PRACTICAL MOTION BASED VIDEO MATTING

Anil Kokaram*
University of Dublin, Trinity College
Dublin, Ireland

Bill Collis[†]
The Foundry
London, UK

Simon Robinson[‡]
The Foundry
London, UK

Abstract

This paper describes a framework for exploiting relatively coarse motion information in the extraction of useable *mattes* from video sequences. While *pulling a matte* from a controlled environment is a well understood problem, increasingly it is desired to segment objects against more difficult, less controlled backgrounds or environments. Further, most usable methods for matting rely on the generation of keyframe garbage mattes by users. The framework presented here employs a Bayesian approach to combine global motion estimates with image based information to automatically create "pretty good" mattes, that are at least as good as *garbage mattes* and can then be refined by subsequent matting algorithms introduced in previous studies. The underlying idea is to separate each frame in the sequence into foreground and background areas by exploiting global motion and *without* the need to generate a clean plate image beforehand. This work is applied to the case of complex, natural background scenes. The principal contribution is an approach for combining various information sources to further reduce the effort required in generating *mattes* in post-production.

Keywords: rotoscoping, matting and compositing, video processing, motion estimation, Bayesian Inference.

1 Introduction

Pulling a matte from a film or video sequence is one of the oldest exercises in film and television post-production. It is used for direct manipulation of the position and nature of objects/actors in scenes in order to create new sequences not originally recorded. In the simplest case, the object is filmed against a green or blue screen. Then, in post-production a combination of detailed manual contour delineation and colour based segmentation (i.e. all that is not blue or green is probably object of interest) is used for creating a mask or *matte*. The mask is non-zero in the region of the object and zero otherwise. It describes the opacity of an object pixel at each location in the image. Thus a mask pixel setting of 1 indicates that that pixel is completely visible as the object of interest, while a mask pixel of 0 indicates that the corresponding object pixel is obscured or not available in some way. This mask or *matte* can then be applied to mix the captured footage containing the object of interest with footage recorded elsewhere. An excellent introduction and background to the Matting problem can be found in [2].

As summarised in [2], traditional methods for pulling video mattes are blue screen matting, rotoscoping and difference matting. Blue screen matting or *chroma keying* relies on capturing the foreground objects against a solid colour background and subsequently pulling the foreground matte by segmentation on the basis of colour. Rotoscoping relies on user drawn editable curves (e.g. Splines) around the foreground object of interest. Snap-to-edge operations as found in commercial packages like Adobe Photoshop, and introduced in previous articles [12] are useful user complements here. Difference matting relies on the generation of a scene containing only background elements (e.g. recording without actors). The image difference between this scene and a scene subsequently recorded with actors is then exploited to generate the matte. The matte here is therefore 1 when the difference is large and 0 otherwise (for instance).

While chroma keying demands a controlled environment for recording, rotoscoping can be achieved regardless of the complexity of the background environment. Employing tracking together with user assisted rotoscoping can greatly improve the utility of contour based approaches. The main limitation of rotoscoping is the inability to correctly express image formation at the boundary between foreground and background. Useable mattes should express the notion that around the boundaries of objects the recorded light is a *mixture* of the background and foreground elements [11]. Difference matting and to some extent chroma keying suffer from the problem that in regions where foreground and background colour are similar, user interaction is required to resolve the matte.

In [3, 2], the authors proposed a combination of limited user interaction followed by direct estimation of a non-binary *alpha matte*. They articulate this information to resolve many of the problems with previous methods. The underlying idea begins with the user specifying what they term a *trimap*. This map divides the scene into regions known to be background (matte pixels set to 0), known to be foreground (matte pixels set to 1), and unknown matte regions. They correctly exploit the knowledge that the difficulty in pulling a very good and useable matte is the proper delineation of the mixing of light effect at the object edge. Hence the restriction of interest to the *unknown* matte region, by exploiting image information from the surrounding known matte regions. Their method is also able to exploit motion information to propagate mattes between user defined and delineated keyframes. They give convincing demonstrations of the matting of translucent material (smoke),

*e-mail: anil.kokaram@tcd.ie
[†]e-mail:bill@thefoundry.co.uk
[‡]e-mail:sam@thefoundry.co.uk

traditionally a very difficult task.

What is notable in all previous work in the area is the implicit acknowledgement that fully automated extraction of mattes for all objects in an arbitrary complex scene is a difficult process. All previous work has adopted the position already in place in the post- production industry. That is, creation of the best image compositing results tend to require a combination of low-level image processing tools and user interaction. This is simply because of the extremely wide variation in scene and lighting complexity that can occur in practice. In the problem of matting however, the generation of keyframe mattes or even *garbage mattes* remains an issue. Even with the automated tools so far presented, users may need to generate keyframe mattes every 10 frames and garbage mattes at *each* frame. In addition, depending on the complexity of the background information, the generation of the *trimaps* required in the work of Chuang et al [3, 2] could require an accuracy that is better than the generation of a *garbage matte*. We will call such a user defined matte, good enough for further refinement as a *pretty good* matte in the remainder of this paper. As a rule of thumb for a PAL sized image, a *pretty good* matte would contain an image edge to an accuracy of ± 10 pixels, while for a *garbage matte* the mask is only guaranteed to contain the main object and its edge accuracy is so poor as to be irrelevant.

The novel contribution of this paper is to push forward the technology for matting by exploiting motion directly in the generation of mattes *without user interaction.* The goal is to reduce the frequency with which the user must interact with the sequence to generate keyframes or garbage mattes before beginning matte extraction. The idea is for the automatically *pretty good* mattes to take the place of garbage matte generation. In doing so, we introduce a framework that combines both clean plate information and motion *without* the need for explicit clean plate generation. Our algorithm for foreground/background segmentation is novel in that it coherently treats spatial and temporal information in a unified framework. Figure 1 shows a typical Garbage Matte and a *Pretty Good* matte. The pretty good matte generally hugs contours more readily and functions as a better start for subsequent matte processing (e.g. via Bayesian Matting). This paper deliberately does not attempt to show results using green screen controlled environments since that situation is well covered by previous work [2]. The focus here is on bringing more automation to the challenging task of pulling mattes from natural scenes.

We begin by introducing in brief an interesting observation that justifies our simplified approach.

2 An observation

In many post-production matting activities we notice that there are generally a limited number of objects that require excessive care in matting. In general, the more objects that are of interest, the less important it becomes to accurately delineate *each* object. In other words, for real scene composition, when two or less objects are of interest, it is likely that the director's goal is to focus the viewer on those objects e.g. a conversation between two actors. Thus the correct manipulation of a few objects, to ensure there are no artefacts, becomes very important. When there are many objects, it is likely that the spectacle of the objects as a unified clump is what is important e.g. a crowd scene or a marching army. Thus all objects in the scene become of interest and it is more important to delineate them as a unified whole contrasted against another background, than to define exactly each object.

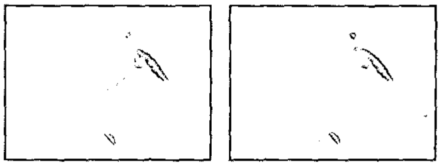

Figure 1: Left: Typical manually scribed Garbage Matte (in red), Right: An automatically generated *Pretty Good* Matte (in red)

This implies that rather than attempt to identify and delineate *each* object in a scene, a generic foreground/background segmentation process could be a reasonable tool for generating initial guesses for mattes in each frame. When there are a limited number of objects, each object is likely to be separable (simply on the basis of position) in the foreground. Thus the user can select objects as needed. While for many objects, the foreground mask could be acceptable as a well defined *blob* that covers all the objects in the scene.

3 Background: Segmentation

Image and video segmentation is a well traversed area [13]. The ultimate goal is to automatically identify semantically connected regions in an image. It is generally agreed that motion in a video sequence is perhaps the most useful feature that could be exploited for delineating an object from the rest of the scene. The first work that exploited motion in this way was presented by Adelson et al [15]. The idea is to use some motion estimation process to estimate motion between two frames. Since all pixels within an object tend to move roughly in the same way, the magnitude and direction of this motion would indicate different objects. However within an object, regions which do not contain textural information, simply do not admit reliable motion information. Furthermore, an articulated, non-rigid moving body (e.g a person) contains many sub- sections (legs, arms) that do not move in exactly the same fashion. To connect these regions into a single segment or mask, automatically it is necessary to incorporate further information e.g. colour and texture. The bulk of modern work in this area has been attempting to incorporate this further information [14].

Rather than attempt to segment all objects and because of the observation in the previous section, the generation of useable *pretty good* mattes could be achieved by identifying just two segments in the scene. One representing the background, and another representing everything that is not background. The advantage of this simplified approach is that articulated objects can be identified as a single semantic object as long as it does not overlap with other objects in the scene. But as is observed, this could be of less importance in the post-production scenario than in a surveillance scenario for instance.

The basic idea of Foreground/Background segmentation is that the motion of the background can be reliably extracted from a scene because of the large area that is experiencing that motion. Subsequent processing is then used to assign pixels to the background motion using some measure of fit. Irani et al [6] appear to present the earliest workable attempt to segment foreground and background regions, although for surveillance rather than compositing. However, that technique does not explicitly acknowledge the spatial coherence of objects, nor does it estimate occlusion and uncovering as the objects move. It is based principally on thresholding the difference between frames compensated for global motion only. In such a circumstance, portions of the image frame that are well explained from past frames by global motion have a low global motion compensated frame difference while the remainder of the image has a large global motion compensated difference. It is likely that the areas of low motion difference are background and those of high difference are foreground. However, in areas of the background that are covered by an object in subsequent or past frames, this difference is also high. Hence although idea is workable, it is not sufficient for use in generating a matte because of the poor edge delineation properties.

3.1 Global Motion Estimation

The estimation of global motion is well established in many different areas: Mosiaking [9], Retrieval: [10], Compression: [4]. The idea is to estimate the motion that most of the image pixels are undergoing due to a global effect like camera motion for instance. A pixel at location $\mathbf{x}$ in frame n ($I_n(\mathbf{x})$) undergoing such motion may have arisen from a location in the previous frame $n-1$ which is shifted from the current location due to some Affine transformation including zoom, rotation and translation. Thus the image sequence model is as follows.

$$I_n(\mathbf{x}) = I_{n-1}(\mathbf{A}\mathbf{x}+\mathbf{d}) \tag{1}$$

where $\mathbf{A}$ denotes a 2×2 matrix encoding the rotation and zoom effect, while $\mathbf{d}$ isa two component vector describing the horizontal and vertical translation component. Most of the techniques (cited above) for estimation of the motion parameters $\mathbf{A}$ and $\mathbf{d}$ use some form of weighted least squares. The idea is to minimize the energy of the global displaced frame difference $E(\mathbf{x})$ defined as

$$E(\mathbf{x}) = \sum_{\mathbf{x}} [I_n(\mathbf{x}) - I_{n-1}(\mathbf{A}\mathbf{x}+\mathbf{d})]^2 \tag{2}$$

over the whole image with respect to the parameters $\mathbf{A}$, d. Since not all the pixels undergo this kind of motion, it is necessary to discard those from consideration and various strategies have been developed to do this. In this paper we employ a strategy described in [7] for estimation of the global motion parameters. In the subsequent sections we assume that the future and next frames are compensated for global motion denoted by I'_{n+1}, I'_{n-1} respectively.

4 Exploiting Global Motion for Matting

To generate a *pretty good* matte, it is required to configure a binary matte l such that the matte pixel $l(\mathbf{x})$ is 1 when the site $\mathbf{x}$ is in the region of definite object and 0 otherwise. Recall that it is not the goal here to establish a non-binary alpha matte at this stage. The binary matte will be refined in subsequent processing to create the actual desired matte.

The basic idea is that pixels which obey the global motion model will agree very well with pixels compensated for that motion in subsequent frames, while others will not. However we must encode as well the notion that pixels at the boundary between object and background occluding surfaces will violate this constraint. Thus we introduce the notion that a background pixel can exist in four states $s = 0,1,2,3$. In state 0 the pixel exists in both the future and previous frames while in states $1,2$ the pixel is covered (occluded) by the object in the future or previous frame respectively. State 3 corresponds to covering in both future and past directions; an outlier state. The situation can therefore be delineated as follows.

$$(l,s) = \begin{cases} 0,0 & \text{Background pixel, exists in past and next frames} \\ 0,1 & \text{Background pixel, covered in next frame} \\ 0,2 & \text{Background pixel, covered in previous frame} \\ 1,3 & \text{Foreground pixel} \end{cases}$$

It may appear that l is a redundant variable but this is not so and this will be addressed later on.

We wish to manipulate $p(l,s|I_n,I'_{n+1},I'_{n-1},L,S)$ to estimate the variables l,s that delineate our *pretty good* matte. L,S denote the site configurations at the eight-connected neighbourhood. Here the arguments for position are dropped as it is assumed that we are dealing with a particular site $\mathbf{x}$. Proceeding in a Bayesian fashion,

$$p(l,s|I_n,I'_{n+1},I'_{n-1},L,S) \propto p(I_n,I'_{n+1},I'_{n-1}|l,s)p(l|L)p(s|S) \tag{3}$$

To proceed we must define the likelihood $p(I_n,I'_{n+1},I'_{n-1}|l,s)$ and priors $p(l|L)$, $p(s|S)$.

4.1 The likelihood

The likelihood should constrain the label field to be 1 when motion compensated differences are large in both the future and past temporal directions. It arises directly from the global motion model given previously as follows.

$$p(\cdot|l,s) \propto \exp - \left(\frac{\Delta_f^2(s \neq 1,3)(1-l) + \Delta_b^2(s \neq 2,3)(1-l))}{\sigma_e^2} \right) + \exp - (\beta(s \neq 0) + \beta(s == 3)\beta l) \quad (4)$$

where Δ_f, Δ_b are global motion compensated pixel differences (using previously calculated motion estimates) defined by

$$\begin{aligned} \Delta_b &= I_n(\mathbf{x}) - I_{n-1}(\mathbf{A}_b\mathbf{x} + \mathbf{d}_b) \\ \Delta_f &= I_n(\mathbf{x}) - I_{n+1}(\mathbf{A}_f\mathbf{x} + \mathbf{d}_f) \end{aligned} \quad (5)$$

where $\mathbf{A}_b, \mathbf{d}_b$ are global motion estimates for the backward image pair while the other estimates are for the forward image pair.

In the likelihood the terms involving β are introduced to suppress the assignment of $l = 1, s = 1,2$ or $s = 3$ everywhere. Without this constraint the posterior can be maximized simply by setting the entire image space to be the object matte. This formulation of the likelihood causes $l = 1$ and $s = 1,2,3$ when the global motion compensated pixel differences are large. Setting $\beta = 2.7^2$ gives a 99% confidence in the decision using the likelihood alone. In practice, σ_e^2 is estimated adaptively from Δ measurements over image blocks of size 64×64 using a robust technique that rejects the maximum 10 % as outliers.

4.2 Priors

An important characteristic of the label field l and discontinuity field s is the spatial coherence. In other words if several neighbours of a site are set to $l = 1$ then one should expect that the current site should also be set to 1. To inject this information we use a well understood Gibbs Energy prior [8] defined as follows.

$$p(l|L) \propto \exp - \left(\Lambda_l \sum_{\mathbf{v}} \lambda(\mathbf{v})(l \neq l(\mathbf{v})) \right) \quad (6)$$

where $\mathbf{v}$ indexes the eight nearest neighbours of l, $l(\mathbf{v})$ are the values of those neighbours and $(l == l(\mathbf{v}))$ is 1 if the condition is satisfied else it is 0. The prior probability is maximized when each of the terms $(l == l(\mathbf{v}))$ is minimized. This implies that the prior encourages all sites in a local region to be configured with the same label. Hence it encourages smoothness. A similar prior is used to encourage smoothness in the discontinuity field s. Replacing l with s in equation 6 gives this prior. Λ and λ defines the strength of this smoothness constraint. $\Lambda = 2.0, 50.0$ is used for the results that follow, and $\lambda(\mathbf{v})$ is a circularly symmetric function that is $1/\sqrt{2}$ for the four diagonal directions and 1 otherwise.

4.3 A practical solution

At each pixel site there are 4 possible solutions as defined in equation 3. The simplest and most direct solution is to evaluate $p(l,s|\cdot)$ for each of these solutions and pick the best. This is called the ICM algorithm [1] and yields a local optimum. We find that this suboptimal method works well in practice. The procedure minimizes the log posterior distribution (i.e. log of equation 3) at each site in turn. This is equivalent to selecting the (l,s) combination that minimizes the following energy E_g.

$$E_g = \left(\Lambda_l \sum_{\mathbf{v}} \lambda(\mathbf{v})(l \neq l(\mathbf{v})) \right) + \left(\Lambda_s \sum_{\mathbf{v}} \lambda(\mathbf{v})(s \neq s(\mathbf{v})) \right) + \left(\frac{\Delta_f^2(s \neq 1,3)(1-l) + \Delta_b^2(s \neq 2,3)(1-l))}{\sigma_e^2} \right) + \exp - (\beta(s \neq 0) + \beta l) \quad (7)$$

Since the Δ terms can be pre-calculated the number of operations at a site is dominated mostly by the spatial energy calculation, requiring 16 operations. The total number of operations at each site is therefore of the order of 20.

Several iterations over the image sites are necessary and a checkerboard scan is useful to prevent error propagation. Iterations are stopped when there is no further change in the estimated variables. This occurs typically after 20 iterations.

4.4 Using clean plate information without generating a clean plate

If the motion of the foreground objects is too small, then there is insufficient energy in Δ_f^2 or Δ_b^2 to differentiate between foreground and background. However, across a longer temporal delay, the motion would become significant. This would boost the classification power of the Δ features. We observe that by averaging Δ_f^2 and Δ_b^2 across several image pairs separated by longer temporal windows, the same framework as discussed above can be used to generate l. Thus a series of measurements of Δ_f^2, made by motion compensating frame pairs $[n,n+1]; [n,n+2]; [n,n+3] \ldots$, are then averaged to yield a Δ_f^2 measurement with greater discrimination power. There is no need to directly calculate these matches, since they can be recursively established as the motion processing progresses. That is the motion between frame pair $[n,n+k]$ can be established by cascading the transformations estimated between pair $[n,n+1]; [n+1,n+2]; [n+2,n+3]; \ldots; [k-1,k]$.

A more careful examination of this procedure shows that in fact this is almost the same kind of information that is being used in the case of background differencing for matte estimation [2]. By concatenating the difference information across several frames, background image information from the future and past is being brought into use for the current frame matte estimation. The crucial difference here is that there is no need to generate a clean plate to exploit this information.

Typically, if information is used across large temporal windows, the occluding and uncovered regions will be greater than that between two frames. This would normally imply some post-processing to extract the true extent of the current image matte. However, the fact that l collects together the cases of $s = 0, 1, 2$ implies that it is still able to delineate the foreground object *pretty* well.

4.5 Multiresolution and spatial coherence

Spatial coherence and algorithm speed is improved by a coarse to fine refinement strategy. Following the work of Heitz and Perez on multiscale MRF's [5] successively coarser grids are defined by grouping together pixels in cells of 2×2, 4×4, 8×8 and so on. Considering each cell as a macro-pixel allows huge savings in computation for each multiresolution level. The algorithm described above is therefore modified firstly to recursively filter the Δ functions with a short separable filter having taps of $[1, 1]$. This generates coarsened versions of Δ measurements that are subjected to identical iterative processes with each finer resolution using a starting point derived from the result at the previous coarser level. The computational overhead in generating the levels is limited because it is the Δ measurements that are filtered *not* the image itself. This improves convergence dramatically and also assists in the filling in of large regions.

To further encourage good spatial behaviour the image itself is used to measure the reliability of Δ measurements. These measurements are reliable only in regions of significant image texture, and the image gradient is used as simple indication of texture. Where image gradient is low, the Λ hyperparameter is increased while it is decreased otherwise. A gamma like expression is used to create this behaviour in Λ as follows.

$$\Lambda = \frac{10}{1 + \exp(g - g_t)} + 1.0 \tag{8}$$

where g is the magnitude of the image gradient at a site, and g_t is 5.0.

4.6 Refinement with still image matting

As discussed previously, the work of [2] illustrated that a Bayesian approach to matting yields very good non-binary alpha-mattes for compositing. In that work, they exploited the concept of the trimap to indicate regions of known foreground $\alpha = 1$, known background $\alpha = 0$ and unknown matte values. The idea is then to estimate the alpha values in the unknown regions. In that work the trimaps had to be generated via user specified keyframes, here these maps are created by exploiting the *pretty good* mattes instead.

Having used global motion for matting above, it is possible to generate a trimap directly by delineating regions in which the segmentation is confident as foreground or background. These regions are created by employing a kind of hysteresis processing of the field l.

Setting Λ_l low (e.g. 2 used here) in equation 6 yields a conservative detection of the foreground region. Setting Λ_l high (e.g. 50 used here) in equation 6 yields an ambitious detection of the foreground region. Hence the inverse of this detection i.e. $l == 0$, yields a conservative detection of the *background* region. These two processes together yield the trimap required for Bayesian Matting. In practice an erosion of the initial fields helps to guarantee conservatism.

The refinement process then continues as described in [2]. In summary, to generate an alpha value for a pixel in the unknown region of the trimap, nearby clusters of known foreground and background pixels are selected. The colour distributions of these clusters are modelled by spatially varying sets of Gaussians. A maximum likelihood criterion is then used to simultaneously estimate the optimum opacity (alpha value), un-multiplied foreground and un-multiplied background colour of the current pixel.

5 Pictures and Discussion

Figures 2, 3, 4 show results from processing natural, complex scenes with the algorithm discussed here. An example from the entire processing chain is shown: the original data, the automatically generated trimap using the Pretty Good Matte concept, and finally the result of alpha-matte extraction using Bayesian matting. The quality of the final matte is shown by *compositing* the forground against a green background using the estimate matte. The corresponding sequences are also available as MPEG4 for reviewing.

The images displayed here are subsampled from the original PAL resolution in order to reduce the size of the .pdf file. The picture material is challenging both because of the camera motion involved and the fact that they are all shot outdoors in natural environments. In the illustration of trimaps, red indicates confident foreground, green confident background and the Bayesian matting process is used to extract the non-binary alpha matte in the intervening region. All results were generated using 3 frames of Δ averaging before and after the current image.

Figure 2 shows two frames from a sequence in which a motorcycle travels across a dune. The idea is to automatically segment the bike from the natural scene. The trimaps generated are very good and the resulting final (pulled) composite image using the non-binary matte generated is believable.

The examples in Figure 3 and 4 show non-rigid motion and the result of interaction with other objects. These sequences show heavy DV artefacting and are therefore difficult to process reliably. in addition, the foreground object in both cases contains similar intensity and colour information as the background. It is therefore difficult to automatically segment these sequences with colour alone. Nevertheless, the results are useful. What is interesting is that although the trimap appears to be well extracted, the high conservatism of the known background matte portion causes some matting confusion. Thus a bit of the background object is attached to the foreground on the right of the face in figure 3. This problem is exacerbated by the fact that the obscured person also moves. Note that the algorithm presented here assumes that all local motion is due to a desired object, hence this behaviour is sensible and expected. Some user interaction is needed to resolve this difficult situation of overlapping but distinct objects.

Figure 4 shows that the pretty good matte is still better than a garbage matte for delineating the woman, even when faced with extreme problems due to articulated motion. The well contrasted regions are well delineated, while those that have ambiguous colour relationship with the background are not. Shadows remain an issue as expected since they travel with the object. This is better seen in the video example corresponding to figure 2.

6 Final Comments

This paper has presented a tool for producing *pretty good* mattes automatically from an image sequence. The material chosen for this work is demanding since it does not constitute a controlled environment i.e. they are *not* the result of green or blue screen scenarios. The tool reduces the amount of keyframing or user interaction needed to kick start previous matting algorithms. There are two novel contributions. The first is the introduction of the idea that rough segmentation could be good enough to automate the garbage matte extraction process. The second has been the introduction of a framework for combining both spatial and temporal information in generating trimaps and final mattes. Necessary extensions to this work must involve incorporation of further image based information like colour and local motion. Resolving issues regarding shadows and DV artefacting hold much potential for future work.

References

[1] J. Besag. On the statistical analysis of dirty pictures. *Journal of the Royal Statistical Society B*, 48:259–302, 1986.

[2] Yung-Yu Chuang, Aseem Agarwala, Brian Curless, David H. Salesin, and Richard Szeliski. Video matting of complex scenes. In *Proceedings of ACM SIGGRAPH*, 2002.

[3] Yung-Yu Chuang, Brian Curless, David H. Salesin, and Richard Szeliski. A bayesian approach to digital matting. In *Proceedings of CVPR*, 2001.

[4] F. Dufaux and J. Konrad. Efficient, robust and fast global motion estimation for video coding. *IEEE Transactions on Image Processing*, 9:497–501, 2000.

[5] F. Heitz, P. Prez, and P. Bouthemy. Multiscale minimization of global energy functions in some visual recovery problems. *CVGIP : Image Understanding*, 59(1):125–134, January 1994.

[6] M. Irani, B. Rousso, and S. Peleg. Computing occluding and transparent motions. *International Journal of Computer Vision*, 12(1):5–16, January 1994.

[7] A. C. Kokaram and Perrine Delacourt. A new global motion estimation algorithm and its application to retrieval in sports events. In *IEEE Workshop on Multimedia Signal Processing*, October 2001.

[8] S. Z. Li. *Markov Random Field Modeling in Computer Vision*. Springer–Verlag, 1995.

[9] H. Nicolas. New methods for dynamic mosaicking. *IEEE Trans. Image Processing*, 10(8):1239–1250, August 2001.

[10] J-M. Odobez and P. Bouthémy. Robust multiresolution estimation of parametric motion models. *Journal of visual communication and image representation*, 6:348–365, 1995.

[11] T. Porter and T. Duff. Compositing digital images. In *Proceedings of ACM SIGGRAPH*, volume 18, pages 253–259, 1984.

[12] P. Prez, A. Blake, and M. Gangnet. Jetstream: Probabilistic contour extraction with particles. In *ICCV 2001, International Conference on Computer Vision*, volume II, pages 524–531, July 2001.

[13] A. Murat Tekalp. *Digital Video Processing*. Prentice Hall, 1995.

[14] Philip H. S. Torr, Richard Szeliski, and P. Anandan. An integrated bayesian approach to layer extraction from image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 23(3):297–303, 2001.

[15] John Y. A. Wang and Edward H. Adelson. Representing moving images with layers. *IEEE Transactions on Image Processing*, 5(3):625–638, September 1994.

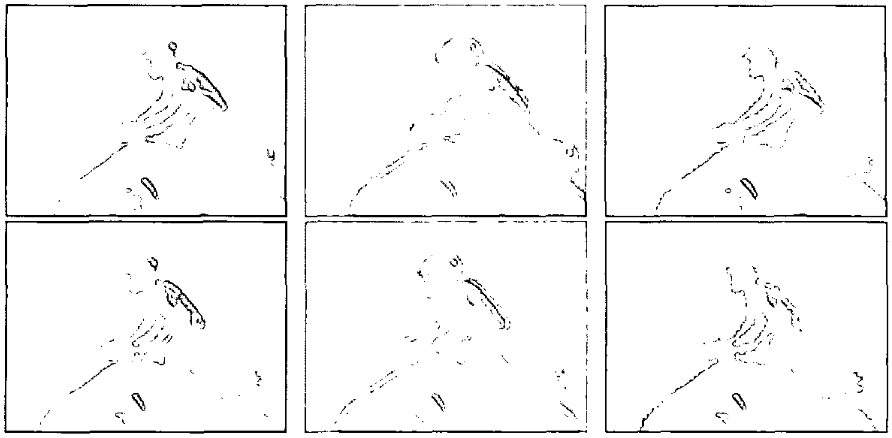

Figure 2: Left to right: Frames from original scene with natural background; The automatically generated *Pretty Good* matte (red=confident background, green = confident foreground), *without the need for a garbage matte*; The final composite frame against a green background using non-binary alpha matting.

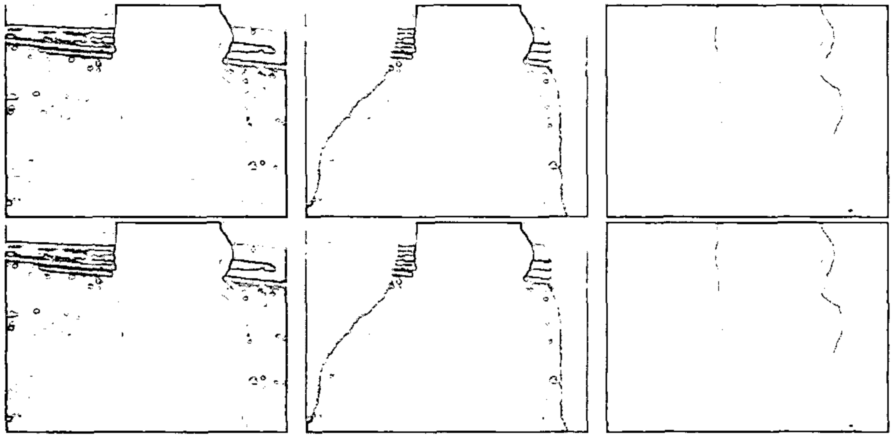

Figure 3: Left to right: Frames from original scene with natural background; The *Pretty Good* matte ; The final composite frame. Note that even though the *pretty good* interior (red) is well defined and corresponds well to the foreground object, the conservative known background estimate causes matting confusion with the partially obscured person on the right hand side of the image.

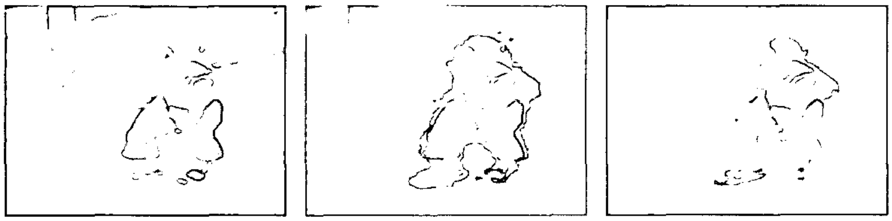

Figure 4: Left to right: Frames from original scene with natural background; The *Pretty Good* matte ; The final composite frame. The woman contains many similar colours to the background therefore the extraction of a pretty good matte is difficult. The usual problems due to shadows and periodic non-movement (where the foot touches the pavement) also yields problems in delineation. Nevertheless the *pretty good* matte is still much more useful than the usual manual garbage matte.