A Novel Approach to Video Matting using Optical Flow

Mikhail Sindeyev[1], Vadim Konushin[1,2]
[1] Graphics and Media Lab, Moscow State Lomonosov University, Moscow, Russia
[2] Keldysh Institute of Applied Mathematics, Russian Academy of Sciences
E-mail: {msindeev, vadim}@graphics.cs.msu.ru

Abstract

The image matting problem refers to foreground object extraction from an image. Similarly, video matting problem refers to extraction of a foreground object from each frame of a video-sequence producing a moving foreground layer. Layer extraction process should deal with the transparency caused by camera point spread function (PSF) and motion blur, thus the natural way is to store extracted layer as a pair of images: color and opacity. The latter is referred to as an alpha-channel.

In this paper we propose an algorithm that takes alpha for the first frame (or an arbitrary key-frame) provided by the user (e.g. by using some image matting algorithm) and tracks its motion in time through the video sequence.

Unlike the obvious approach that models the whole scene motion from one frame to the next with an optical flow (e.g. to warp the rough input segmentation (trimap) or the alpha channel with this flow), we use it to model the foreground layer motion only. This prevents us from otherwise unavoidable artifacts on the boundaries of foreground objects, which are of the main interest in the matting problem.

Instead of matching pixel colors in the pair of frames, we measure how the optical-flow-warped alpha-channel fits the next frame (we use color matching as an additional regularization though). By minimizing this measure with respect to optical flow we fully preserve the structure of the foreground object and prevent temporal incoherence artifacts.

Keywords: *video matting, optical flow, digital compositing, foreground extraction.*

Figure 1. *Harbor* video sequence from [3]. **Top**: initial user-specified mask for frame #0. **Middle**: frame #30. **Bottom**: generated mask for frame #30 by forward-tracking the mask from frame #0 using the proposed algorithm.

1. INTRODUCTION

In this paper we address the problem of extraction of a moving foreground object from a natural video sequence. This is referred to as a video matting problem. Similarly to the image matting problem, in each frame a source image C is assumed to be a composite of two image layers F and B (foreground and background) with opacity channel α. In each pixel their RGB values should satisfy *the compositing equation*:

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

where C, F and B are 3D vectors of RGB values, $0 \le \alpha \le 1$. The problem is to reconstruct the α, F and sometimes B images from the source image C using some additional user input. An example of such input is trimap – a rough segmentation of the image into foreground ($\alpha=1$), background ($\alpha=0$) and unknown region (where α is to be reconstructed by the algorithm).

The goal is usually to put the foreground layer onto a new background or process foreground and background layers separately using some image filter.

According to [11] video matting has the following challenges in comparison to image matting:

- Large data size (width × height × duration vs. width × height)
- Temporal coherence: video frames may look well when observed independently, but produce notable motion artifacts
- Fast motion vs. low temporal resolution: it may be difficult to automatically identify motion between frames at typical frame rates (15-30 FPS for most cameras)

and has the following advantages:

- Ability to create more complete model of background by analyzing its motion over a range of frames
- Possibility to detect foreground/background edge when part of the background is covered up or revealed as a result of the background or object motion.

2. PREVIOUS WORK

Despite of the recent success in the image matting field (Closed-Form Solution [7], RobustMatting [12]), video matting is poorly developed. Only simple approaches exist. We overview them below.

Obvious extension of image matting to video matting is to process all frames independently using some image matting algorithm. Additional input information used by the algorithm (such as trimap or user strokes) however can be interpolated in some way (either of itself or by using motion from the video sequence). An

example of such method is [3], where a trimap is interpolated using optical flow, after which video frames are matted with Bayesian Matting algorithm independently.

Disadvantages of the independent frame matting are:

- Temporal incoherence
- Needs much user work to provide enough input for each frame

Another approach is to treat video volume as a 3-D image. This is typically used with binary segmentation methods which perform boundary matting as a post-processing. 3-D volume is usually oversegmented first and at this step the spatial and temporal dimensions may be treated asymmetrically. An example of such method is [8]. Another method [2] is fully automatic but works only on simple videos with distinct separation of foreground and background layers by their motion.

Disadvantage of this method is a lack of consistent motion model: parts of moving objects in one frame do not necessarily match to the same parts in the next frame – they tend to disappear and reappear rapidly in the result mask when the motion is fast. The definition of 'fast' here assumes that these parts of object do not overlap (or have little overlap) in two consecutive frames when projected onto one frame along the temporal axis.

3. PROPOSED ALGORITHM

3.1 Workflow overview

We start from a known alpha-channel for the first frame. It can be produced by a user using any existing image matting algorithm. Then we try and deform it with a smooth optical flow, until we get the best match for the next frame. We process consequent frames in the same way.

In the proposed scheme input data during processing the i-th frame consists of just alpha of the (i-1)-th frame and i-th frame image.

Thus we need a cost function that measures how well the warped alpha-channel fits the current image. In general case we use the Laplacian proposed in [7].

In case of a known background we use a simpler functional of squared difference between left-hand and right-hand parts of the compositing equation (1).

3.2 Energy function

When processing the current frame, we use the following energy:

$$E(V,\alpha) = E_d(V,\alpha) + \lambda E_s(V), \tag{2}$$

where V is the optical flow, α is the alpha-channel of the previous frame (as a column-vector) and λ is a smoothness parameter. α is fixed.

An image I can be warped by an optical flow what we denote by $V(I)$. Each pixel of the optical flow map consists of two components: $V=(V_x, V_y)$. Warping is performed in the following way:

$$V(I)\big|_{(x,y)} = I(x + V_x(x,y), y + V_y(x,y)), \tag{3}$$

where I is an arbitrary image to be warped. For fractional coordinate values we use a bilinear interpolation. Later on we'll refer to pixel index just as i, not as (x,y), thus treating images as vectors.

In our workflow we obtain only V by solving

$$V = \arg\min_V E(V,\alpha) \tag{4}$$

Then an alpha-channel for the current frame is constructed by warping α with the found optical flow V:

$$\alpha_{next} = V(\alpha). \tag{5}$$

The constructed alpha is used to process the next frame.

3.2.1 Laplacian-based data term

The data term uses the Laplacian from the Closed-form Matting algorithm [7]:

$$E_{d1}(V,\alpha) = V(\alpha)^T L V(\alpha). \tag{6}$$

Additionally, we use color information too, weighted by the alpha values:

$$E_{d2}(V,\alpha) = \sum_i V(\alpha)_i \left\| C_i - V(C_{prev})_i \right\|^2, \tag{7}$$

where the sum is taken over all pixels and subscript I refers to the i-th component of vector (that is, the i-th pixel of the image). C and C_{prev} are color images of the current and previous frame, respectively. This data term allows to consider motion in foreground region which also affects boundary via smoothness. Finally,

$$E_d(V,\alpha) = E_{d1}(V,\alpha) + \mu E_{d2}(V,\alpha). \tag{8}$$

3.2.2 Data term for known background

In case of a known background (which can be filmed separately if it's static, or reconstructed by planar tracking, e.g. with Mokey software [6]) we use a simpler functional, that however involves also the foreground color image F. F and α are being warped with the same optical flow, so the compositing equation (1) becomes:

$$C = V(\alpha)V(F) + (1 - V(\alpha))B, \tag{9}$$

where F and α are from the previous frame, while B and C are from the current frame. Data term for the known background case becomes:

$$E_d(V,\alpha) = \left\| C - V(\alpha)V(F) + (1 - V(\alpha))B \right\|^2. \tag{10}$$

After processing each frame, F (to be used in the next frame) is reconstructed from B, C and α using the equation (1), or better by using Bayesian Matting [4] (i.e. F is not being tracked from the very first frame, as is α).

3.2.3 Smoothness term

To regularize the optical flow we use a simple first-order smoothness term:

$$E_s(V) = \sum_i \sum_{j \in w(i)} \left\| V_i - V_j \right\|^2, \tag{11}$$

where $w(i)$ is a 3x3 pixel neighborhood of the i-th pixel.

3.3 Energy minimization

In our implementation we use QPBO method of discrete optimization [9] . We limit the search space at each pixel by a fixed-size window and use alpha-expansion algorithm iteratively to obtain a labeling for each pixel, where the label is optical flow vector $V=(V_x,V_y)$.

3.4 Hierarchical processing

The search-space of the optical flow estimation problem can be reduced by starting at lower resolution and then upscaling the intermediate results while using small search-window at each scale (e.g. 3x3 pixels).

3.5 Refinement

The result of tracking can be refined by applying Bayesian Matting with smoothness [10] to prevent the accumulation of tracking error and interpolation artifacts. Unknown region to be processed is constructed from all pixels where alpha is not equal 0 or 1. The mean value for the alpha at each pixel is taken from the generated

alpha map (4). The variance depends on the distance between foreground and background color distributions, thus regions with strict color separation are adjusted according to the image information, while in the undetermined regions alpha channel structure is preserved.

4. RESULTS AND COMPARISON

We have tested our algorithm on several videos. Due to the lack of test datasets for video matting, we used example videos from video matting/segmentation papers and created our own keyframe masks. To test known-background functional, we reconstructed background for some of the video sequences using homography tracking (i.e. approximating the background with a plane in 3D-space).

For comparison we used RobustMatting implemented by its authors. It has a video matting feature, which interpolates the trimap using optical flow or color difference (selectable by the user)

The testing showed that the resulting matte quality varies heavily depending on the 'complexity' of the video. The best results were achieved on simple videos from [3]. An example is shown in figure 1. When testing on the videos taken from different papers, our algorithm in most cases gave better result than the existing approaches. Examples are shown in figures 2 and 3.

Some additional videos from various sources demonstrated poor results for all algorithms including the proposed one. The difference on such 'hard' videos is that our algorithm fails to track the mask and it drifts and distorts inadequately, while other approaches produce random semi-transparent regions that quickly cover the entire image during the course of processing.

We also note that adding a known (or reconstructed) background improves the video matting result.

Processing time of our unoptimized C++ implementation of the algorithm is ~3 minutes per frame (640x480) on a 1.8GHz processor.

5. CONCLUSION AND DISCUSSION

Our approach works most well on video segments in which foreground objects do not drastically change their shape. Appearance of new parts of objects usually causes our method to fail.

Some of the examples of motion types for which our algorithm demonstrates noticeable advantages over the earlier approaches are:

- Hair blowing in the wind over complex background: overall hair structure is preserved while earlier approaches tend to catch some foreign background elements that show through the gaps in the hair
- Out-of-image-plane rotation of people arms, heads and bodies: high-speed motion of texture usually occurs near the boundaries of rotating objects while the boundary contour itself doesn't change much and is modeled well enough with smooth optical flow in the alpha-channel

Typical conditions of applicability of a matting algorithm usually include some (local) color separation between foreground and background across the object boundary. In our global minimization approach we weaken this condition: we require either color separation or motion smoothness in different parts of foreground object.

Assume that the left part of some solid object has a distinct edge with good color separation while the right part does not. In our approach the right side contour will most likely be dragged by the left one due to smoothness of the optical flow. Graph cut and trimap interpolation based methods will in contrast produce noisy and chaotic matte on the right side of the object (because they match pixels/segments on both sides of the object independently).

Additionally, our algorithm allows tracking of non-opaque objects to some extent, if their color-blending model satisfies the compositing equation (1).

We cannot however account for topology changes of the object silhouette. Indeed such cases need some information from the user, though simpler cases could be handled semi-automatically by adding some heuristics, which we do not address in our work.

5.1 Future work

Being rather immature, our algorithm has numerous possibilities for improvement, namely:

- Better regularization of optical flow smoothness, e.g. accounting for affine warping
- Using a pair of key-frames (at the both ends of the video segment being processed) and replace the extrapolation problem with the interpolation one to prevent the error accumulation
- Using different approaches of measuring how well the warped alpha-channel fits the image
- Model both background and foreground layers with independent optical flows (the background flow however should somehow be global rather than interframe, because the background layer suffers from occlusions, which may be pretty hard to achieve)
- It may be possible to switch to continuous optimization, similarly to Horn-Schunck method [5], by taking the linear term of the Taylor series for the α (and also for αF for the known-background case)

6. REFERENCES

[1] Agarwala, A., Hertzmann, A., Salesin, D., Seitz, S. *Keyframe-Based Tracking for Rotoscoping and Animation*, Proc. of SIGGRAPH, Vol. 32, No. 3, pp. 584–591, 2004

[2] Apostoloff, N., Fitzgibbon, A., *Automatic video segmentation using spatiotemporal T-junctions*, Proc. of the British Machine Vision Conference, 2006.

[3] Chuang, Y.-Y., Agarwala, A., Curless, B., Salesin, D., Szeliski, R. *Video Matting of Complex Scenes*, ACM Transactions on Graphics, Vol. 21, No. 3, pp. 243–248, 2002

[4] Chuang, Y., Curless, B., Salesin, D. and Szeliski, R.. *A Bayesian Approach to Digital Matting*, Proc. of IEEE CVPR, pp. 264–271, 2001

[5] Horn, B., Schunck, B., *Determining optical flow*, Artificial Intelligence, Vol 17, pp. 185–203, 1981

[6] Imagineer Systems Mokey http://www.mokey.com/products/mokey/

[7] Levin, A., Lischinski, D., Weiss, Y. *A Closed Form Solution to Natural Image Matting*, Proc. of IEEE CVPR, pp. 61–68, 2006

[8] Li, Y., Sun, J., Shum, H.-Y. *Video Object Cut and Paste*. Proc. of SIGGRAPH, Vol. 24, No. 3, pp. 595–600, 2005

[9] Rother C., Kolmogorov V., Lempitsky V., Szummer M. *Optimizing Binary MRFs via Extended Roof Duality*, Proc. of IEEE CVPR, pp. 1–8, 2007

[10] Sindeyev M., Konushin, V. and Vezhnevets, V. *Improvements of Bayesian Matting*, Proc. of Graphicon, pp. 88–95, 2007

[11] Wang, J. and Cohen, M. *Image and Video Matting: A Survey*. Foundations and Trends in Computer Graphics and Vision, Vol. 3, No. 2, pp. 97–175, 2007

[12] Wang, J. and Cohen, M. *Optimized Color Sampling for Robust Matting*, Proc. of IEEE CVPR, pp. 1–8, 2007

About the authors

Mikhail Sindeyev is a 5th year student at Graphics and Media Laboratory of Moscow State Lomonosov University, Department of Computational Mathematics and Cybernetics. His research interests include image and video processing, 3D reconstruction, computer vision and adjacent fields. His email address is msindeev@graphics.cs.msu.ru.

Vadim Konushin is a Ph.D. student at Keldysh Institute of Applied Mathematics, Russian Academy of Sciences. His research interests include image and video processing, pattern recognition, computer vision and adjacent fields. His email address is vadim@graphics.cs.msu.ru.

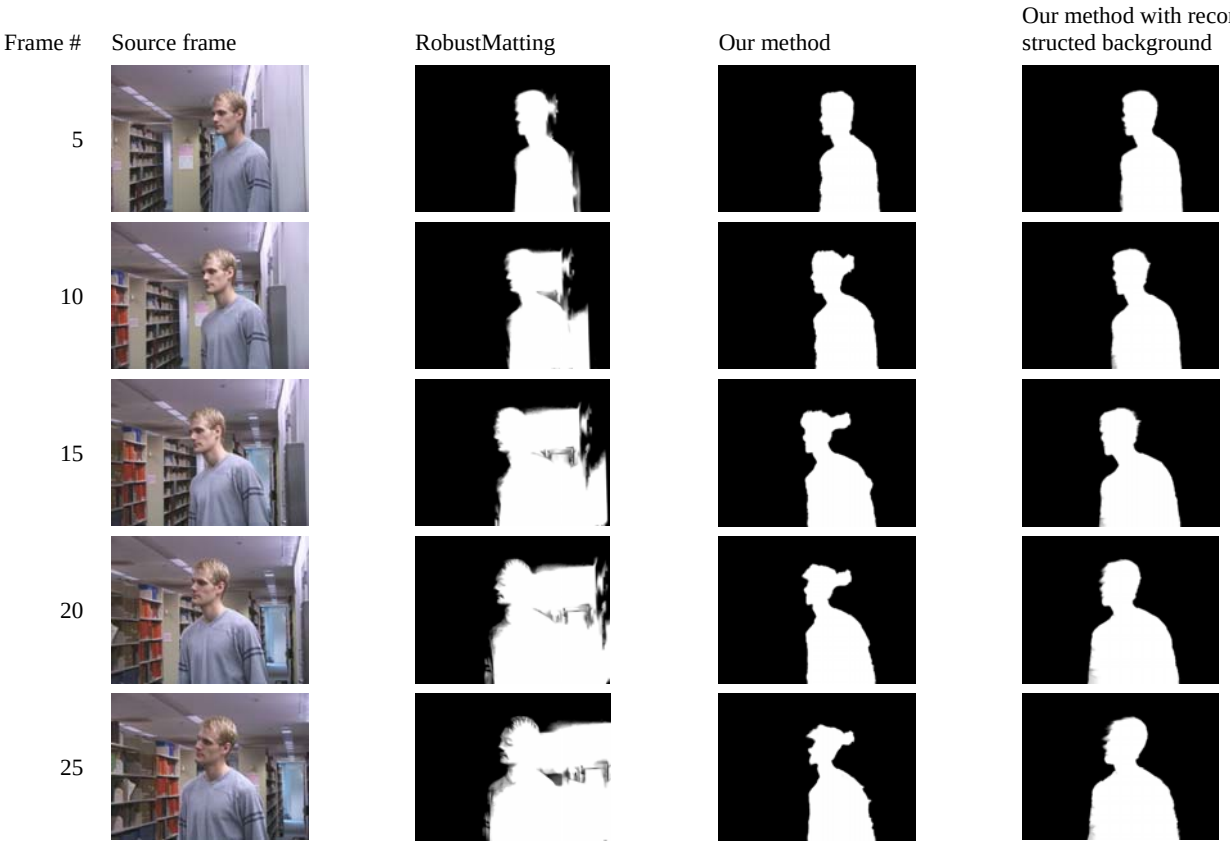

Figure 2. *Adam-lib-walk* videosequence from [1]. Keyframe was created for only the first frame.

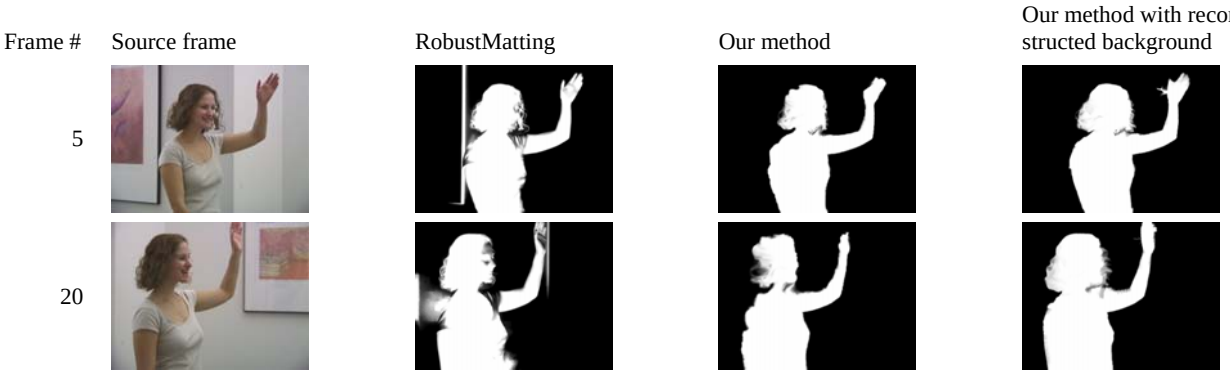

Figure 3. *Amira-queen* videosequence from [1]. Keyframe was created for only the first frame.