Bayesian Video Matting Using Learnt Image Priors

Nicholas Apostoloff and Andrew Fitzgibbon
Robotics Research Group
University of Oxford
Oxford, OX1 4AJ, UK
{nema, awf}@robots.ox.ac.uk

Abstract

Video matting, or layer extraction, is a classic inverse problem in computer vision that involves the extraction of foreground objects, and the alpha mattes that describe their opacity, from a set of images. Modern approaches that work with natural backgrounds often require user-labelled "trimaps" that segment each image into foreground, background and unknown regions. For long sequences, the production of accurate trimaps can be time consuming. In contrast, another class of approach depends on automatic background extraction to automate the process, but existing techniques do not make use of spatiotemporal consistency, and cannot take account of operator hints such as trimaps.

This paper presents a method inspired by natural image statistics that cleanly unifies these approaches. A prior is learnt that models the relationship between the spatiotemporal gradients in the image sequence and those in the alpha mattes. This is used in combination with a learnt foreground colour model and a prior on the alpha distribution to help regularize the solution and greatly improve the automatic performance of such systems.

The system is applied to several real image sequences that demonstrate the advantage that the unified approach can afford.

1 Introduction

Video matting is a technique that is central to special effects in both the film and television industry. In any situation where actors are extracted from footage to be later composited onto an artificial or alternate background, accurate separation of the actor and the background is vital. Video matting is similar to the computer vision problem of layer extraction [1, 2, 9, 14, 15, 16], but has a stronger emphasis on deriving object boundaries which accurately represent the sub-pixel blending of foreground and background layers. This problem is interesting because it is a difficult inverse problem: the number of unknowns exceeds the number of measurements, so regularization is crucial to the success of any automatic solution. Traditionally, the video matting framework relies upon user interaction to assist the process, while layer extraction concentrates on automatic decomposition of the sequence, with less emphasis on sub-pixel accuracy. In this paper, we combine these two strands of research to produce a system which allows accurate mattes to be extracted from video sequences with little or no user intervention. Key features of this paper are threefold: first, in contrast to previous efforts in this area, the prior probabilities which are the key to regularizing the problem are learnt from examples and fit to models from natural image statistics. This means that stronger, more appropriate regularization is applied than has been seen previously. Second, because we employ a consistent MAP estimation framework, the panoply of previous techniques is easily unified in one approach. Finally, because we use well-honed numerical algorithms to solve for the MAP maxima, we have reliable convergence.

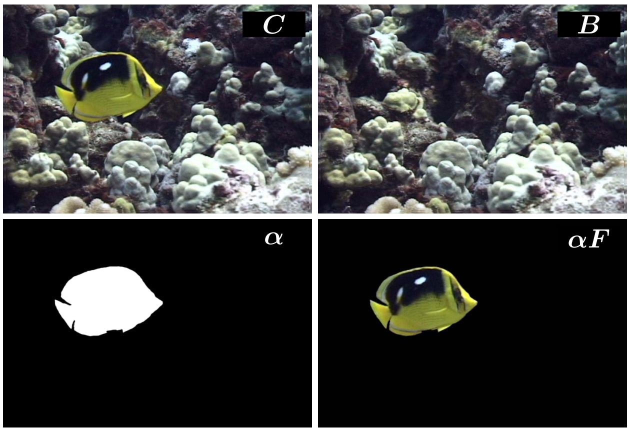


Figure 1: Video matting. Given a sequence of composite images (C) we wish to compute: background (B), opacity information represented as an alpha matte (α) and foreground layer (αF).

The central problem is illustrated in figure 1. The input to video matting is an image C which is a *composite* of a foreground image F and a background image B. The *compositing equation* defines how C is formed as a linear combination of F and B. Each pixel in the composite is

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

where α, the *alpha* or opacity value is a number between zero and one. Given the composite image, the task of video matting is to recover the values of α, B, F at every pixel in the sequence. Given only this statement of the problem, it is apparent that it is hugely under-constrained: at each pixel in a single-channel image, there are three unknowns (α, B, F) and only one equation. For a three-channel colour image, this becomes three simultaneous equations in seven unknowns: a small improvement in the ratio of constraints to variables, but still an ill posed problem. In order to obtain a solution, additional assumptions must be incorporated, and the techniques in the literature are characterized by the assumptions used.

Scientific study of the matting problem may be first associated with Smith and Blinn [12] who examined the traditional blue-screen matting scenario, in which the background B is engineered to be known. They showed that in order to extract a unique alpha matte, the foreground must be seen against at least two different background colours.

Wang and Adelson [15] and Irani et al [9] showed how optic flow information could be used to automatically extract layers from an image sequence. Robust motion segmentation identifies candidate layers which have consistent motion, and then each layer is estimated by warping the sequence so that the layer is stationary. The layer colour is extracted at each pixel using a temporal median. This works well when all layers in the sequence obey a tractable motion model, but on the sequences which we analyse, the foreground objects are typically articulated and do not obey a parametric motion model for long enough to allow the median operation to choose the correct colour. However, the background image B is successfully estimated in the majority of cases, and we used a local implementation of the technique to estimate backgrounds for each of our sequences. Thus, our situation is the analogue of blue-screen matting when some of the foreground may contain blue pixels. Figure 2 illustrates that even with knowledge of the background, the problem remains under-constrained.

The second class of matting techniques loosens the requirement that the background be precisely known, and assumes only that background colours are drawn from a known probability distribution [5, 7, 10]. This assumption leads naturally to statistical inference problems where the background, foreground and alpha values are simultaneously estimated from the composite image. Because such approaches are even less well constrained than the single-view blue-screen matting approach, significant human input is required. This typically takes the form of a *trimap* where pixels are labelled as definitely inside the object of interest $(\alpha = 1)$, definitely outside $(\alpha = 0)$, or unknown. Propagating α from the known to the unknown regions produces high-quality mattes. Chuang et al [4] combine this class of technique with optic flow in order to temporally propagate the trimaps, leading to impressive video-sequence mattes which require a minimum of manual input.

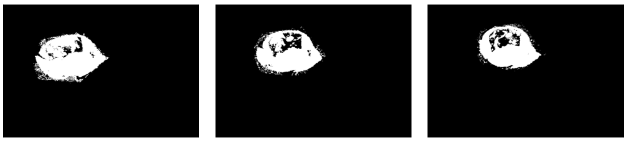


Figure 2: Background subtraction. Restricting to binary α, and with known background B, an estimate of alpha is obtained by thresholding the difference image $|C - B|$. No threshold value yields a correct alpha matte. The introduction of spatial and temporal priors will permit a good solution to be computed.

Although many of the previous approaches are declared to be Bayesian techniques, they eschew one of the most important characteristics of the Bayesian approach: there are no priors. Priors are an important component of any Bayesian solution to an inverse problem as they embody the regularization that is essential if reliable estimates are to be computed. An exception is the work of Wexler et al [16], who impose priors on the distribution of α values, and on the joint distribution of edge magnitudes in the composite image and in the alpha matte. However, their framework is deficient in two ways: first, ad-hoc models were employed for the priors; and second, temporal coherence of the video sequence was not fully exploited, leading to unsatisfactory results when the single-image approach is extended to multiple-image sequences.

The main contribution of this paper is to extend the Bayesian approach to take proper account of spatiotemporal information. This confers many of the advantages of the optic-flow based technique, but with a system that has less dependence on accurate trimaps. We show that a general-purpose nonlinear optimization strategy can efficiently incorporate these priors leading to a robust, highly automated, system for video matte extraction. If manual hints are available, they can easily be included into our framework.

This paper is structured as follows: we derive the Bayesian matting framework, and then discuss how the *maximum a posteriori* (MAP) estimate may be found in practice. Demonstrations of the technique on some example sequences precedes a discussion of the merits and disadvantages of the new technique.

2 Bayesian video matting

The input to the algorithm is a sequence of composite images with RGB pixels $C(x, y, t)$. We assume that we can obtain an estimate of the background [4, 9, 15, 16], which we label $B(x, y, t)$. The unknowns then are the sequence of alpha mattes $\alpha(x, y, t)$ and foreground colours $F(x, y, t)$. Collecting the pixel location into a vector $\mathbf{x} = (x, y, t)$, and colouring unknowns red, we obtain the per-pixel compositing equations

$$\begin{aligned} C_r(\mathbf{x}) &= \alpha_{\mathbf{x}} F_r(\mathbf{x}) + (1 - \alpha_{\mathbf{x}}) B_r(\mathbf{x}) \\ C_g(\mathbf{x}) &= \alpha_{\mathbf{x}} F_g(\mathbf{x}) + (1 - \alpha_{\mathbf{x}}) B_g(\mathbf{x}) \\ C_b(\mathbf{x}) &= \alpha_{\mathbf{x}} F_b(\mathbf{x}) + (1 - \alpha_{\mathbf{x}}) B_b(\mathbf{x}) \end{aligned} \tag{2}$$

The Bayesian formulation of the video matting problem now becomes one of finding the MAP estimate of the foreground image F and the alpha-matte α given C and B:

$$\{F, \alpha\} = \underset{F,\alpha}{\operatorname{argmax}}\, p(F, \alpha | C, B) \tag{3}$$

Using Bayes rule, the posterior can be expressed as a combination of priors on F, α, C and B, and a conditional probability on C and B:

$$p(F, \alpha | C, B) = \frac{p(C, B | F, \alpha) p(F) p(\alpha)}{p(C) p(B)} \tag{4}$$

The MAP estimation is converted into an energy minimization by taking the negative log of the posterior, and noting that $p(C)$ and $p(B)$ do not depend on the unknowns F and α:

$$\{F, \alpha\} = \underset{F,\alpha}{\operatorname{argmin}} \{L(C, B | F, \alpha) + L(F) + L(\alpha)\} \tag{5}$$

where $L(C, B|F, \alpha)$ is the *reconstruction error*, $L(F)$ is the *foreground energy* and $L(\alpha)$ is the negative log *alpha prior*. These terms are described in more detail below.

2.1 Reconstruction likelihood $L(C, B|F, \alpha)$

The first term $L(C, B|F, \alpha)$ in eq. (5) is written

$$L(C, B | F, \alpha) = \sum_{\mathbf{x}} E_{re}\left(F_{\mathbf{x}}, \alpha_{\mathbf{x}}\right) \tag{6}$$

where the per-pixel reconstruction error $E_{re}\left(F_{\mathbf{x}}, \alpha_{\mathbf{x}}\right)$ is

$$E_{re}(\mathbf{x}) = \frac{1}{2\sigma^2} \left\| C_{\mathbf{x}} - \alpha_{\mathbf{x}} F_{\mathbf{x}} - (1 - \alpha_{\mathbf{x}}) B_{\mathbf{x}} \right\|^2 \tag{7}$$

This corresponds to an assumption that the deviations of the given composite image from the exact composition are drawn from a Gaussian distribution of covariance $\sigma^2 \mathbf{I}$ where $\mathbf{I}$ is the 3×3 identity matrix. A value of $\sigma = 5$ graylevels was typical in our experiments.

2.2 Foreground energy $L(F)$

The second term in eq. (5) is the foreground energy $L(F)$. We follow previous authors [5, 10] and use a Gaussian mixture model (GMM) in the RGB colour space to model the distribution of foreground pixels colours for the foreground object. The distribution at each pixel is generated from all foreground pixels in a square neighborhood in the automatically generated trimap (§3.3) as in [10]. The energy over the image F is the sum over all foreground pixels $F_{\mathbf{x}}$, given by

$$L(F) = -\sum_{\mathbf{x}} \log\left(\sum_{k}^{N_k} G(F_{\mathbf{x}}; \Sigma_k, \mu_k)\right) \tag{8}$$

where each per-cluster Gaussian is

$$G(X; \Sigma, \mu) = \frac{1}{\sqrt{(2\pi)^3 |\Sigma|}} \exp(-(X - \mu)^{\top} \Sigma^{-1} (X - \mu)) \tag{9}$$

2.3 Alpha prior $L(\alpha)$

The previous likelihood terms occur in some form, either implicitly or explicitly, in almost all previous video matting work. The transition to a fully Bayesian approach comes when priors are placed on the parameters to be estimated. In this problem, the parameters of interest are the foreground image F, and the alpha matte α. The alpha matte is very tightly constrained both spatially and temporally, and offers an excellent opportunity to gain useful regularization at little computational cost.

However, the alpha matte is unusual in that although large areas are smooth, sharp edges must be maintained between the foreground and background objects. Previous attempts to impose a prior [16] have tended to over-smooth edges or to fail to enforce smoothness in the object interior. By *learning* a prior which models the joint distribution of *edges* in the alpha matte and in the composite image, we obtain a prior which smooths strongly in areas where the composite image is uniform, but can introduce sharp edges where there are edges in the composite image. Note that this imposes a constraint on the sequences which we can successfully process: if there is no edge in the composite image, the alpha image boundary will be blurred. However, this constraint is implicit in all current video matting work, as if there is no edge in the composite image, the foreground and background pixel colour distributions are very close, so alpha is poorly constrained.

In summary, these priors allow us to incorporate two important constraints into the estimation:

- alpha values of zero and one are more likely than mid-range values;
- alpha edges are tightly correlated with edges in the composite image.

By capturing training sequences of objects moving against blue-screen backgrounds, we are able to learn the parameters of the prior distributions.

2.3.1 Alpha distribution

A Beta distribution [16] is used to model the distribution of alpha values as primarily 0's or 1's and has the density and energy functions:

$$
\begin{aligned}
p(\alpha(\mathbf{x})) &= \frac{\alpha(\mathbf{x})^{\eta-1}(1-\alpha(\mathbf{x}))^{\tau-1}}{\beta(\eta,\tau)} \\
E_a(\alpha(\mathbf{x})) &= (1-\eta)\log(\alpha(\mathbf{x})) \\
&+ (1-\tau)\log(1-\alpha(\mathbf{x})) + K_\beta
\end{aligned}
$$

where K_β is a constant and is ignored in the energy function. The values of η and τ are determined by comparing the distribution to ground-truth alpha mattes, and typical values are $\eta = \tau = 0.25$.

2.3.2 Spatiotemporal Consistency Prior

The spatiotemporal consistency prior has the effect of smoothing alpha along, but not across, edges in space and time. Specifically, the prior relates gradients in the alpha matte and gradients in the composite image. The gradient of the composite image is denoted ∇C and is defined by the central-difference approximation

$$
\nabla C(x,y,t) = \frac{1}{2\delta}\begin{pmatrix} \underline{C}(x+\delta,y,t) - \underline{C}(x-\delta,y,t) \\ \underline{C}(x,y+\delta,t) - \underline{C}(x,y-\delta,t) \\ \underline{C}(x,y,t+\delta) - \underline{C}(x,y,t-\delta) \end{pmatrix}
$$

where $\underline{C}$ is the grayscale representation of C. The gradient of the alpha matte, $\nabla\alpha$ is defined analogously. If we further define the discrete directional derivative of α in the direction of $\mathbf{p}$ as

$$
\nabla_{\mathbf{p}}\alpha = \frac{1}{2\|\mathbf{p}\|}\left(\alpha(\mathbf{x}+\mathbf{p}) - \alpha(\mathbf{x}-\mathbf{p})\right) \qquad (10)
$$

then we may represent the prior of Wexler et al [16] by

$$
E_{sc}(\mathbf{x}) = \sum_{\mathbf{p}\in N} (e^{-\|\nabla_{\mathbf{p}}C\|^2} \cdot \|\nabla_{\mathbf{p}}\alpha\|)^2 \qquad (11)
$$

where N is a small collection of pixel offsets, for example $\{(1,0,0),(0,1,0),(0,0,1)\}$. On examining real images, however, it quickly becomes apparent that this prior is a poor approximation to the true situation. Figure 3 compares the approximation to the values learnt from real image sequences (see §3.2), for the case where $\mathbf{p} = (1,0,0)$.

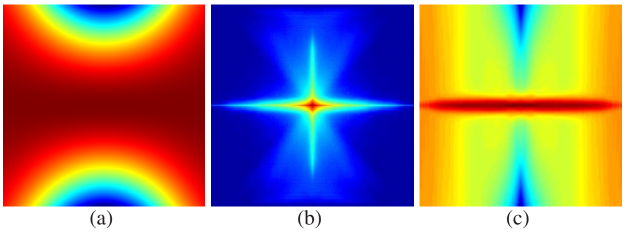
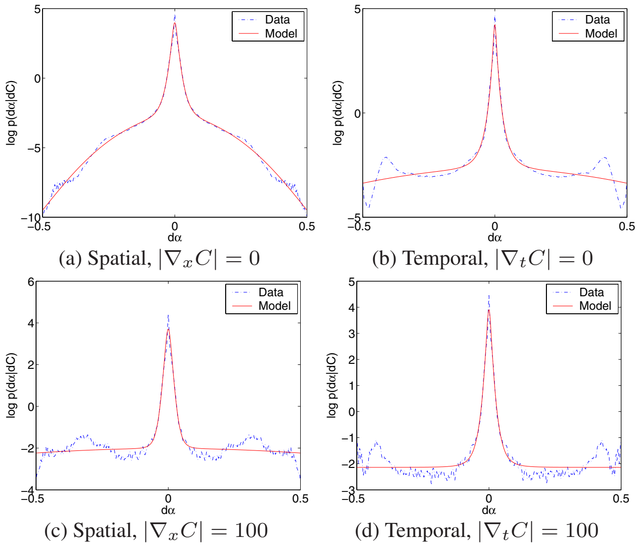

(a) (b) (c)

Figure 3: Spatiotemporal edge log priors. (a) Wexler *et al* spatial consistency energy term E_{sc} as a function of edge strength in alpha and composite images. (b) Learnt marginal prior $\log p(d\alpha, dC)$. (c) Modelled conditional prior $\log p(d\alpha|dC)$. X axes are $\nabla_{\mathbf{p}}C$, Y axes $\nabla_{\mathbf{p}}\alpha$, with $\mathbf{p} = (1,0,0)$. Blue corresponds to low log-probability, red to high. The difference between the learnt and analytic priors is considerable.



(a) Spatial, $|\nabla_x C| = 0$ (b) Temporal, $|\nabla_t C| = 0$

(c) Spatial, $|\nabla_x C| = 100$ (d) Temporal, $|\nabla_t C| = 100$

Figure 4: Sampled and fitted spatiotemporal priors. The dashed lines are the sample distributions of alpha-matte gradients over the training set, for particular values of the composite image gradient. The solid lines show the fit of the data to a mixture of a t-distribution and a Gaussian. As well as allowing analytic derivatives to be computed, the fitted model is guaranteed to be unimodal, leading to improved convergence of the minimization.

2.3.3 Learnt spatiotemporal consistency

In contrast to previous work, we learn the relationship between alpha and composite gradients using a library of ground-truth sequences. Each ground-truth sequence provides a collection of composite-image gradients ∇C and alpha gradients $\nabla\alpha$. The statistic we wish to learn is the joint distribution $p(\nabla C, \nabla\alpha)$, a probability density over $\mathbb{R}^6$.

$$\begin{aligned}
E_{\mathbf{x}} = \quad & ||C_{\mathbf{x}} - \boxed{\alpha_{\mathbf{x}}}\boxed{F_{\mathbf{x}}} - \left(1 - \boxed{\alpha_{\mathbf{x}}}\right) B_{\mathbf{x}}||^2 && \text{— Reconstruction (§2.1)} \\
& - \lambda_1 \log\Big(\sum_{k}^{N_k} \frac{1}{\sqrt{(2\pi)^3|\Sigma_k|}} \exp(-(\boxed{F_{\mathbf{x}}} - \mu_k)^\top \Sigma_k^{-1}(\boxed{F_{\mathbf{x}}} - \mu_k))\Big) && \text{— Foreground mixture model (§2.2)} \\
& - \lambda_2 \left((\eta - 1)\log \boxed{\alpha_{\mathbf{x}}} + (\tau - 1)\log\left(1 - \boxed{\alpha_{\mathbf{x}}}\right)\right) && \text{— Alpha biased to 0 and 1 (§2.3.1)} \\
& - \lambda_3 \sum_{\mathbf{p}\in N} \log\left(\gamma_1(\mathbf{x})(1 + \gamma_2(\mathbf{x})(\boxed{\nabla_{\mathbf{p}}\alpha_{\mathbf{x}}})^2)^{-\gamma_3(\mathbf{x})} + \gamma_4(\mathbf{x})e^{-\gamma_5(\mathbf{x})(\boxed{\nabla_{\mathbf{p}}\alpha_{\mathbf{x}}})^2}\right) && \text{— Learnt edge prior (§2.3.2)}
\end{aligned}$$

Table 1: The complete log likelihood that the algorithm minimizes is $E(\boxed{\alpha}, \boxed{F}) = \sum_{\mathbf{x}} E_{\mathbf{x}}$, with $E_{\mathbf{x}}$ being the term above. The red terms are the variables whose values are desired. Although visually busy, it is smooth, and analytic first and second derivatives are easily computed and translated to source code. Optimizing the function using an off-the-shelf nonlinear minimizer is straightforward, and converges from a wide range of starting positions.

In order to compactly represent the distribution we factor it into the product of marginal distributions over the spatial derivatives ∇_x and ∇_y and the temporal derivative ∇_t.

$$p(\nabla C, \nabla\alpha) = p(\nabla_x C, \nabla_x\alpha)p(\nabla_y C, \nabla_y\alpha)p(\nabla_t C, \nabla_t\alpha)$$

However, since $p(\nabla\alpha|\nabla C) = p(\nabla\alpha, \nabla C)/p(\nabla C)$ and because we assume the prior over C is a function only of the image derivatives, so $p(C) = p(\nabla C)$, then eq. (4) becomes a function of the conditional distribution $p(\nabla\alpha|\nabla C)$ (figure 3(c)).

Each conditional is modelled analytically as a mixture of a t-distribution and a Gaussian. This model is inspired by the statistics of derivatives of natural images [8, 11, 13]. The parameters of this model are determined separately as a function of ∇C by fitting to the sample data. This yields three conditionals each of which is written

$$p(d\alpha|dC) = \gamma_1(1 + \gamma_2 d\alpha^2)^{-\gamma_3} + \gamma_4 \exp(-\gamma_5 d\alpha^2)$$

where $d\alpha$ represents ∇_x, ∇_y or ∇_t of α as appropriate, and dC appears as follows. Because ∇C is constant throughout the estimation of α, the coefficients $\gamma_{1\ldots5}$ are functions of dC, stored in a lookup table. One lookup table is used for the spatial derivatives, and one for the temporal.

Thus, the spatiotemporal priors are represented by the energy term

$$E_{st}(\alpha, C) = -\sum_{\mathbf{x}}\sum_{\mathbf{p}\in N} \log(p(\nabla_{\mathbf{p}}\alpha_{\mathbf{x}}|\nabla_{\mathbf{p}}C_{\mathbf{x}})). \qquad (12)$$

The structure within the modelled prior (figure 3(c)) is worth some explanation. First, there is a maximum at $\nabla\alpha = 0$ for all ∇C. Second, the value of this maximum decreases the further ∇C is from zero. This says that we allow edges in α wherever there are edges in C and that the strength of this belief increases with increasing ∇C.

2.4 Combined likelihood

To summarize this section, the log posterior of a given α, F combination is the sum of the reconstruction likelihood eq. (6), the foreground colour term eq. (8), the log prior on α and the log prior on the joint distribution of edges in the alpha matte and edges in the composite image eq. (12). Table 1 shows the full energy equation for reference. The relative weights of the terms are controlled by constants $\lambda_{1,2,3}$ which are user-visible tuning parameters, whose values default to 1.

3 Implementation

The preceding section has motivated the error functional which is minimized in order to yield estimates of F and α. The overall strategy has been to ensure that the function is sufficiently smooth that a general-purpose nonlinear optimizer can be used to find local minima, and that local minima are widely separated. In this section, we elaborate on some of the implementation considerations involved in minimizing the function.

3.1 Constrained nonlinear minimization

It is important to include in the minimization of the error functional the constraints that α and F are between zero and one. Therefore an optimizer which can handle simple bound constraints on the variables is required. For this work, we used MATLAB's constrained nonlinear optimizer `fmincon`. This is a trust-region variant of the Newton method, and requires second derivatives of the objective function to be supplied. Derivatives of the objective function were computed using Maple, and automatically written to C source code. Using common subexpression elimination [6], the common terms among the various derivatives are identified, so that the C code which computes all derivatives is not much longer than that which computes the objective function itself. Carefully assembling these expressions into a sparse Hessian allows the use of sparse solvers which are efficient even though the number of unknowns is in the hundreds of thousands.

Like graduated non-convexity [3], the minimization is applied three times using smoothed versions of the spatiotemporal prior, with the result from one level of smoothing seeding the algorithm in the next level.

3.2 Learning the alpha prior

An important theme of this work is that priors on the variables are learnt from real image sequences. To this end, we collected several blue-screen sequences for which reliable alpha mattes can be generated using existing techniques. These sequences can be used directly to determine the values of the Beta distribution which characterizes the prior distribution over alpha values, and the results we obtain are similar to those reported in [16]. The sequences are not suitable, however, as data from which to learn the spatiotemporal prior $p(\nabla\alpha|\nabla C)$, because the blue-screen background is not a typical natural image. In order to generate representative training data, we artificially composed the blue-screen foregrounds with natural movie backgrounds, using the ground-truth alpha values, and measured edge statistics from these images. This gives reasonable samples, and figure 4 shows 1D slices from $p(\nabla_{\mathbf{x}}\alpha|\nabla_{\mathbf{x}}C)$ for fixed values of $\nabla_{\mathbf{x}}C$.

3.3 Automatic trimap extraction

Background subtraction can be used to automatically generate a coarse trimap, which confers three benefits. First, it allows the Gaussian mixture model for the foreground to be estimated. Second, it reduces the computational burden for the minimization as only unknown pixels are included in the optimization. Third, it eliminates the "all-foreground" solution $F = C, \alpha = 1$, which would otherwise be the global optimum of eq. (5).

A binary matte is formed by background subtraction, setting $\alpha = 1$ where $|B - C|$ exceeds a preset threshold. Then morphological operations on α are used to generate a spatially coherent trimap. Figure 5 shows typical trimaps extracted from a sequence.

4 Results

Figure 6 shows the output of the algorithm on a test sequence. For these examples, $\sigma = 4$ graylevels and $\lambda_1 = \lambda_3 = 1$ while $\lambda_2 = 4$, representing a desire for more binary alpha values. The recovered alpha matte has correctly recovered the sub-pixel whiskers from the large area which automatic trimap extraction has retained.

Figure 7 shows the output of the algorithm on a test sequence with significant clutter in the background. The actor's hair and facial boundaries are recovered, however the final matte does not have the fine structure that might be expected in the hair regions, because of the low resolution of the original sequence. On compositing with a new background, however, the smooth boundaries do not cause visible artifacts.

Figure 8 shows the effect of temporal consistency on the solution using an image sequence with artificially poor trimaps. Temporal consistency helps to fill some of the holes that spatial consistency alone could not.

5 Discussion

This paper has shown how the incorporation of learnt priors in Bayesian video matting allows fully automatic layer extraction to closely approach the accuracy of manually supervised techniques. It is the first time that priors learned from training sequences have been used in the video matting problem.

By imposing spatiotemporal consistency at edges, we essentially incorporate the propagation of trimaps which is achieved in earlier work by the explicit use of optic flow code. In our work, this confers the advantage that propagation happens only when there is ambiguity in the choice of alpha. On the other hand, the explicit use of optical flow gives the user access to algorithms which have been highly tuned to perform well on a wide range of image sequences. Incorporating flow-based hints is a potentially valuable direction in which to take this work.

Although the various priors have been learnt from training examples, there remains some parameter tuning in the system. For example, the relative weighting of the error terms may need to be adjusted in order to deal with particular image sequences.

The automatic trimap extraction is not new, but has not been useful before. This is because it can produce only coarse trimaps, and previous algorithms tend to require manual touch-up of the trimaps in order to generate clean results. Because our solution has the additional regularization conferred by the priors, it can use a coarser trimap and still obtain good results.

We have shown that formally expressing the problem as an energy minimization, and solving that minimization problem using a general-purpose minimizer yields results as reliable as those obtained by the special-purpose strategies used in previous work. Because a general-purpose optimizer is used, the new approach is significantly easier to

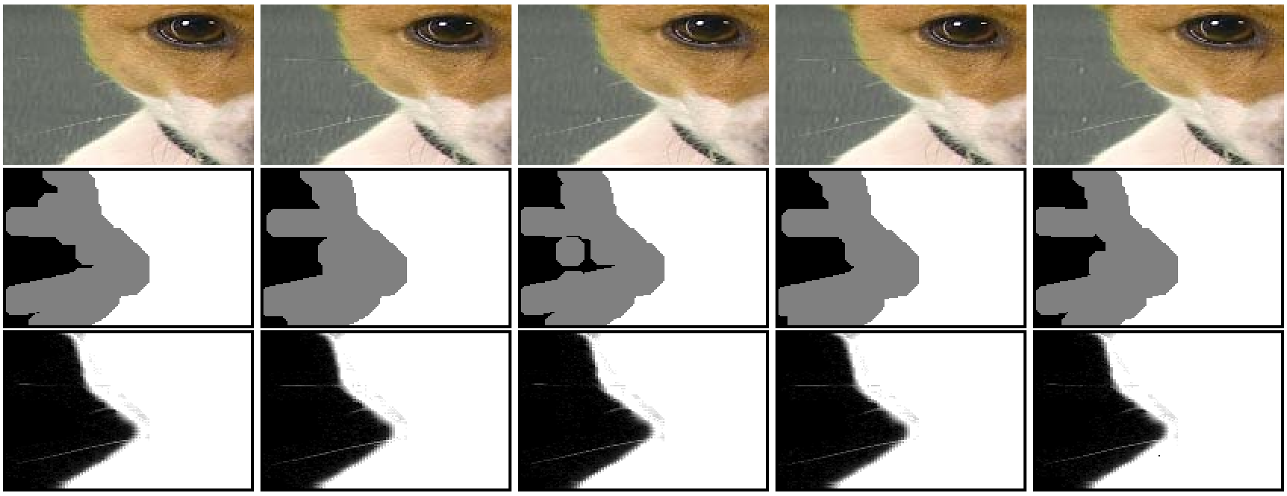
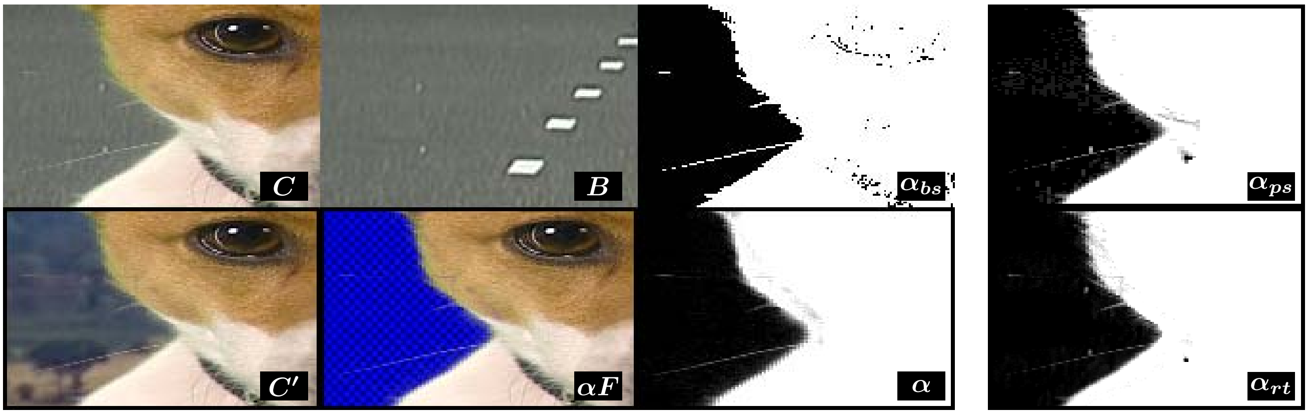


Figure 5: Dog sequence. Top row: input images. Second row: trimaps. Third row: computed alpha mattes. The whiskers are accurately reconstructed in each frame.



(a)

(b)

Figure 6: Results: Dog sequence. (a) Our results. The input composite is C, the recovered background is B. The background subtraction solution is $\alpha_{bs} = |C - B| < 15$ graylevels. The bottom row shows a new composite C', the recovered foreground layer αF, and the recovered α. Note the fine detail in the dog's whiskers which has been automatically recovered. (b) Top row: results using the extract tool from photoshop α_{ps}. Bottom row: the Ruzon and Tomasi solution α_{rt}.

modify, for example in order to incorporate stronger priors for particular problems.

References

[1] S. Ayer and H. S. Sawhney. Layered representation of motion video using robust maximum-likelihood estimation of mixture models and MDL encoding. In *Proc. ICCV*, pages 777–783, 1995.

[2] M. J. Black and P. Anandan. A framework for the robust estimation of optical flow. In *Proc. ICCV*, pages 231–236, 1993.

[3] A. Blake and A. Zisserman. *Visual Reconstruction*. MIT Press, Cambridge, USA, 1987.

[4] Y. Chuang, A. Agarwala, B. Curless, D. H. Salesin, and R. Szeliski. Video matting of complex scenes. *ACM Transactions on Graphics*, 21(3):243–248, July 2002. SIGGRAPH 2002 Proceedings, special issue.

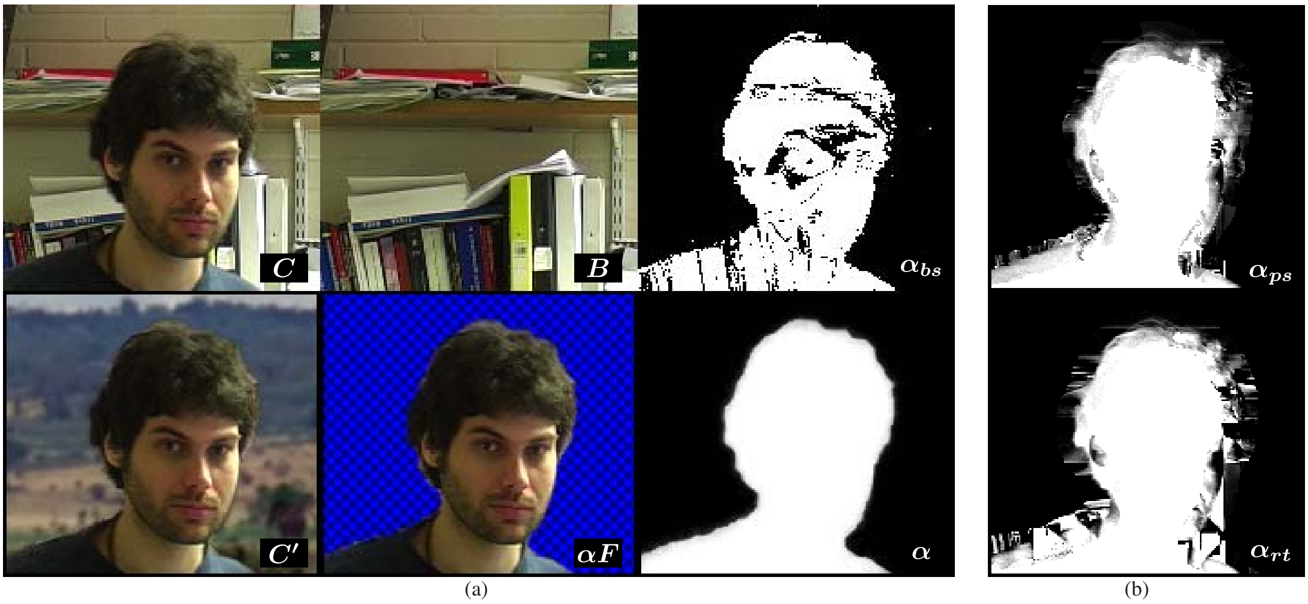
[5] Y.-Y. Chuang, B. Curless, D. Salesin, and R. Szeliski. A Bayesian approach to digital matting. In *Proc. CVPR*, volume 2, pages 264–271, 2001.

[6] J. Cocke. Global common subexpression elimination. *ACM SIGPLAN notices*, 5(7):20–24, 1970.

[7] P. Hillman, J. Hannah, and D. Renshaw. Alpha channel estimation in high resolution images and image sequences. In *Proc. CVPR*, pages 1063–1068, 2001.

[8] J. Huang and D. Mumford. Statistics of natural images and models. In *Proc. CVPR*, pages 1541–1547, 1999.

[9] M. Irani, B. Rousso, and S. Peleg. Computing occluding and transparent motions. *IJCV*, 12(1):5–16, 1994.



(a)

(b)

Figure 7: Results: Josef sequence. (a) Our solution. The input composite is C, the recovered background is B. The background subtraction solution is $\alpha_{bs} = |C - B| < 15$ graylevels. The bottom row shows a new composite C', the recovered foreground layer αF, and the recovered α. (b) Top row: results using the extract tool from photoshop α_{ps}. Bottom row: the Ruzon and Tomasi solution α_{rt}. Note that the solutions in (b) suffer due to poorly separated background and foreground colour models.

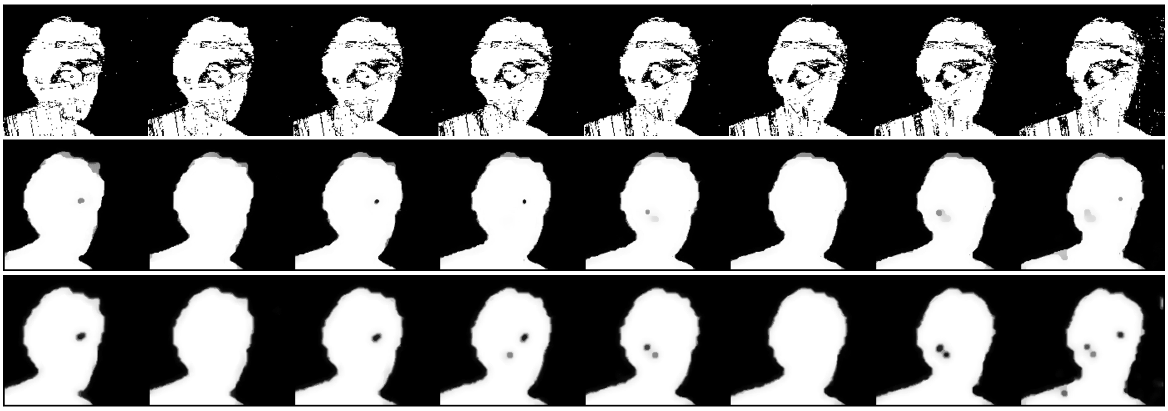


Figure 8: Results: Josef sequence using artificially coarse trimaps to highlight the effect of temporal consistency. Top row: background subtraction solution. Second row: spatiotemporal consistency solution. Third row: spatial consistency solution (no temporal consistency). Note how a number of the holes are filled in with the spatiotemporal consistency but not with spatial consistency.

[10] M. Ruzon and C. Tomasi. Alpha estimation in natural images. In *Proc. CVPR*, volume 1, pages 18–25, June 2000.

[11] H. Scharr, M. J. Black, and H. W. Haussecker. Image statistics and anisotropic diffusion. In *Proc. ICCV*, pages 840–847, 2003.

[12] A. R. Smith and J. F. Blinn. Blue screen matting. In *Proc. ACM SIGGRAPH*, pages 259–268, 1996.

[13] A. Srivastava, A. Lee, E. Simoncelli, and S. Zhu. On advances in statistical modeling of natural images. *Journal of Mathematical Imaging and Vision*, 18(1):17–33, January 2003.

[14] R. Szeliski, S. Avidan, and P. Anandan. Layer extraction from multiple images containing reflections and transparency. In *Proc. CVPR*, volume 1, pages 246–253, 2000.

[15] J. Y. A. Wang and E. H. Adelson. Representing moving images with layers. *The IEEE Transactions on Image Processing Special Issue: Image Sequence Compression*, 3(5):625–638, September 1994.

[16] Y. Wexler, A. W. Fitzgibbon, and A. Zisserman. Bayesian estimation of layers from multiple images. In *Proc. ECCV*, volume 3, pages 487–501. Springer-Verlag, 2002.