Excel@FIT 2015
http://excel.fit.vutbr.cz

Mobile Application for Real-Time Interactive Montage

Petr Palata*

Abstract
The aim of this paper is to introduce an application for mobile phones for real-time interactive montage. To be able to provide a preview of the segmentation to the user in real-time on a mobile phone, a fast segmentation algorithm must be introduced. We address this issue by proposing a segmentation algorithm which runs in real-time on mobile devices and is suitable as a preprocessing step to more accurate but slower segmentation algorithms. We also elaborate on solving user-experience issues connected to user interface design of the mentioned application.

Keywords: Fast Segmentation — Interactive Montage — Mobile Application

Supplementary Material: *N/A*

*xpalat06@stud.fit.vutbr.cz, *Faculty of Information Technology, Brno University of Technology*

1. Introduction

On of the most common uses of image matting is in creation of a photographic montage where a selected foreground object is pasted into a new scene.

However, it is very hard to create such a montage for regular user without the knowledge of image editing software. And even if the user has experience with this category of software, it still remains a very time consuming and tedious task. Furthermore, the software that provides image matting functionality is almost exclusively commercial and usually takes significant amount of time to learn properly.

Previous approaches to still-image matting for example include GraphCut, which minimizes Gibbs energy by finding a cut with min-cut algorithm [1]. GrabCut further expands on this by repeating this process iteratively on previously obtained segmentations [2]. [3] and [4] expand the matting problem to video and exploit additional information available from a video sequence and adapts intrinsic parameters to provide better segmentation.

The problem with these algorithms is their focus on accuracy and the negligence of speed. For these reasons, they are not suitable for real-time video processing.

We propose an application which offers a possibility to create a montage prototype in a matter of seconds on a smartphone without the need of previous exposure to image editing software. Application with this capability could be useful in a wide range of applications such as photographics composition prototyping, furniture placement prototyping, interactive clothing preview or creating a funny montage.

Since the application needs an algorithm that is capable of real-time segmentation and can be used as an input to a more precise segmentation algorithm, we propose an implementation of segmentation algorithm than runs in real-time on a smartphone.

We also address user-experience challenges such as the minimization of users frustration in case the

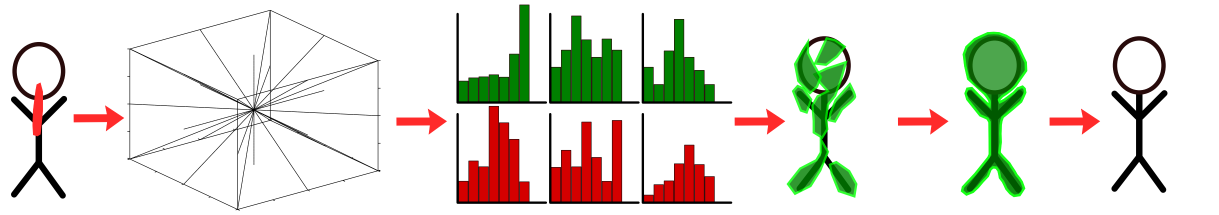

Figure 1. (1) The user highlights foreground object by making a scribble. (2) Projection lines in RGB spaces (3) Obtained histograms (4) After per pixel classification (5) Morphological closing (6) Largest blob is marked as the foreground object

segmentation does not yield usable results.

2. Proposed Application

An algorithm for real-time segmentation is proposed in this section along with an implementation of a user interface to this algorithm in a mobile application. Both the algorithm overview and the implemented application workflow is discussed in detail in the following chapters.

2.1 Segmentation algorithm overview

Our proposed algorithm has several steps. First step is extraction of foreground pixels from a scribble. Background pixels are sampled from a thin line around the image border. Obtained pixels are projected onto 13 lines in RGB space as described in [5]. Each foreground projection is then compared with the corresponding background histogram and only few histograms which can differentiate between foreground and background are picked.

Search for foreground object in each frame starts with projecting each pixel onto lines selected during the training phase. A simple distance from a line histogram is computed and the resulting class is the one with the smallest distance. A few iteration of morphological closing is executed to create larger blobs in the foreground mask. The largest blob is then declared as foreground and the remaining are dropped.

2.2 Application Workflow

Workflow of our application consists of a few easy steps. First, the user makes a scribble over an object representing the foreground with a finger. A segmentation preview is presented in real-time and the user decides if it meets the requirements and proceeds to the selection of the background image. The segmentation preview was selected as the first step because it helps to minimize the frustration and the energy spent in case the segmentation does not work as expected.

After the background image is selected, an editing interface where background adjustments such as translation, rotation or scale can be made with commonly known touch gestures is presented to the user. The segmentation result from the first step is shown along with the background to provide immediate preview of the resulting montage.

When the user makes necessary adjustments and confirms, the preliminary segmentation mask is used as an input to a more precise but slower GrabCut segmentation. The described workflow is depicted in Figure 2.

3. Results

Several experiments in various conditions were conducted with the applications prototype. The scope of the experiments ranged from a well color separated background up to a significant color similarity between the segmented object and the background.

Most of the experiments were conducted ad-hoc using the prototype on a mobile phone. Appropriate amount of screenshots were obtained from several testing sessions. A sample from screenshots obtained during testing is presented in figure 3.

Figure 3. Segmentation results

The segmentation algorithm yields best results in case when the object is not glossy, the user marks pixels that provide good description of the objects color distribution and the background color model is well separated in our model space. Furthermore, the algorithm performs well even on background with complex texture when the previously mentioned conditions are met.

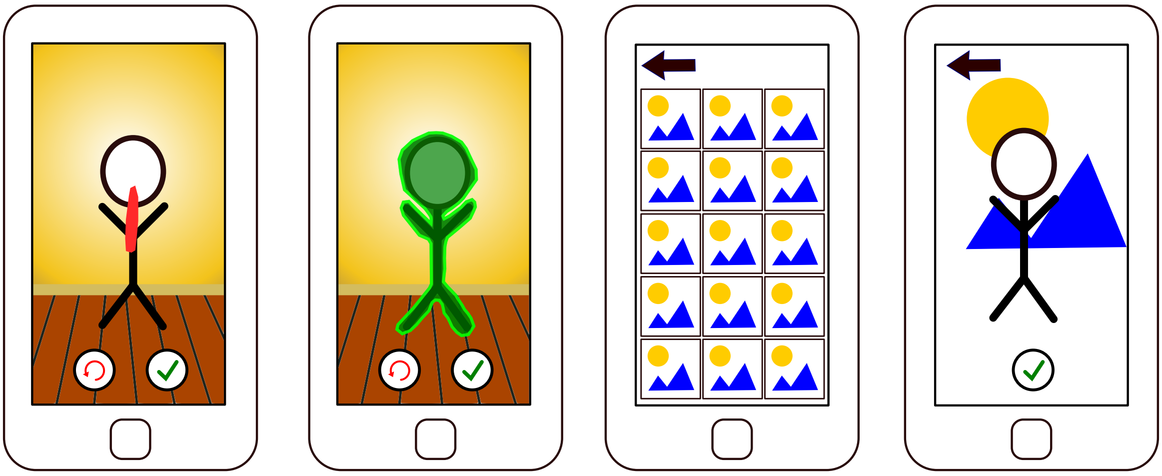

Figure 2. (a) User makes a scribble with his finger (b) Segmentation preview (c) Background image picker (d) Background adjustment

On the other hand, if the user does not highlight a representative pixel sample or the object contains a wide spectrum of colors, the algorithm does not bear convincing results. Algorithm also delivers poor results when trying to segment glossy objects because the white color present in reflections shifts the model to imprecise values.

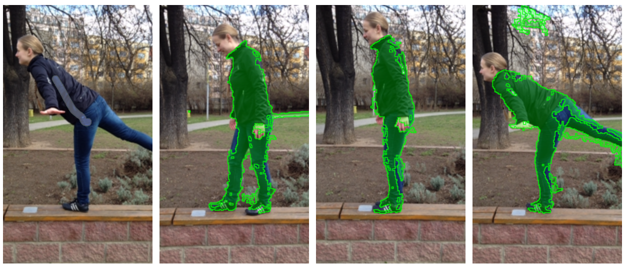

Figure 4. Segmentation of a moving object in a sequence of consecutive frames

An important aspect of our application is the segmentation stability of a moving object in a sequence of consecutive frames. The experiments showed that the segmentation is robust to objects movement in most cases as seen in figure 4. However, if light conditions change rapidly, segmentation errors can appear as seen also in fourth frame of the figure 4.

4. Conclusion

This article proposed an application for interactive montage along with algorithm for real-time video segmentation capable of semi-accurate results with as little information as one user scribble. The algorithm was shown to run in simple enough cases where foreground and background has sufficient distance in RGB space. The application was implemented on an iPhone 5 with A7 chip and runs within 10-15 frames/s range with 352x288px frame size.

Future work will be focused on improving the algorithm for more complex segmentation scenarios and collecting feedback from users about their experience.

References

[1] Yuri Boykov and Gareth Funka-Lea. Graph cuts and efficient nd image segmentation. *International journal of computer vision*, 70(2):109–131, 2006.

[2] Carsten Rother, Vladimir Kolmogorov, and Andrew Blake. Grabcut: Interactive foreground extraction using iterated graph cuts. *ACM Transactions on Graphics (TOG)*, 23(3):309–314, 2004.

[3] Jian Sun, Weiwei Zhang, Xiaoou Tang, and Heung-Yeung Shum. Background cut. In *Computer Vision–ECCV 2006*, pages 628–641. Springer, 2006.

[4] Jue Wang, Pravin Bhat, R Alex Colburn, Maneesh Agrawala, and Michael F Cohen. Interactive video cutout. *ACM Transactions on Graphics (TOG)*, 24(3):585–594, 2005.

[5] Yichen Wei, Jian Sun, Xiaoou Tang, and Heung-Yeung Shum. Interactive offline tracking for color objects. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, pages 1–8. IEEE, 2007.