

AR MUSEUM: A MOBILE AUGMENTED REALITY APPLICATION FOR INTERACTIVE PAINTING RECOLORING

Mattia Ryffel¹, Fabio Zünd², Yağız Aksoy³, Alessia Marra¹, Maurizio Nitti¹, Tunç Ozan Aydın¹ and Bob Sumner¹

¹Disney Research Zürich, Switzerland

²Game Technology Center, ETH Zürich, Switzerland

³ETH Zürich, Switzerland

ABSTRACT

We present a mobile augmented reality application that allows its users to modify colors of paintings via simple touch interactions. Our method is intended for museums and art exhibitions and aims to provide an entertaining way for interacting with paintings in a non-intrusive manner. Plausible color edits are achieved by utilizing a set of layers with corresponding alpha channels, which needs to be generated for each individual painting in a pre-processing step. Manually performing such a layer decomposition is a tedious process and makes the entire system infeasible for most practical use cases. In this work, we propose the use of a fully automatic soft color segmentation algorithm for content generation for such an augmented reality application. This way, we significantly reduce the amount of manual labor needed for deploying our system and thus make our system feasible for real-world use.

KEYWORDS

Image Processing, Color Editing, Soft Color Segmentation, Augmented Reality

1. INTRODUCTION

Augmented reality represents a great opportunity to make everyday scenes more engaging to the user. In this work, we create an augmented reality application to make museums interactive and hence more engaging for visitors by letting them create personalized versions of the paintings via recoloring. We call our application AR Museum with the tagline Create your own masterpiece. Using our application that runs on mobile devices, users can change the colors of any painting via simple pointing at the color they wish to change on the mobile device. This way, we aim to attract the attention of especially young visitors to the art pieces in museums.



Figure 1. Given a digital image of a painting (a), we automatically determine a set of representative color distributions and corresponding so color segments (b). We present an augmented reality application aimed at museums that allows its users to change colors of the original painting in real-time (c). The content creation for such a system is streamlined through the use of soft color segments, which makes our application realizable for museums with large collections of paintings. The original painting is Portrait of Patience Escalier by Vincent van Gogh

Having a large number of paintings in a museum interactive instead of a small number of selected pieces is essential for a more effective application. However, manually creating the layers that are required for a seamless recolorization of complex paintings is a tedious task for an artist. We solve this scalability issue through the use of soft color segmentation. Figure 1 shows the automatically generated soft color segments and our recoloring application in action.

Decomposing an image into a set of constituent layers is one of the fundamental steps required in various image-editing applications. In fact, commonly used image editing tools (such as Adobe Photoshop, Gimp, among many others) take a layer-based approach to image editing. When creating compositions from scratch, it is common practice to create a layer structure where each layer contains a set of assets that will most likely receive similar edits. Equally important is to ensure that edits in one layer do not affect other layers, as well as not having an adverse effect on the overall image quality. In that sense, the layer structure of an image defines the demarcations between different image edits.

Unfortunately, in most cases, an artist does not have access to a corresponding layer structure while working with existing images. The application we present in this paper, namely interactively modifying colors of real-world paintings in an augmented reality environment, falls under this category. In our application, an existing image needs to be modified in a visually plausible manner in the absence of any accompanying layer structure. Creating such a layer structure is possible to a certain extent by manually drawing masks and utilizing range selection tools, but is often a very tedious and time-consuming task. Thus, automatically performing the layer decomposition is highly desirable, as this process is not a means for expressing artistic intention but merely a necessary step to make the final augmented reality system work.

In this production paper, we propose an end-to-end pipeline for locally changing colors of famous paintings interactively in real-time in an augmented reality environment. To refrain from color changes affecting the whole image, the paintings are decomposed into a set of layers that constitute similar colors in a pre-processing step. In order to ensure that changing the color of a layer does not produce visual artifacts such as abrupt transitions, the spatial extent of each layer is defined as an alpha channel instead of a binary edge. In our system, we perform this pre-computation step fully automatically using a recent soft color segmentation technique (Aksoy et al., 2017), significantly reducing the manual labor required in the process. As such, our paper demonstrates a novel use case of soft color segmentation in an augmented reality application.

2. RELATED WORK

Augmented Reality (AR) holds unique and promising potential to bridge between real-world activities and digital experiences, allowing users to engage their imagination and boost their creativity (Zhou et al., 2008). Combining different input modalities, such as touch and gestures, combined with the inherent intuitive camera control in AR applications empowers content creators to produce new forms of interactive experiences. Hence, the use of AR in museums has been attracting the research community for a long time. Through AR, the handheld museum guides can be enhanced in capability to provide a better flow of information to the user about the museum (Miyashita et al., 2008). Other work (Wojciechowski et al., 2004) created a system in early days of mobile AR that displays 3D versions of some of the pieces in museums. The positive effect of AR for an enhanced learning environment in museums has been shown in multiple studies (Wu et al., 2013) (Yoon et al., 2012). Other than an enhanced learning environment, Chang et al. (Chang et al. 2014) show that casual visitors can be guided to appreciate the art better through AR. On the other hand, reflections (Aydin et al., 2009) can have an adverse effect on presenting artwork in an AR environment. AR can also be used for personalized visits to the museum via creating stories (Keil et al., 2013). Recent work (Billinghurst et al. 2015) presents a comprehensive survey on applications on AR including its use in museums. In this work, we attempt to make museums more engageable especially for young visitors by allowing the users to personalize the color of paintings through the use of a mobile device. Recent work (Magenat et al., 2015) (Zund et al., 2015) has shown how AR can be a host for enhancing creative real-world activities in a playful way. In this context, we employ AR to enhance the experience of visiting an art exhibition with an interaction component.

Soft color segmentation is defined as decomposition of an image into layers of homogeneous colors with opacity channels. Early work on the subject, notably the alternating optimization algorithm (Tai et al., 2007), show several applications of soft color segments for image editing. The alternating optimization algorithm

takes the number of layers as input and iteratively solves for the color content of the layers, referred as the color model, the opacity channels and the layer colors. Their approach especially lacks in terms of the smoothness of the opacity channels and homogeneity of layer colors. Recent work (Tan et al., 2016) (RGBSG) selects the color model before opacity estimation and fixes the layer colors to be solid. RGBSG requires the model colors to encompass the colors in the image from outside in the color space, which results in unrealistic colors in the model. This approach also suffers from prohibitively long run times as they optimize for the opacities of every pixel simultaneously, and requires the layer ordering beforehand, which is ill-defined and a nonintuitive task for a user. The unmixing-based approach (Aksoy et al., 2017) decreases the run time to around 1 minute per image by separating the opacity estimation and spatial smoothness problems. Their approach, which we use in our application, gives smooth layers with little variation in layer colors fully automatically, without requiring the number of layers or layer ordering. Algorithmic details for this method is discussed in detail in Section 3.

3. SOFT COLOR SEGMENTATION AND CONTENT GENERATION

In the proposed system the generation of layers is done automatically using the soft color segmentation algorithm (SCS) (Aksoy et al., 2017). SCS has two main steps: color model estimation and color unmixing. The color model is composed of a set of 3D RGB normal distributions representing the dominant colors in the image. It is constructed such that the colors of all the pixels are well represented, hence the number of layers depend on the color content of the image. The model is then given to the color unmixing algorithm for the estimation of layer colors and opacities for each pixel.

The color model estimation is done based on a voting scheme, where the most under-represented color is determined at each iteration. For a selected color, a seed pixel is selected from regions without strong edges in the image. From the local neighborhood of the selected seed pixel, a normal distribution is estimated. Figure 2 shows the seed pixels and the estimated color model for an example image.

We use the alpha add representation for overlaying opacity channels (Aksoy et al., 2017). As we assume purely opaque input paintings, our representation of opacities give two constraints for the opacities and layer colors to give us the original painting:

$$\sum_i \alpha_i^p \mathbf{u}_i^p = \mathbf{c}^p \quad \text{and} \quad \sum_i \alpha_i^p = 1 \quad \forall p, \quad (1)$$

where α_i^p and \mathbf{u}_i^p represent the opacity and color of layer i at pixel p .

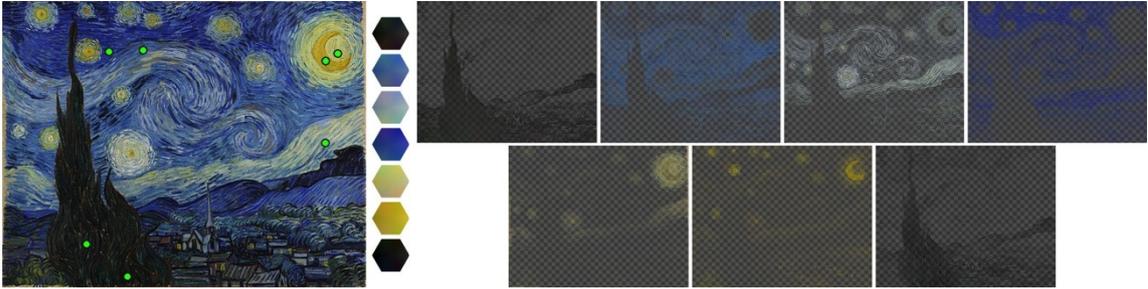


Figure 2. The unmixing-based soft color segmentation algorithm (Aksoy et al., 2017) first selects seed pixels (le) to determine a set of normal distributions in the RGB space to represent the colors in the input image. The hexagons show the color content of each normal distribution in the automatically estimated color model. This color model is then fed to the color unmixing algorithm to get spatially smooth layers of homogeneous colors, which is used for recoloring in our application. The original painting is The Starry Night by Vincent van Gogh

The opacities and colors for each layer are determined in two steps. The first step is the minimization of the sparse color unmixing energy (F_s). Given the color model, sparse color unmixing energy is minimized with respect to layer colors and opacities for each pixel independently:

$$\mathcal{F}_S = \sum_i \alpha_i^p \mathcal{D}_i(\mathbf{u}_i^p) + \sigma \left(\frac{\sum_i \alpha_i^p}{\sum_i (\alpha_i^p)^2} - 1 \right), \quad (2)$$

where the layer color cost $\mathcal{D}_i(\mathbf{u}_i^p)$ is defined as the squared Mahalanobis distance of the layer color \mathbf{u}_i^p to the layer distribution $N(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$, and σ is the fixed sparsity weight. The energy minimization with constraints in Equation 1 is done via the algorithm used by (Aksoy et al., 2016).

As this energy minimization is defined independently for each pixel, the opacity channels are not necessarily spatially smooth. The second step addresses this issue by first smoothing the opacity channels using the guided filter (He et al., 2013) and then reestimating the layer colors to satisfy the first constraint in Equation 1.

The resulting layers, such as the ones presented in Figure 2, are characteristically of homogeneous colors and have smooth opacity transitions. Such layers make high-quality color editing straightforward: simply changing one of the colors in the color model results in natural color transitions in the image.

Before porting the layers to the mobile application, we define interaction regions as shown in Figure 3. These regions link the spatial extent of the user interaction with corresponding layers. Notice that we define these regions for layers with distinct hues and typically leave the black and white layers constant in order to preserve the luminance of the image constant.

4. INTERACTIVE COLOR EDITING APPLICATION

We developed a mobile app that allows users to play and interact with the colors of paintings by utilizing the soft color layers we discussed in the previous section and visualize their work through AR. To use the app, the user stands in front of a painting, holding a mobile device facing the painting. On the screen, a live video stream from the back-facing camera is presented to the user. As they point the mobile device at the painting, a virtual version of the same painting is shown, seamlessly integrated into the real world. To the user, the physical painting is replaced with a virtual, digitally enhanced version. The user can now click on different regions of the virtual painting to throw paint drops, which recolors regions of the virtual painting. The app allows the user to be creative and produce a personal, recolored version of the original painting. Figure 4 illustrates two individual examples. Figure 6 shows the application in action in a setup similar to a museum on a hand-held device for multiple paintings.

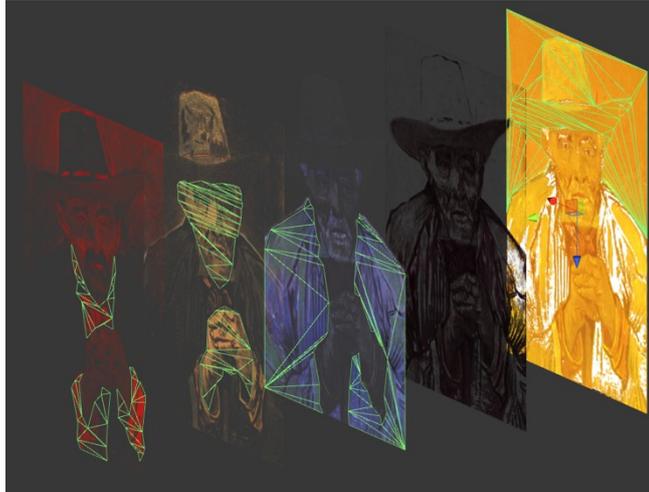


Figure 3. The regions for the user to interact with to change the color of a layer is roughly defined by the artist before the layers are ported to our mobile application



Figure 4. Our augmented reality application for interacting with paintings. The physical painting is seamlessly replaced on the mobile device with a digitally enhanced version, which allows the user to recolor regions and create their own masterpiece. The user recolored parts of *Café Terrace at Night* (left), and of *Self-Portrait* (right), both by Vincent Van Gogh

4.1 Implementation

In a preprocessing step, the virtual painting is divided into color layers using the soft color segmentation technique (Aksoy et al., 2017) to computationally provide a set of layers. For each layer, we manually define the collision regions as shown in Figure 3 that will allow the user to interact with the painting.

During runtime, the app analyzes the live camera stream and detects and tracks the templates of the paintings, which are registered in the app as image markers appearing in the camera image. As the virtual camera pose is estimated, the corresponding virtual painting is rendered on top of the camera image aligned with the image of the physical painting. When the user touches a region of the painting on the screen, a ray is cast to determine which layer is being activated. The app generates a random color and creates a visual effect of painting drops shooting from the virtual camera towards the painting. Upon impact, the drops recolor the selected corresponding layer by progressively shifting the hue to match the new color.

The app was realized using the Unity 5.5 game engine. The different layers of each painting were imported in the engine as ARGB textures. Each texture is then placed in the scene on different quads overlaid over the virtual painting marker. For each plane, we assign a set of colliders that match the corresponding colored regions of the painting. When the user taps on the tablet a ray is cast to identify the area of the painting that is activated. In addition, a custom shader is employed to create a smooth color shifting effect. Starting from the hue of the initial average color of the layer, whenever a new color is applied, it is converted to HSV space and the hue of each pixel is progressively shifted towards its difference from the initial value. In this way, we can preserve the individual features of each pixel without colorizing the whole layer. For the image marker tracking, we used Vuforia 6.2.6 as it provides a fast and reliable way to track the paintings and estimate the camera position.

Our application runs smoothly at 30 fps on second generation iPads, iPhones 5s, Nexus 7 (second generation), and Xperia Z tablets. A live video capture of our system is presented as supplemental material.

4.2 Automatic vs. Manual Layer extraction

While it is certainly possible to produce the color layers manually instead of using the aforementioned soft color segmentation algorithm, the automatic method has significant advantages. In Figure 5, we show the manually generated soft color layers for the painting we present in our teaser image. The manually generated layers were produced by utilizing the color range selection tool in Adobe Photoshop CC. In order to obtain plausible layers, our artist had to adjust the fuzziness control of the tool, as well as performing additional manual edits. In general, a significant limitation of this tool is that it does not unmix the colors as the automatic soft color segmentation algorithm does. Performing the unmixing manually would have resulted in significant additional manual labor, which we did not attempt in this example. Even though we avoided manual unmixing, producing the layers in Figure 5 took roughly 10 minutes of artist time, versus

approximately 1 minute fully-automatic processing time required by the automatic method. As a result, the use of the soft color segmentation method for our application makes the system scalable to a large number of paintings, making it possible to realize the AR application for a large collection of paintings with little cost.

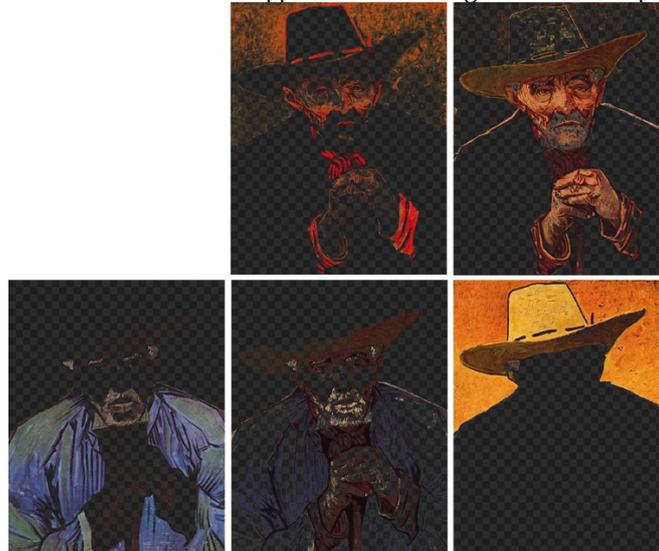


Figure 5. Manually generated layers by an artist for the painting in the teaser

5. CONCLUSION

We presented an AR system for interacting with paintings by manipulating their colors. We demonstrated that the manual labor required for realizing such a system can significantly be reduced by utilizing automatic soft color segmentation. We presented various results captured using our prototype application, in addition to a live capture video provided as supplemental material. We believe that our production pipeline can be improved further by creating mesh colliders automatically from the alpha channel of the corresponding layers, therefore creating a fully automated process that can be easily deployed in any museum or exhibition.



Figure 6. Screen captures of our application when used on a hand-held device for recoloring multiple paintings

REFERENCES

- Yağız Aksoy et al. 2016. Interactive High-Quality Green-Screen Keying via Color Unmixing. *ACM Trans. Graph.* 35, 5 (2016), 152:1–152:12.
- Yağız Aksoy et al. 2017. Unmixing-Based Soft Color Segmentation for Image Manipulation. *ACM Trans. Graph.* 36, 2 (2017), 19:1–19:19.
- Tunç Ozan Aydın et al. 2015. Predicting display visibility under dynamically changing lighting conditions. *Computer Graphics Forum (Proc. EUROGRAPHICS)* (2009), vol 28, pp 173-182.
- Mark Billinghurst et al. 2015. A Survey of Augmented Reality. *Foundations and Trends in Human-Computer Interaction* 8, 2-3 (2015), 73–272.
- Kuo-En Chang et al. 2014. Development and behavioral pattern analysis of a mobile guide system with augmented reality for painting appreciation instruction in an art museum. *Computers & Education* 71 (2014), 185 – 197.
- Kaiming He et al. 2013. Guided Image Filtering. *IEEE Trans. Pattern Anal. Mach. Intell.* 35, 6 (2013), 1397–1409.
- J. Keil et al. 2013. A digital look at physical museum exhibits: Designing personalized stories with handheld Augmented Reality in museums. In *Digital Heritage International Congress*.
- S. Magnenat et al. Live Texturing of Augmented Reality Characters from Colored Drawings. *IEEE Trans. Vis. Comput. Graphics* 21, 11 (2015), 1201–1210.
- T. Miyashita et al. 2008. An Augmented Reality Museum Guide. In *Proc. ISMAR*.
- Tai et al. 2007. Soft Color Segmentation and Its Applications. *IEEE Trans. Pattern Anal. Mach. Intell.* 29, 9 (2007), 1520–1537.
- Jianchao Tan et al. 2016. Decomposing Images into Layers via RGB-space Geometry. *ACM Trans. Graph.* 36, 1 (2016), 7:1–7:14.
- Rafal Wojciechowski et al. 2004. Building Virtual and Augmented Reality Museum Exhibitions. In *Proc. Web3D*.
- Feng Zhou et al. 2008. Trends in augmented reality tracking, interaction and display: A review of ten years of ISMAR. In *Proc. ISMAR*.
- Fabio Zünd et al. 2015. Augmented Creativity: Bridging the Real and Virtual Worlds to Enhance Creative Play. In *SIGGRAPH Asia Mobile Graphics and Interactive Applications*, Vol. 1, No. 1, Article 1. Publication date: April 2017.