENCES

Ken-ichi Anjyo, Shuhei Wemler, and William Baxter. 2006. Tweakable Light and Shade for Cartoon Animation. In *Proc. NPAR*. ACM, New York, 133–139. https://doi.org/10.1145/1124728.1124750

Saeideh Bakhshi, David A. Shamma, Lyndon Kennedy, and Eric Gilbert. 2015. Why We Filter Our Photos and How It Impacts Engagement. In *Proc. ICWSM*. AAAI Press, 12–21. http://www.aaai.org/ocs/index.php/ICWSM/ICWSM15/paper/view/10573/10484

Luca Benedetti, Holger Winnemöller, Massimiliano Corsini, and Roberto Scopigno. 2014. Painting with Bob: Assisted Creativity for Novices. In *Proc. ACM UIST*. ACM, New York, 419–428. https://doi.org/10.1145/2642918.2647415

Moritz Gerl and Tobias Isenberg. 2013. Interactive Example-based Hatching. *Computers & Graphics* 37, 1–2 (2013), 65–80. https://doi.org/10.1016/j.cag.2012.11.003

Marc Hassenzahl. 2005. *The Thing and I: Understanding the Relationship Between User and Product"*. Springer Netherlands, Dordrecht, 31–42. https://doi.org/10.1007/1-4020-2967-5_4

Tobias Isenberg. 2016. Interactive NPAR: What Type of Tools Should We Create?. In *Proc. NPAR*. Eurographics Association, Goslar, Germany, 89–96. https://doi.org/10.2312/exp.20161067

David Mould and Paul L. Rosin. 2016. A Benchmark Image Set for Evaluating Stylization. In *Proc. NPAR*. Eurographics Association, Goslar, Germany, 11–20. https://doi.org/10.2312/exp.20161059

David Mould and Paul L. Rosin. 2017. Developing and Applying a Benchmark for Evaluating Image Stylization. *Computers & Graphics* 67, C (Oct. 2017), 58–76. https://doi.org/10.1016/j.cag.2017.05.025

Jakob Nielsen. 1994. *Usability Engineering*. Morgan Kaufmann.

Sebastian Pasewaldt, Amir Semmo, Jürgen Döllner, and Frank Schlegel. 2016. BeCasso: Artistic Image Processing and Editing on Mobile Devices. In *Proc. SIGGRAPH ASIA Mobile Graphics and Interactive Applications*. ACM, New York, 14:1–14:1. https://doi.org/10.1145/2999508.2999518

Michael P. Salisbury, Michael T. Wong, John F. Hughes, and David H. Salesin. 1997. Orientable Textures for Image-based Pen-and-ink Illustration. In *Proc. ACM SIGGRAPH*. ACM, New York, 401–406. https://doi.org/10.1145/258734.258890

Amir Semmo, Jürgen Döllner, and Frank Schlegel. 2016. BeCasso: Image Stylization by Interactive Oil Paint Filtering on Mobile Devices. In *ACM SIGGRAPH 2016 Appy Hour*. ACM, New York, 6:1–6:1. https://doi.org/10.1145/2936744.2936750

Hideki Todo, Ken-ichi Anjyo, William Baxter, and Takeo Igarashi. 2007. Locally Controllable Stylized Shading. *ACM Trans. Graph.* 26, 3 (2007), 17:1–17:7. https://doi.org/10.1145/1276377.1276399

Holger Winnemöller. 2013. NPR in the Wild. In *Image and Video based Artistic Stylisation*, Paul Rosin and John Collomosse (Eds.). Computational Imaging and Vision, Vol. 42. Springer, London/Heidelberg, Chapter 17, 353–374. https://doi.org/10.1007/978-1-4471-4519-6_17

Holger Winnemöller, Jan Eric Kyprianidis, and Sven Olsen. 2012. XDoG: An eXtended Difference-of-Gaussians Compendium including Advanced Image Stylization. *Computers & Graphics* 36, 6 (Oct. 2012), 740–753. https://doi.org/10.1016/j.cag.2012.03.004

Holger Winnemöller, Sven C. Olsen, and Bruce Gooch. 2006. Real-Time Video Abstraction. *ACM Transactions on Graphics* 25, 3 (July 2006), 1221–1226. https://doi.org/10.1145/1141911.1142018