Spectral Video Matting

Martin Eisemann, Julia Wolf and Marcus Magnor

Computer Graphics Lab, TU Braunschweig, Germany
Email: eisemann@cg.tu-bs.de, wolf@tu-bs.de, magnor@cg.tu-bs.de

Abstract

We present a new, simple-to-use and rapid approach to video matting, the process of pulling a high-quality alpha matte from a video sequence. Our approach builds upon techniques in natural image matting, namely spectral matting, and optical flow computation. No additional hardware, despite a single camera, is needed, and only very few and intuitive user interactions are required for foreground estimation. For certain scenes the approach is able to estimate the alpha matte for a video, consisting of up to 102 frames, without any user interaction at all.

1 Introduction

In digital matting a foreground object along with an opacity estimate for each object pixel is extracted. This gives the user the possibility to seamlessly insert new elements into the scene, e.g., an actor can be recorded and later pasted into a different scene. Such techniques are frequently used in commercial television or film production. A recent survey can be found in [26].

The most simple matting technique is arguably *blue screen matting*, also known as *chroma keying*. Foreground elements are recorded in front of a solid color background, and a number of heuristics are used to extract the matte from each frame [20]. While being fairly effective, the method requires a controlled studio environment.

More sophisticated methods such as *natural image matting* do not impose any restrictions on the background. However, the problem becomes inherently under-constrained and additional information in form of a trimap [7, 21, 9], fore- and background scribbles [24, 10, 15] or tracing along the edges between fore- and background [22] is required.

Our work is mostly inspired by the *spectral matting* approach by Levin *et al.* [16]. Spectral matting extends the ideas of *spectral segmentation* [27, 19, 12]. The real-valued matting components are obtained via a linear transformation of the smallest eigenvectors of the *matting Laplacian* matrix [15]. These matting components are then combined to form the complete foreground matte.

While object cutout in still images has more or less been solved, *video matting* remains a challenging problem. In video matting the task is to estimate the foreground matte of each frame of a video without the need for the user to edit each frame and without introducing temporal artifacts. Besides blue screen matting, a similar approach is *difference matting* [13] where the mapping of the difference between the recorded scene and a background shot yields the opacity values. In *rotoscoping* a user draws an editable curve, like a B-spline, around the foreground element at selected keyframes, often with the aid of snapping tools that are auto-aligned along high gradient areas [8, 18, 4, 1]. These are then interpolated between the keyframes. However, a lot of manual adjustment is required to pull a high-quality silhouette, plus the matte is only binary and usually does not provide any alpha values for blending.

Quite a lot of different directions have been explored recently in the field of video matting. Graph cut segmentation has been extended to work directly on the 3D video volume [17, 23, 2] and spatially varying color models have been tracked [28]. The recently published *Video SnapCut* by Bai *et al.* [3] combines a set of local classifiers with a coherent matting approach to achieve high-quality results with the possibility for local refinements. Still it requires considerable user-interaction for longer sequences.

A common technique to propagate a segmentation result over time is to use optical flow [4, 6, 3]. Unfortunately relying on optical flow can introduce accumulation errors which takes away the reliability of the estimation, forcing user intervention to guide

the algorithm. Algorithms presented in the literature reported that typically the alpha matte for up to a dozen frames on the average without the need for user interaction can be pulled [6]. In contrast, our approach reinitializes the foreground estimation on a per-frame basis, enabling for less error-prone propagation. In theory, it should be possible to pull a high-quality alpha matte for a complete video sequence without any user interaction at all, using unsupervised matting [16]. The main contribution of this paper, however, is finding and adapting a good set of existing algorithms to devise a combined framework, for optimal results with very few user-interaction.

The remainder of the paper is organized as follows. We review spectral matting for still images (Section 2). We then describe our video matting approach in detail (Section 3). Next, we show results of applying our video matting approach to several test scenes (Section 4). Finally, we conclude with a short summary and a number of future research directions (Section 5).

2 Spectral Matting

The *compositing equation* (1) describes the digital matting process as a linear combination of foreground color F and background color B in every pixel i:

$$I(i) = \alpha(i)F(i) + (1 - \alpha(i))B(i), \qquad (1)$$

where $I(i)$ is the pixel color at position $i \in \{1, \ldots, N\}$, $\alpha(i) \in [0, 1]$ is the alpha matte value at i and N is the number of pixels in the image.

Spectral matting by Levin *et al.* [16] generalizes this idea to multiple layers:

$$I(i) = \sum_{k=1}^{K} \alpha_k(i)F_k(i), \qquad (2)$$

with $\sum_{k=1}^{K} \alpha_k(i) = 1$ and $\alpha_k(i) \in [0, 1]$. Where K is the number of layers F_k and α_k are the different matting components encoding the influence of F_k at each pixel i. Despite its proven high-quality (see [16] for a comparison to other methods) the real benefit of spectral matting for video matting is the decomposition into K alpha matting components α_k. All that is needed to obtain the desired foreground is to specify the components belonging to this. Suppose $\alpha_{k_1} \ldots \alpha_{k_n}$ are designated as belonging to the foreground, then the complete matte M is obtained by simply adding them together $M = \alpha_{k_1} + \ldots + \alpha_{k_n}$. An example is given in Figure 1, where the red marked components are added together for the final foreground estimation. Additionally an unsupervised matting can be used to fully automatically estimate a complete foreground out of the matting components, based on balanced cuts [14] and the *matting Laplacian* defined in [16].

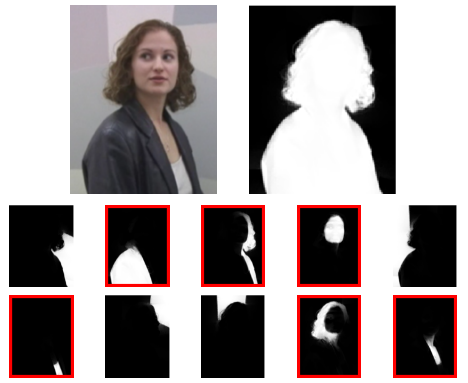

Figure 1: Top left: The input image. Top Right: The estimated alpha matte M (the same matte would have been proposed using unsupervised matting in this case). Bottom: The matting components α_k. The components forming the foreground are marked in red.

3 Spectral Video Matting

In the following our approach is presented, which makes use of the characteristics of spectral matting, and extends it to video matting with minimal user input. In a preprocessing stage, the matting components for the complete video sequence are extracted using the spectral matting technique [16]. As the outcome is highly dependent on the number K of matting components used, we exemplarily estimate a good number for the first frame. For cutting out certain foreground objects out of a multitude of possible objects the user has to decide which components α_k should be part of the foreground. The sum of these α_k forms the matte M^0 for the first frame. We usually set the number of clusters to lie between 10 and 20 for our test scenes, therefore the foreground clusters can be chosen very quickly in most

cases. A simple scribble interface, as proposed in [16], could be incorporated for selection, if more clusters are needed for a complex scene. The best performance is achieved if the number of clusters is as small as possible, still fulfilling the constraint that two distinct objects do not share a cluster, because otherwise a separation would be impossible. Apart from that, this step is the only user interaction that is needed for the algorithm to start. For some simpler scenes even this initial foreground estimation can be automated by using unsupervised matting as described in [16].

Using this initial set of matting components, the foreground is propagated through the video volume using optical flow. Given two neighbouring frames $I^{(j-1)}$ and I^j plus the matte $M^{(j-1)}$ for the $(j-1)$-th frame, we can compute a relation between $I^{(j-1)}$ and I^j to satisfy the following equation:

$$I^j = W_{I^{(j-1)} \rightarrow I^j} \circ I^{(j-1)} \quad , \tag{3}$$

where $W_{I^{(j-1)} \rightarrow I^j} \circ I^{(j-1)}$ warps an image $I^{(j-1)}$ towards I^j according to the warp field $W_{I^{(j-1)} \rightarrow I^j}$. The problem of determining this warp field $W_{I^{(j-1)} \rightarrow I^j}$ is known as optical flow estimation. In our case we are not interested in warping the image itself to the next frame but rather the alpha matte. We compute an initial guidance G^j for the foreground matte of frame I^j by warping $M^{(j-1)}$ using the warp field $W_{I^{(j-1)} \rightarrow I^j}$:

$$G^j = W_{I^{(j-1)} \rightarrow I^j} \circ M^{(j-1)} \tag{4}$$

To estimate the foreground matte M^j of frame I^j we search for the combination of precomputed foreground clusters α_k which minimizes the difference between the new alpha matte M^j and G^j in a least-squares sense. Therefore the task is to minimize the following error function:

$$E = ||G - \sum_{k=1}^{K} b_k \alpha_k||, \quad b_k \in \{0, 1\} \tag{5}$$

where b_k is the binary solution vector we solve for. We found that a simple greedy approach solves this optimization problem well in all our encountered test cases. Starting with an empty initial estimate for M^j, we assume there is no foreground in the image and all α_k are 0. We then add the single matting component α_k to our solution which reduces the error function the most and set b_k to 1. The process is repeated until E converges to a minimum.

As the clusters for the new image have been computed beforehand and are independent of the solution of the previous frame, this method works very robust even in the case where the optical flow cannot compute very precise warp fields, see Figure 2 for a comparison. Also disocclusion, i.e. newly appearing regions, which are usually a big problem, can be handled robustly if the disoccluded parts belong to the same cluster as already visible parts.

Repeating the described process of warping the previous matte to the current frame and reestimating the foreground for the following frames computes the alpha matte for the whole video sequence. The computation of the optical flow could also be computed in the preprocessing stage, but as fast optical flow implementations exist, this would only waste storage space. To prove the inherent robustness due to the reinitialization of the foreground using the matting components, we used a simple block-matching method [11], beeing aware that better optical flows exist, which could be incorporated in future versions.

In some situations, it is possible that certain changes of the shape of the foreground object cannot be identified automatically. This situation occurs because the optical flow algorithm is only suitable for small and smooth motions and not for strong changes in the shape of the foreground. Most of the times the reestimation using the matting components handles also imprecise flows, but if the error becomes too large manual adjustment of the user is necessary. In this case, the user scans the results until he finds a frame which has an incorrect estimated foreground. He then adds a new keyframe by reinitializing the foreground clusters manually, as done for the first frame. The algorithm then recomputes the rest of the video with the new foreground estimation. If the content of a scene changes drastically throughout the video, it might be helpful to not only reinitialize the foreground clusters, but also to change the number of cluster or eigenvalues, this could also easily be incorporated into the framework, as the computation can be done in the preprocessing stage.

4 Results and Discussion

Our test PC used for the evaluation is equipped with an Intel Core2Duo (2.40GHz), only one core used, with 2GB of RAM. As test scenes we present

sequence	resolution	frames	edited frames	matting components per frame	frames/sec
Amira	360×240	66	1	10	1.23
Kim	360×240	102	0	10	1.03

Table 1: Details for the used test sequences

two commonly known sequences, namely Amira (Fig. 1) and Kim (Fig. 2). Both contain complex foreground shapes and motion, like hair, plus a non-static background and occlusion and disocclusion is apparent.

In a pre-processing phase the spectral clusters are computed and saved to disk, which, using the unoptimized algorithm, takes approximately 5 minutes for the computation of a 360×240 frame. For the interactive online phase, specifying the foreground clusters took about 10 seconds for a trained user while editing further keyframes takes even less time. Therefore the user interaction involved in matting each of our test scenes took less than one minute for the complete Amira video, consisting of 66 frames. The Kim sequence, consisting of 102 frames was computed completely automatic without any user interaction at all. The initial alpha matte was automatically estimated using the best hypothesis from the unsupervised matting [16]. The results for both sequences are included in the accompanying video.

The computation of the optical flow plus warping of the alpha matte and optimization for the next matte took less than one second per frame. Generally the spectral matting performs very well even for complex structures such as hair (for a detailed comparison to other methods see [16]). Very fast movements or newly appearing foreground objects usually require some user interaction, as the optical flow will fail in these cases. The number of edited frames, along with the computation times are given in Table 1.

5 Conclusion

In this paper, we have presented a new, easy to use technique for pulling mattes of video foregrounds having complex silhouettes. Our approach is based on different methods and combines their individual strengths. By combining spectral matting and optical flow, we obtain high quality mattes for different recorded video scenes. We introduce a simple, yet efficient way to propagate the alpha matte information to successive frames. We optimize the new foreground matte to prevent accumulation errors. A limitation to our current approach is a relatively large memory footprint, due to the spectral matting algorithm, and that we did not include more sophisticated user interaction techniques in cases the spectral matting fails to estimate good clusters, but we plan to integrate fallbacks for these cases in the future. One way to diminish these errors could be temporal filtering of the alpha mattes using the flow fields to find the temporal neighbors in successive frames. We also intend to incorporate a GPU version of the spectral matting process and the optical flow computation, including a temporal smoothness constraint, in order to remove the preprocessing stage and possible temporal artifacts completely. Background estimation as well as back propagation of the optical flow [6] can further improve the results. As a final goal, we aim for developing a complete tool with an even more intuitive user-interface that could be tested and improved by rotoscoping artists.

6 Acknowledgements

We would like to thank Yung-Yu Chuang from the National Taiwan University for providing us with the videos shown in this paper.

References

[1] Aseem Agarwala, Aaron Hertzmann, David H. Salesin, and Steven M. Seitz. Keyframe-based tracking for rotoscoping and animation. In *SIGGRAPH 2004*, pages 584–591, New York, NY, USA, 2004. ACM.

[2] Christopher J. Armstrong, Brian L. Price, and William A. Barrett. Interactive segmentation of image volumes with live surface. *Computer Graphics*, 31(2):212–229, 2007.

[3] X. Bai, J. Wang, D. Simons, and G. Sapiro. Video snapcut: Robust video object cutout us-

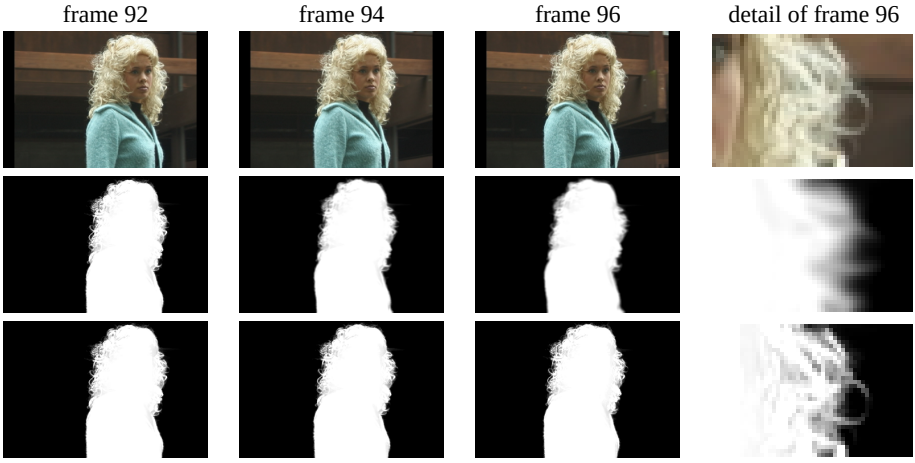

Figure 2: The potential to reinitialize the warped alpha matte due to the spectral matting components improves the final result in the alpha matte estimation for successing frames and prevents accumulation errors. For frames 92 to 96 of the Kim sequence (shown above) the alpha matte is provided for the first frame. Warping the alpha matte in forward direction yields mattes with an increasing error (second row). Reinitializing the alpha mattes with our proposed techniques shows improved results without visible accumulated errors (bottom row).

ing localized classifiers. In *SIGGRAPH 2009*, New York, NY, USA, 2009. ACM.

[4] Andrew Blake and M. Isard. *Active Contours: The Application of Techniques from Graphics,Vision,Control Theory and Statistics to Visual Tracking of Shapes in Motion*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 1998.

[5] Gary Bradski and Adrian Kaehler. *Learning OpenCV: Computer Vision with the OpenCV Library*. O'Reilly Media, 1 edition, 10 2008.

[6] Yung-Yu Chuang, Aseem Agarwala, Brian Curless, David H. Salesin, and Richard Szeliski. Video matting of complex scenes. In *SIGGRAPH 2002*, pages 243–248, New York, NY, USA, 2002. ACM.

[7] Yung-Yu Chuang, Brian Curless, David H. Salesin, and Richard Szeliski. A bayesian approach to digital matting. In *IEEE CVPR 2001*, volume 2, pages 264–271. IEEE Computer Society, December 2001.

[8] Michael Gleicher. Image snapping. In *SIGGRAPH 1995*, pages 183–190, New York, NY, USA, 1995. ACM.

[9] Leo Grady, Thomas Schiwietz, Shmuel Aharon, and Rüdiger Westermann. Random walks for interactive alpha-matting. In J. J. Villanueva, editor, *Proceedings of the Fifth IASTED International Conference on Visualization, Imaging and Image Processing*, pages 423–429, Benidorm, Spain, Sept. 2005. ACTA Press.

[10] Yu Guan, Wei Chen, Xiao Liang, Zi'ang Ding, and Qunsheng Peng. Easy matting - a stroke based approach for continuous image matting. *Computer Graphics Forum*, 25(3):567–576, 9 2006.

[11] Yan Huang and Xinhua Zhuang. Motion-partitioned adaptive block matching for video compression. In *ICIP '95: Proceedings of the 1995 International Conference on Image Processing (Vol. 1)-Volume 1*, page 554, Washington, DC, USA, 1995. IEEE Computer Society.

[12] Stella Yu Jianbo, Stella X. Yu, and Jianbo Shi. Multiclass spectral clustering. In *In International Conference on Computer Vision*, pages 313–319, 2003.

[13] D Kelly. *Digital Composition*. The Coriolis Group, 2000.

[14] Kevin Lang. Fixing two weaknesses of the spectral method. In Y. Weiss, B. Schölkopf, and J. Platt, editors, *Advances in Neural In-*

formation Processing Systems 18, pages 715–722. MIT Press, Cambridge, MA, 2006.

[15] Anat Levin, Dani Lischinski, and Yair Weiss. A closed-form solution to natural image matting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):228–242, 2008.

[16] Anat Levin, Alex Rav-Acha, and Dani Lischinski. Spectral matting. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[17] Yin Li, Jian Sun, and Heung-Yeung Shum. Video object cut and paste. In *SIGGRAPH 2005*, pages 595–600, New York, NY, USA, 2005. ACM.

[18] Eric N. Mortensen and William A. Barrett. Intelligent scissors for image composition. In *SIGGRAPH 1995*, pages 191–198, New York, NY, USA, 1995. ACM.

[19] Andrew Y. Ng, Michael I. Jordan, and Yair Weiss. On spectral clustering: Analysis and an algorithm. In *Advances in Neural Information Processing Systems 14*, pages 849–856. MIT Press, 2001.

[20] Alvy Ray Smith and James F. Blinn. Blue screen matting. In *SIGGRAPH 1996*, pages 259–268, New York, NY, USA, 1996. ACM.

[21] Jian Sun, Jiaya Jia, Chi-Keung Tang, and Heung-Yeung Shum. Poisson matting. *SIGGRAPH 2004*, 23(3):315–321, August 2004.

[22] Jue Wang, Maneesh Agrawala, and Michael F. Cohen. Soft scissors: an interactive tool for realtime high quality matting. In *SIGGRAPH 2007*, page 9, New York, NY, USA, 2007. ACM.

[23] Jue Wang, Pravin Bhat, R. Alex Colburn, Maneesh Agrawala, and Michael F. Cohen. Interactive video cutout. In *SIGGRAPH 2005*, pages 585–594, New York, NY, USA, 2005. ACM.

[24] Jue Wang and Michael F. Cohen. An iterative optimization approach for unified image segmentation and matting. In *ICCV 2005: IEEE International Conference on Computer Vision*, pages 936–943, Washington, DC, USA, 2005. IEEE Computer Society.

[25] Jue Wang and Michael F. Cohen. Image and video matting: a survey. *Found. Trends. Comput. Graph. Vis.*, 3(2):97–175, 2007.

[26] Jue Wang and Michael F. Cohen. *Image and Video Matting*. Now Publishers Inc., Hanover, MA, USA, 2008.

[27] Yair Weiss. Segmentation using eigenvectors: A unifying view. In *In International Conference on Computer Vision*, pages 975–982, 1999.

[28] Ting Yu, Cha Zhang, Michael Cohen, Yong Rui, and Ying Wu. Monocular video foreground/background segmentation by tracking spatial-color gaussian mixture models. In *Proceedings of the IEEE Workshop on Motion and Video Computing*, page 5, Washington, DC, USA, 2007. IEEE Computer Society.