# Natural Image Matting with Total Variation Regularisation

Stephen Tierney and Junbin Gao
School of Computing and Mathematics, Charles Sturt University
Bathurst, NSW 2795, Australia
Email: {stierney, jbgao}@csu.edu.au

***Abstract*—Image mating is the process of isolating the foreground in images and video. This task is challenging as it is severely under constrained. At each pixel we must estimate the foreground and background colour and the blending between them (alpha value). Most approaches calculate an affinity matrix and then minimise a system of linear equations to find the alpha matte. In this work we propose an extension to this class of affinity based matting techniques by introducing a Total Variation constraint over the alpha matte. We show that our Total Variation Regularisation method improves results in the presence of hard boundaries, gaps and holes.**

## I. Introduction

Image matting is the process of extracting the foreground component from an image. Separating a foreground object is called "pulling a matte" or simply "matting". This task is typically performed in image and video editing and computer vision and it has been studied extensively. Most commonly the image capture process is conducted under controlled environments against a constant-coloured background known as "blue or green screen matting". This controlled capture environment enables very fast and accurate algorithms to be used, provided that the foreground component does not contain colours that match the background. In this work we are concerned with the problem of Natural Image Matting, meaning matting with images that are captured in uncontrolled environments.

The complex nature of the problem has encouraged research and the development of a wide array of techniques and solutions. Most techniques rely on the compositing equation [1]

$$I_i = \alpha_i F_i + (1 - \alpha_i) B_i \quad (1)$$

where $I_i$, $\alpha_i$, $F_i$ and $B_i$ are the pixel value, alpha value, foreground colour value and background colour value respectively at pixel $i$. Porter and Duff introduced the alpha channel to control the blending of foreground and background in a compositing scenario. We call the $\alpha$ values over the entire image the "alpha matte". The alpha matte value lies in the range between 0 (true background pixels) and 1 (true foreground pixels).

It can be seen that this is a severely under constrained problem, at each pixel the $\alpha, F$ and $B$ are unknown. Per pixel there are seven unknown variables that need to be solved from three known (RGB) values. To alleviate this problem user input techniques have been developed to provide extra information. By labelling regions of the image as foreground or background the computational load is reduced and the extra information can be used to calculate the unknown regions more easily. The two common input methods are the trimap and scribbles. A trimap is an image of the same dimensions as the original that defines known foreground (white), known background (black) and unknown (grey) regions. Scribbles indicate the definitely foreground and definitely background regions by white and black scribbles over the image. The scribble method is a response to the trimap method in that it attempts to reduce the amount of user input required. A comparison of the trimap and scribble inputs are shown in Figure 1.

In this paper we present an extension to the class of affinity based matting methods by introducing a Total Variation constraint over the alpha matte in an attempt to create an alpha matte that is smoother, more accurately respects boundaries and improves results in the presence of gaps and holes. We perform our experiments with two representative methods: Closed Form Matting proposed by Levin et al. [3] and Image Matting via Local Tangent Space Alignment as proposed by Gao [4].

The paper is organised as follows: Section II is a brief survey of existing matting techniques. Section III formally introduces Total Variation and a minimisation technique. Section IV discusses the affinity based matting which we extend with a Total Variation constraint. Section V details the foreground and background reconstruction process. Section VI provides analysis using a benchmark dataset and in Section VII we make our conclusions.

## II. Related Work

Due to image matting being such an under constrained problem it has generated a lot of interest and potential solutions. The website alphamatting.com [5] has been created for the sole purpose of comparing techniques in a uniform manner with a variety of image types. Most methods fall into two categories: Colour sampling or Affinity matting. Colour sampling techniques usually involve pixel by pixel methods such as Bayesian matting [6], whereas affinity matting techniques define an affinity or similarity matrix for the whole image and use it as the basis for finding an alpha matte, e.g. [3].

Bayesian matting [6] is a colour sampling technique in which alpha estimation is performed pixel by pixel. Two sliding windows march inwards on the unknown region from

978-1-4673-2181-5/12/$31.00 ©2012 IEEE


Fig. 1: The matting process: (a) An image to perform matting on, (b) User supplied trimap where black indicates known background, grey the unknown region and white the known foreground, (c) the Scribble alternative to a trimap with white for known foreground and black for known background, and (d) a resulting alpha matte after matting is performed.

the foreground and background areas and progressively calculate alpha values. Bayesian matting models foreground and background colours as mixtures of Gaussians which can cause it to be prone to failure in highly textured regions of an image where the Gaussian model is insufficient to accurately represent the colour distribution.

In Poisson matting [7], the authors formulated the problem as one of solving Poisson equations with the matte gradient field. Poisson matting relies on the assumption that that colour changes are gradual or smooth in the foreground and background. There are two steps involved: firstly an approximate gradient field of the matte is computed from the input image and trimap, and secondly the matte is obtained from the gradient field by solving Poisson equations.

Grady et al. proposed Random Walk matting [8] in which the alpha value at a pixel is interpreted as the probability that a random walker starting from said pixel will reach a foreground pixel before a background pixel. Contrary to the title of the paper the algorithm does not use a typical random walker technique. Random walk matting merely constructs the graph matrix (Laplacian) and then uses a system of linear equations to find an alpha matte. The paper does introduce a novel manifold learning technique to image matting called Locality Preserving Projection (LPP) [9]. The original RGB values are projected into LPP space upon which the Laplacian graph is constructed. Using LPP is shown to be better at discriminating boundaries than using raw RGB values.

Closed-Form matting [3] is an attempt to eliminate smoothness assumptions and Gaussian modelling. Instead the assumption is that within a small local window (usually $3 \times 3$ pixels) colours are approximately constant, this is referred to as the colour line model. It is shown that under this model that it is possible to analytically remove the foreground and background from the alpha expression. The derived expression is then a quadratic cost function for alpha which is solved as a sparse linear system of equations.

Gao [4] formulated an affinity matting technique called Image Matting via Local Tangent Space Alignment (LTSA). The approach is similar to the Closed-form technique developed by Levin et al.[3]. The difference lies in the formulation of the affinity matrix, once the affinity matrix is created finding a solution for the alpha is identical. Gao uses a manifold learning based on latent variable models and a dimensionality reduction technique called Local Tangent Space Alignment, which was originally proposed by Zhang et al. [10], as the basis for the affinity matrix.

Both works [3] and [4] are representative of machine learning based image matting approaches, proposed in recent years. Readers are referred to some similar works [11], [12], [13], [14].

## III. Denoising with Total Variation Regularisation

Total Variation (TV) is a measurement of how much a function changes over a domain. It has seen primary use as a mathematical basis for denoising signals and images [15]. The principle is that signals or images with noise will have a high total variation and minimising the total variation over an image will bring it to closer to the noiseless original. The class of Total Variation minimisation denoising methods have the advantage that they will remove unwanted detail and preserve the important details such as edges.

The effect of total variation minimisation is best demonstrated graphically. Figure 2 is a recreation of the "bars" experiment as originally performed in [15]. The experiment consists of taking an image of black bars, adding noise and performing total variation minimisation to take the noisy image to an approximation of the original. The surface plots show a 3D representation of the images at the three stages. In the last plot we can see that TV minimisation has preserved the original edges and removed the spurious noise from the image. The only drawback is a slight clipping of the height (intensity) of the bars.

### A. Total Variation

There are two commonly used TV variants for images, see [16]: the isotropic TV defined as (2) and the $l_1$ based and anisotropic TV defined as (3).

Given an (alpha matte) image $\boldsymbol{\alpha} \in \mathbb{R}^{m\times n}$ of dimensions $m \times n$, the isotropic TV is defined as

$$TV_I(\boldsymbol{\alpha}) = \sum_{i=1}^{m-1}\sum_{j=1}^{n-1}\sqrt{(\alpha_{i,j}-\alpha_{i+1,j})^2+(\alpha_{i,j}-\alpha_{i,j+1})^2} + \sum_{i=1}^{m-1}|\alpha_{i,n}-\alpha_{i+1,n}| + \sum_{j=1}^{n-1}|\alpha_{m,j}-\alpha_{m,j+1}| \quad (2)$$

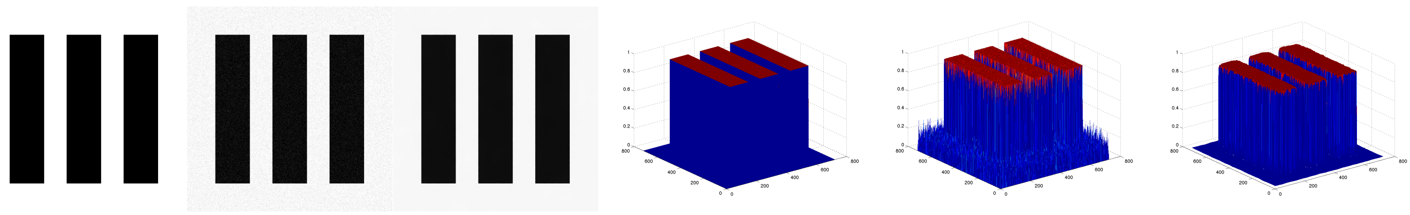

Fig. 2: The bars experiment: (a) The original image, (b) with noise added, (c) after TV minimisation, (d), (e) and (f) the corresponding surface plots

and the $l_1$ based anisotropic version is defined as

$$TV_{l_1}(\boldsymbol{\alpha}) = \sum_{i=1}^{m-1}\sum_{j=1}^{n-1}\{|\alpha_{i,j}-\alpha_{i+1,j}|+|\alpha_{i,j}-\alpha_{i,j+1}|\} + \sum_{i=1}^{m-1}|\alpha_{i,n}-\alpha_{i+1,n}|+\sum_{j=1}^{n-1}|\alpha_{m,j}-\alpha_{m,j+1}| \quad (3)$$

where $\alpha_{i,j}$ is a pixel value at location $[i,j]$.

### B. A Fast Algorithm for TV-Based Denoising

The TV of an image can be considered as prior knowledge which can be introduced into the matting process as a regulariser. For our purpose of image matting, in this section, we provide an overview of a fast algorithm for solving the TV-based image denoising problem, as detailed in [16]. We adopt their techniques in our proposed TV-based matting algorithm, see section IV.

Consider the problem of image denoising with TV regularisation

$$\min \|\mathbf{b}-\boldsymbol{\alpha}\|^2 + 2\lambda_{\text{TV}}TV(\boldsymbol{\alpha}), \quad (4)$$

where $\mathbf{b}$ is a corrupted observation of the "true" image $\boldsymbol{\alpha}$, $TV(\boldsymbol{\alpha})$ is the TV norm defined by (2) or (3) and $\lambda_{\text{TV}}$ is a constant regulariser with an appropriate value controlling the trade-off between degrees of noise and image regularisation.

There are two non-trivial issues with the problem described by (4). Firstly the TV norm is non-smooth. Secondly the large scale of the optimisation problem makes the task of building fast and simple numerical methods difficult, particularly in the case of images. A fast algorithm to (4) is practically desired. Such an algorithm is going to be the building block of our new matting algorithm under the TV constraint, detailed in the next section.

There are many different approaches for dealing with the difficult term $g(\boldsymbol{\alpha}) = 2\lambda_{\text{TV}}TV(\boldsymbol{\alpha})$ in (4) [17], [18], [19]. In recent years, the dual approach becomes affordable via the development of optimisation algorithms such as the Fast Gradient Projection (FGP) algorithm [16], [17].

To demonstrate the resulting dual problem from (4) we must fix some notation:

For the isotropic TV norm, denote $\mathcal{P}_I$ as the set of matrix pairs $(\mathbf{p},\mathbf{q})$ where $\mathbf{p}\in\mathbb{R}^{(m-1)\times n}$ and $\mathbf{q}\in\mathbb{R}^{m\times(n-1)}$ that satisfy

$$p_{i,j}^2+q_{i,j}^2\le 1, \quad |p_{i,n}|\le 1, \quad |q_{m,j}|\le 1 \quad (5)$$

In the case of the anisotropic $l_1$, denote TV norm $\mathcal{P}_{l_1}$ as the set of matrix pairs $(\mathbf{p},\mathbf{q})$ where $\mathbf{p}\in\mathbb{R}^{(m-1)\times n}$ and $\mathbf{q}\in\mathbb{R}^{m\times(n-1)}$ that satisfy

$$|p_{i,j}|\le 1, \quad |q_{i,j}|\le 1 \quad (6)$$

Without loss of generality, we simply use $\mathcal{P}$ for two different cases, actually $\mathcal{P}$ is a constrained subset of $\mathbb{R}^{(m-1)\times n}\otimes\mathbb{R}^{m\times(n-1)}$.

The dual approach is based on the dual relation of the TVs:

$$TV(\boldsymbol{\alpha}) = \max_{(\mathbf{p},\mathbf{q})\in\mathcal{P}} T(\boldsymbol{\alpha},\mathbf{p},\mathbf{q}) \quad (7)$$

where

$$T(\boldsymbol{\alpha},\mathbf{p},\mathbf{q}) = \sum_{i=1}^{m-1}\sum_{j=1}^{n-1} p_{i,j}(\alpha_{i,j}-\alpha_{i+1,j})+q_{ij}(\alpha_{i,j}-\alpha_{i,j+1}) + \sum_{i=1}^{m-1}p_{in}(\alpha_{i,n}-\alpha_{i+1,n})+\sum_{j=1}^{n-1}q_{mj}(\alpha_{m,j}-\alpha_{m,j+1})$$

Taking (7) into (4) and swapping $\min_{\boldsymbol{\alpha}}$ and $\max_{(\mathbf{p},\mathbf{q})\in\mathcal{P}}$ operations, we will find it is easy to work out $\min_{\boldsymbol{\alpha}}$, this results in the following dual problem

$$\max_{\mathbf{p},\mathbf{q}\in\mathcal{P}}\{h(\mathbf{p},\mathbf{q}) = -||\mathbf{b}-\lambda_{\text{TV}}\mathcal{L}(\mathbf{p},\mathbf{q}))||_F^2\} \quad (8)$$

where $\mathcal{L}:\mathbb{R}^{(m-1)\times n}\otimes\mathbb{R}^{m\times(n-1)}\to\mathbb{R}^{m\times n}$ is the discrete divergence linear operator, defined by

$$\mathcal{L}(\mathbf{p},\mathbf{q})_{i,j} = p_{i,j}+q_{i,j}-p_{i-1,j}-q_{i,j-1} \quad (9)$$

where we assume that $p_{0,j}=p_{m,j}=q_{i,0}=q_{i,n}=0$. Once we have a solution $(\mathbf{p}^*,\mathbf{q}^*)$ to (8), the optimal solution to (4) can be given by

$$\boldsymbol{\alpha}^* = \mathbf{b}-\lambda_{\text{TV}}\mathcal{L}(\mathbf{p}^*,\mathbf{q}^*). \quad (10)$$

The dual problem (8) is constrained but its objective function $h(\boldsymbol{p},\boldsymbol{q})$ is smooth. In order to use gradient type methods on the dual problem (8) we must find its gradient which is easily obtained

$$\nabla h(\mathbf{p},\mathbf{q}) = -2\lambda_{\text{TV}}\mathcal{L}^T(\mathbf{b}-\lambda_{\text{TV}}\mathcal{L}(\mathbf{p},\mathbf{q})) \quad (11)$$

where $\mathcal{L}^T$ is the adjoint operator of $\mathcal{L}$.

The method used to solve the dual problem is the Fast Gradient Projection (FGP) algorithm, which is an extension of the Gradient Projection (GP) algorithm. The improvement in FGP over GP stems from the inclusion of information from the previous two iterations. Values from the previous iterations are linearly combined to create an estimation for the next function value so that convergence is achieved much more quickly. For comparison FGP achieves a rate of convergence of $O(1/k^2)$. This means that the algorithm's accuracy is defined as $1/k^2$ where $k$ is the number of iterations. This is a large improvement over the current convergence rate of $O(1/k)$ achieved by GP.

**Algorithm 1 FGP-TV: Fast Gradient Projection algorithm for TV minimisation**

**Require:** $\boldsymbol{b}$ - observed image, $\lambda_{\text{TV}}$ - regularisation parameter and $N$ - number of iterations

1: Take $(\mathbf{r_1}, \mathbf{s_1}) = (\mathbf{p_0}, \mathbf{q_0}) = (\mathbf{0_{(m-1)\times n}}, \mathbf{0_{m\times(n-1)}})$ and $t_1 = 1$

2: **for** $k = 1$ to $N$ **do**

3: $(\mathbf{p_k}, \mathbf{q_k}) = P_{\mathcal{P}}\left[(\mathbf{r_k}, \mathbf{s_k}) + \frac{1}{8\lambda_{\text{TV}}}\mathcal{L}^T(\mathbf{b} - \lambda_{\text{TV}}\mathcal{L}(\mathbf{r_k}, \mathbf{s_k}))\right]$

4: $t_{k+1} = \frac{1+\sqrt{1+4t_k^2}}{2}$

5: $(\mathbf{r_{k+1}}, \mathbf{r_{k+1}}) = (\mathbf{p_k}, \mathbf{q_k}) + (\frac{t_k - 1}{t_{k+1}})(\mathbf{p_k} - \mathbf{p_{k-1}}, \mathbf{q_k} - \mathbf{q_{k-1}})$

6: **end for**

7: **return** $\boldsymbol{\alpha}^* = [\mathbf{b} - \lambda\mathcal{L}(\mathbf{p_N}, \mathbf{q_N})]$

The above algorithm depends on the projection onto the constrained subset $\mathcal{P}$. The conditions of the constraints are given by (5) and (6), as such the projection is easily defined. Given a pair $(\mathbf{p}, \mathbf{q})$, the projection $(\mathbf{r}, \mathbf{s}) = P_{\mathcal{P}_I}(\mathbf{p}, \mathbf{q})$ where $\mathbf{r} \in \mathbb{R}^{(m-1)\times n}$ and $\mathbf{s} \in \mathbb{R}^{m\times(n-1)}$ is given by,

$$r_{ij} = \begin{cases} \frac{p_{ij}}{\max\{1, \sqrt{p_{ij}^2 + q_{ij}^2}\}} & i = 1, \ldots, m-1, j = 1, \ldots, n-1 \\ \frac{p_{in}}{\max\{1, |p_{in}|\}}, & i = 1, \ldots, m-1 \end{cases}$$

$$s_{ij} = \begin{cases} \frac{q_{ij}}{\max\{1, \sqrt{p_{ij}^2 + q_{ij}^2}\}} & i = 1, \ldots, m-1, j = 1, \ldots, n-1 \\ \frac{q_{mj}}{\max\{1, |q_{mj}|\}}, & i = 1, \ldots, m-1 \end{cases}$$

Similarly, the projection $(\mathbf{r}, \mathbf{s}) = P_{\mathcal{P}_{l_1}}(\mathbf{p}, \mathbf{q})$ is given by

$$r_{ij} = \frac{p_{in}}{\max\{1, |p_{in}|\}}, \quad s_{ij} = \frac{q_{mj}}{\max\{1, |q_{mj}|\}}$$

## IV. Matting with Total Variation Regularisation

### A. Formulation

We begin this section by describing the typical approach in affinity based matting for finding an alpha matte then extend it to include the non-smooth Total Variation constraint. With the user-specified constraints like the scribbles or a trimap $s$, one can formulate the matting problem as solving the following smoothed objective function for the best alpha matte $\alpha_*$

$$\min_{\boldsymbol{\alpha}} f(\boldsymbol{\alpha}) = \boldsymbol{\alpha}^T L\boldsymbol{\alpha} + \lambda_{\boldsymbol{\alpha}}(\boldsymbol{\alpha} - b_s)^T D_s(\boldsymbol{\alpha} - b_s) \tag{12}$$

where $L$ is the affinity matrix like Laplacian [3] or LTSA [4], $\lambda$ is a scalar that indicates how confident the user is with their input, $D_s$ is a diagonal matrix whose diagonal elements are one for constrained pixels and zero for all others, and $b_s$ is the vector containing the specified alpha values ($\alpha = 0$ for background and $\alpha = 1$ for foreground) for the contained pixels and zero for others. For more details refer to [3].

In this section we propose to use the TV regularisation to enforce edge preservation in the matting process. In the sequel, without loss of generality, we consider a general unconstrained formulation with a TV regularization

$$\min_{\boldsymbol{\alpha}} \boldsymbol{\alpha}^T L\boldsymbol{\alpha} + \lambda_{\boldsymbol{\alpha}}(\boldsymbol{\alpha} - b_s)^T D_s(\boldsymbol{\alpha} - b_s) + 2\lambda_{\text{TV}} TV(\boldsymbol{\alpha}), \tag{13}$$

### B. Suggested Optimization Algorithm

In (13), denote $f(\boldsymbol{\alpha}) = \boldsymbol{\alpha}^T L\boldsymbol{\alpha} + \lambda_{\boldsymbol{\alpha}}(\boldsymbol{\alpha} - b_s)^T D_s(\boldsymbol{\alpha} - b_s)$, $g() = 2\lambda_{\text{TV}} TV(\boldsymbol{\alpha})$ and $F(\boldsymbol{\alpha}) = f(\boldsymbol{\alpha}) + g(\boldsymbol{\alpha})$, then we can see that $f(\boldsymbol{\alpha})$ is a convex smooth function while $g(\boldsymbol{\alpha})$ is a convex nonsmooth function. Solving (13) is very challenging. In this paper, we propose an efficient TV-Matting algorithm (TV-M) by adopting Nesterov's approach [20] which offers a convergence rate of $O(1/k^2)$ for $k$ iterations. This is optimal for the first-order black-box methods.

To construct the ultimate algorithm, we approximate the objective function $F(\boldsymbol{\alpha})$ in (13) by $f(\boldsymbol{\alpha})$'s second order expansion at a given point $\widetilde{\boldsymbol{\alpha}}$,

$$\begin{aligned} h_{C,\widetilde{\boldsymbol{\alpha}}}(\boldsymbol{\alpha}) &= f(\widetilde{\boldsymbol{\alpha}}) + f'(\widetilde{\boldsymbol{\alpha}})(\boldsymbol{\alpha} - \widetilde{\boldsymbol{\alpha}})^T \\ &+ \frac{C}{2}\|\boldsymbol{\alpha} - \widetilde{\boldsymbol{\alpha}}\|^2 + g(\boldsymbol{\alpha}), \end{aligned} \tag{14}$$

where $C > 0$ is a constant. Thus the problem (13) is converted to minimizing (14).

Nesterov's method for solving (14) is based on two sequences $\{\boldsymbol{\alpha}_k\}$ and $\{\mathbf{s}_k\}$ in which $\{\boldsymbol{\alpha}_k\}$ is the sequence of approximate solutions while $\{\mathbf{s}_k\}$ is the sequence of search points. The search point $\mathbf{s}_k$ is the convex linear combination of $\boldsymbol{\alpha}_{k-1}$ and $\boldsymbol{\alpha}_k$ as

$$\mathbf{s}_k = \boldsymbol{\alpha}_k + \beta_k(\boldsymbol{\alpha}_k - \boldsymbol{\alpha}_{k-1})$$

where $\beta_k$ is a properly chosen coefficient. The approximate solution $\boldsymbol{\alpha}_{k+1}$ is computed as the minimizer of $h_{C_k,\mathbf{s}_k}(\widetilde{\boldsymbol{\alpha}})$. It can be proven that

$$\boldsymbol{\alpha}_{k+1} = \operatorname*{argmin}_{\boldsymbol{\alpha}} \frac{C_k}{2}\left\|\boldsymbol{\alpha} - \left(\mathbf{s}_k - \frac{1}{C_k}f'(\mathbf{s}_k)\right)\right\|^2 + g(\boldsymbol{\alpha}) \tag{15}$$

where $C_k$ is determined by the line search according to the Armijo-Goldstein rule so that $C_k$ should be appropriate for $\mathbf{s}_k$, see [21].

Nesterov's method needs an exact solver for problem (15), however we employ the FPG-TV algorithm introduced in the last section as the building block for an inexact $\boldsymbol{\alpha}_{k+1}$ for problem . Our experiments demonstrate that the final algorithm as described below has great accuracy.

**Algorithm 2** **The Efficient Nesterov's Algorithm for TV-Matting**

**Require:** $C_0 > 0$ and $\boldsymbol{\alpha}_0$, $K$

Initialize $\boldsymbol{\alpha}_1 = \boldsymbol{\alpha}_0$, $\gamma_{-1} = 0, \gamma_0 = 1$ and $C = C_0$.
**for** $k = 1$ to $K$ **do**
  Set $\beta_k = \frac{\gamma_{k-2}-1}{\gamma_{k-1}}$, $\mathbf{s}_k = \boldsymbol{\alpha}_k + \beta_k(\boldsymbol{\alpha}_k - \boldsymbol{\alpha}_{k-1})$
  Find the smallest $C = C_{k-1}, 2C_{k-1}, ...$ such that

$$F(\boldsymbol{\alpha}_{k+1}) \leq h_{C,\mathbf{s}_k}(\boldsymbol{\alpha}_{k+1}),$$

  where $\boldsymbol{\alpha}_{k+1}$ is defined by (15) and solved by FPG-TV algorithm.
  Set $C_k = C$ and $\gamma_{k+1} = \frac{1+\sqrt{1+4\gamma_k^2}}{2}$
**end for**
**return** $\boldsymbol{\alpha}_{k+1}$

## V. Reconstruction of F and B

Once we have the alpha matte $\alpha$ we also wish to reconstruct the foreground and background. A naive approach is to multiply the alpha matte with the input image and multiply the background with the inverse of the alpha matte as follows:

$$F = \alpha I, \quad \text{and} \quad B = (\alpha - 1)I \tag{16}$$

This is the "hard" approach and often does not yield ideal results. Instead we use the approach outlined by Levin et al. [3]. The approach finds $F$ and $B$ by using the original compositing equation (1) and introducing smoothness priors on $F$ and $B$. These smoothness priors are strongest in the presence of matte edges. The objective function for finding $F$ and $B$ is

$$\min_{F,B} \sum_i ||\alpha_i F_i + (1-\alpha_i)B_i - I_i||^2 + \langle \partial\alpha_i, (\partial F_i)^2 + (\partial B_i)^2 \rangle \tag{17}$$

where $\partial$ is the gradient operator. For a fixed $\alpha$ the cost is quadratic and the minimum is found by solving a set of linear equations.

## VI. Results

In this section we provide visual results and quantitative analysis of our technique. It is important to consider both visual and numeric analysis because numeric methods may not reflect the visual quality of an image [5]. For numerical comparison we use the Mean Square Error (MSE) and the Sum of Absolute Differences (SAD), which are the standard measurements in the field of image matting.

We performed experiments using the low resolution training images from the alphamatting.com [5] dataset. This dataset was chosen as it is the benchmark dataset and the ground truths for the alpha mattes are provided. It should be noted that we used the trimaps provided by the dataset and not scribble based input. All the tested techniques were implemented in MATLAB on a consumer notebook with a 2.53Ghz Intel Core 2 Duo processor with 8GB of memory. To reduce computation time we down sampled the images and ground truth alpha mattes via the $downSmpIm$ function provided by Levin et al. in their MATLAB implementation of the Closed Form Solution [3]. Parameters were fixed across the entire dataset with $\lambda_\alpha = 1000$ and $\lambda_{TV} = 0.000005$.

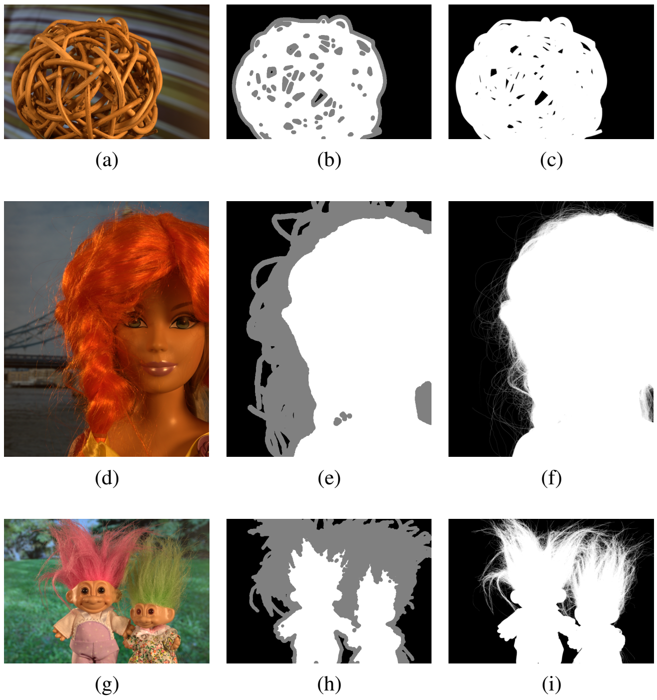


Fig. 3: Three images chosen for detailed analysis. (a) GT02 - Wicker ball, (d) GT03 - Barbie and (g) GT04 - Trolls. The second column is the corresponding trimaps and the third column is the corresponding ground truth alpha mattes

Detailed analysis is provided for three images from the dataset: GT02 (Wicker ball), GT03 (Barbie) and GT04 (Trolls). These images are shown in Figure 3. These images were chosen because they best represent the strengths and weaknesses of our proposed algorithm. Numerical results are provided in Table I and Table II and visual results can be seen in Figures 5, 4 and 6.

| Technique | GT02 | GT03 | GTO4 |
|---|---|---|---|
| Levin | 0.0074 | 0.0022 | 0.0076 |
| Levin + TV | 0.0014 | 0.0071 | 0.0290 |
| LTSA | 0.0074 | 0.0022 | 0.0089 |
| LTSA + TV | 0.0034 | 0.0019 | 0.0405 |

TABLE I: MSE Results. Lower is better.

| Technique | GT02 | GT03 | GTO4 |
|---|---|---|---|
| Levin | 1.1938 | 1.3608 | 2.6633 |
| Levin + TV | 0.5670 | 2.8998 | 8.3170 |
| LTSA | 1.2229 | 1.3764 | 3.2919 |
| LTSA + TV | 0.8327 | 1.4867 | 9.8517 |

TABLE II: SAD Results. Lower is better.

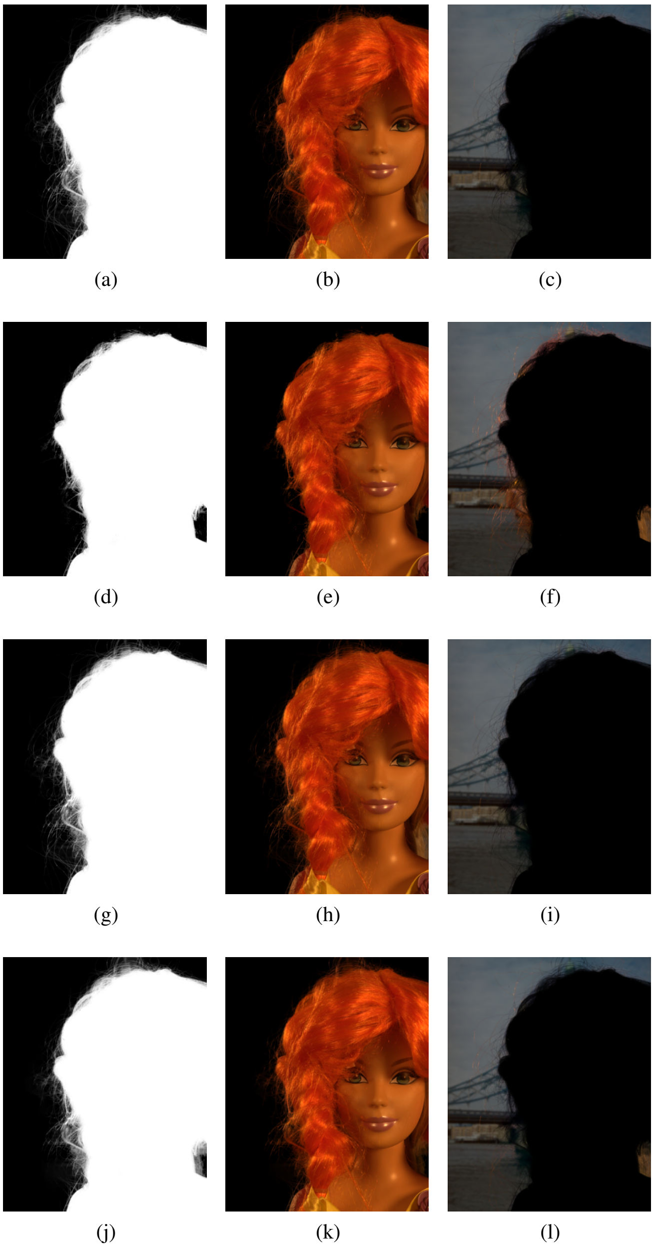


Fig. 4: Barbie - GT03. First row: Levin. Second row: Levin + TV. Third Row: LTSA. Fourth Row: LTSA + TV. First column is the reconstructed alpha, second column is reconstructed foreground and the third column is the reconstructed background.

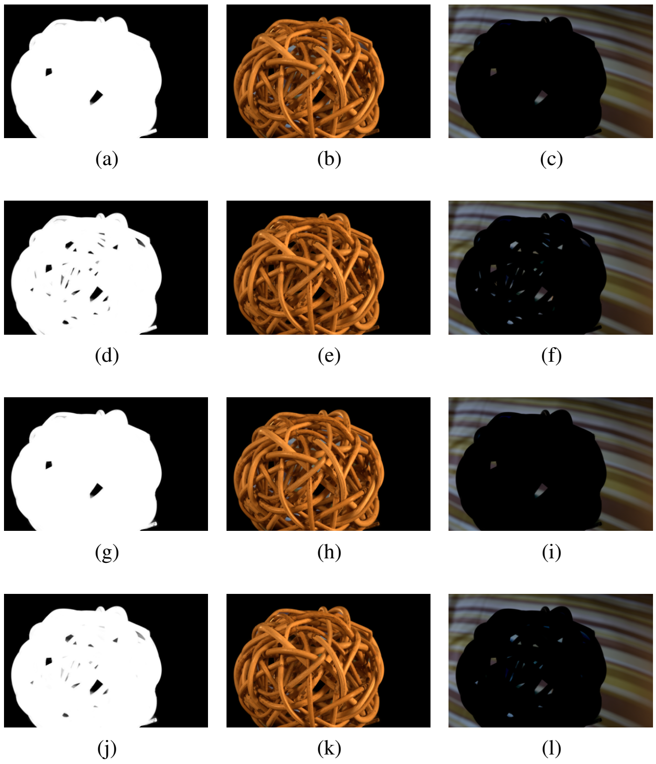


Fig. 5: Wicker Ball - GT02. First row: Levin. Second row: Levin + TV. Third Row: LTSA. Fourth Row: LTSA + TV. First column is the reconstructed alpha, second column is reconstructed foreground and the third column is the reconstructed background.

The Wicker ball (GT02) demonstrates the power of the TV constraint. Visual comparison can be seen in Figure 5. Both Levin + TV and LTSA + TV outperform the non constrained counterparts. The TV constraint more consistently adheres to boundaries and as such will reconstruct the holes in the wicker ball more thoroughly, particularly when combined with the Closed-form approach of Levin et al..

The Barbie image (GT03) represents a difficult case for image matting in that there is a lot of fine detail in the hair. The Levin + TV approach performs the worst (Figure 4 (e)) as large amounts of hair on the left hand side and at the bottom right at neck have been removed. The LTSA + TV approach has the lowest MSE of all as it captures most of the small hairs on the left hand side and does not completely engulf the transparent hairs on the right side of the neck as the non TV approaches do. Minor dulling of the colours can be observed in the TV constrained cases.

The Trolls (GT04 - Figure 6) image illustrates the trade off that must be made between edge preservation and fine detail preservation. The TV constrained approaches accurately reconstruct the gap between the two dolls but fail to reconstruct the entirety of the hair. The non-TV approaches fail to reconstruct the gap but more accurately reconstruct the hair.

Results for the entire alphamatting.com dataset can be seen in Figures 8 and 9. From our experimentation it is demonstrated that applying the TV constraint yields improvements in cases with hard edges but can struggle to produce accurate reconstructions for fine details such as hair and fur. The TV constraint approach excels at dealing with solid objects with holes, much more than the non constrained approaches. It should be noted that including the TV constraint significantly increases computation time. There is a trade off to be made when using these image matting techniques as they are suited to different image types. It is evident that user input is required

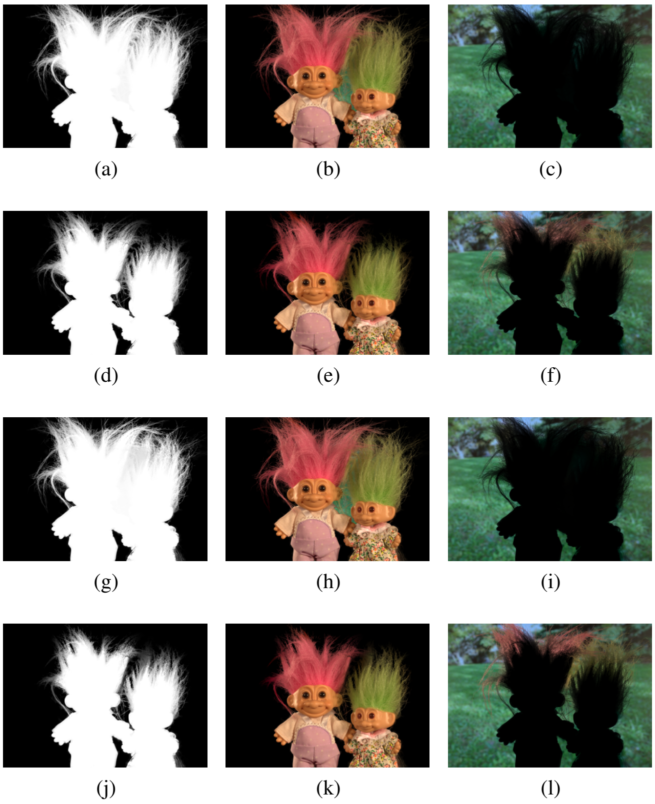


Fig. 6: Trolls - GT04. First row: Levin. Second row: Levin + TV. Third Row: LTSA. Fourth Row: LTSA + TV. First column is the reconstructed alpha, second column is reconstructed foreground and the third column is the reconstructed background.

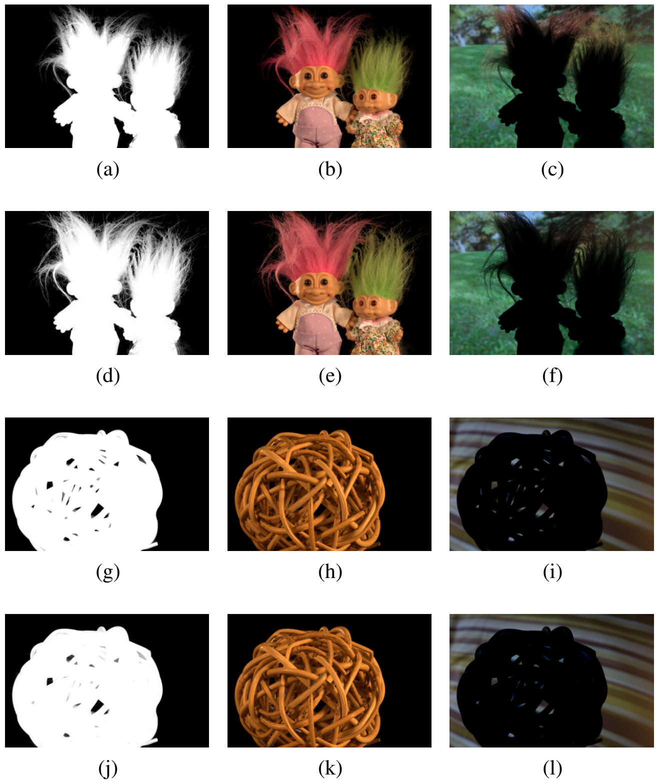


Fig. 7: A comparison of different $\lambda_\alpha$ values for Levin + TV. Rows 1 and 3 $\lambda_\alpha = 1000$ and rows 2 and 4 $\lambda_\alpha = 1$.

to select the more suitable approach on a per image basis. Additionally users may have to fine tune the $\lambda$ parameters to achieve the best possible result as shown in Figure 7. A higher $\lambda_\alpha$ value (e.g. 1000) was found to be good at finding object holes and boundaries when the foreground was more solid, while a lower value causes fine detail to be preserved. On average using $\lambda_\alpha = 1000$ yielded the best results over the dataset.

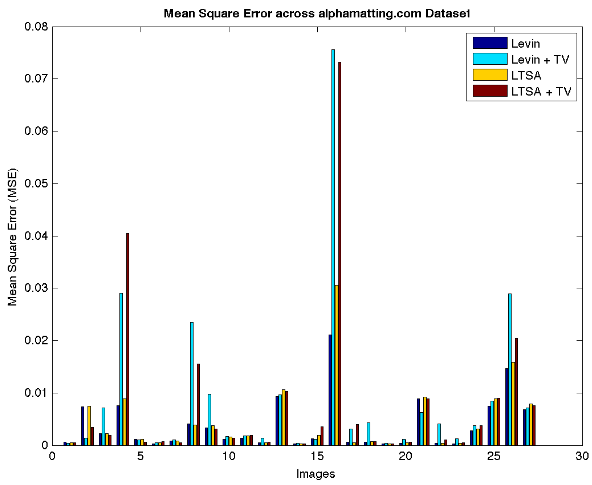


Fig. 8: MSE across the alphamatting.com dataset

## VII. Discussion

We have provided an extension to existing matting techniques by introducing a Total Variation constraint over the alpha matte. This method is applicable to a wide variety of affinity based matting techniques. The TV constraint has shown it's effectiveness in particular cases such as solid objects with holes or gaps. In images that present difficult scenarios for the TV constraint such as hair and fur it still manages to improve the result by providing a much more consistent alpha matte overall. The results reinforce the belief that matting is still an interactive process as different techniques are better suited to different images.

## Acknowledgement

This work is partially supported by the Research Compact Grant from the Faculty of Business at Charles Sturt University and by the Commonwealth of Australia under the Australia-China Science and Research Fund (ACSRF01222).

## References

[1] T. Porter and T. Duff, "Compositing digital images," *Computer Graphics*, vol. 18, no. 3, pp. 253–259, July 1984.

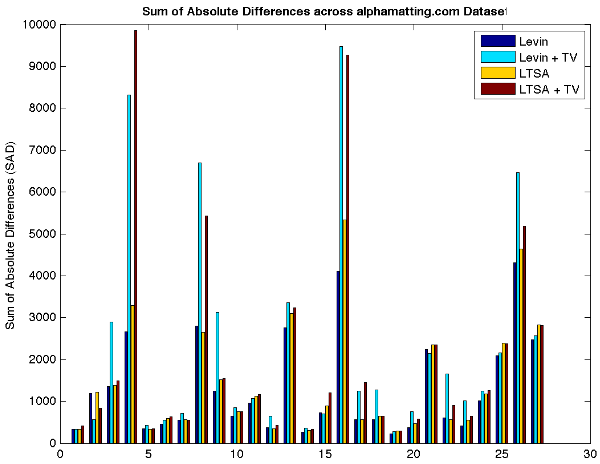


Fig. 9: SAD across the alphamatting.com dataset

[2] J. Wang and M. F. Cohen, "An iterative optimization approach for unified image segmentation and matting," in *Proceedings of the Tenth IEEE International Conference on Computer Vision - Volume 2*, ser. ICCV '05. Washington, DC, USA: IEEE Computer Society, 2005, pp. 936–943. [Online]. Available: http://dx.doi.org/10.1109/ICCV.2005.37

[3] A. Levin, D. Lischinski, and Y. Weiss, "A closed-form solution to natural image matting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 228 – 242, February 2008.

[4] J. Gao, "Image matting via local tangent space alignment," *International Conference on Digital Image Computing: Techniques and Applications*, 2011.

[5] C. Rhemann, C. Rother, J. Wang, M. Gelautz, P. Kohli, and P. Rott, "A perceptually motivated online benchmark for image matting," *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

[6] Y. Chuang, B. Curless, D. Salesin, and R. Szeliski, "A bayesian approach to digital matting," *CVPR*, 2001.

[7] J. Sun, J. Jia, C.-K. Tang, and H.-Y. Shum, "Poisson matting," *ACM Trans. Graph.*, vol. 23, pp. 315–321, Aug. 2004. [Online]. Available: http://doi.acm.org/10.1145/1015706.1015721

[8] L. Grady, T. Schiwietz, and S. Aharon, "Random walks for interactive alpha-matting," *VIIP*, 2005.

[9] X. He and P. Niyogi, "Locality preserving projections," in *Advances in Neural Information Processing Systems*, S. Thrun, L. Saul, and B. Schölkopf, Eds., vol. 16. Cambridge, MA: MIT Press, 2004.

[10] Z. Zhang and H. Zha, "Principal manifolds and nonlinear dimensionality reduction via tangent space alignment," *SIAM J. Sci. Comput.*, vol. 26, no. 1, pp. 313–338, Jan. 2005. [Online]. Available: http://dx.doi.org/10.1137/S1064827502419154

[11] K. He, J. Sun, and X. Tang, "Fast matting using large kernel matting Laplacian matrices," in *Proceedings of CVPR 2010*, 2010, pp. 2165–2172.

[12] K. H. Won, S.-Y. Park, and S. K. Jung, "Natural image matting based on neighbor embedding," in *Lecture Notes in Computer Sciences*, A. Gagalowicz and W. Philips, Eds. Sprin, 2007, vol. 4418, pp. 449–460.

[13] P. Lee and Y. Wu, "Nonlocal matting," in *Proceedings of CVPR 2011*, 2011, pp. 2193 – 2200.

[14] J. Gao, M. Paul, and J. Liu, "The image matting method with regularized matte," in *Proceedings of ICME*, 2012.

[15] L. I. Rudin, S. Osher, and E. Fatemi, "Nonlinear total variation based noise removal algorithms," *Phys. D*, vol. 60, no. 1-4, pp. 259–268, Nov. 1992. [Online]. Available: http://dx.doi.org/10.1016/0167-2789(92)90242-F

[16] A. Beck and M. Teboulle, "Fast gradient-based algorithms for constrained total variation image denoising and deblurring problems," *IEEE Transactions on Image Processing*, vol. 18, no. 11, pp. 2419–2434, June 2009.

[17] A. Chambolle, "An algorithm for total variation minimization and applications," *Journal of Mathematical Imaging and Vision*, vol. 20, pp. 89–97, 2004.

[18] J. Yang, Y. Zhang, and W. Yin, "A fast alternating direction method for TV l1-l2 signal reconstruction from partial fourier data," *IEEE Journal of Selected Topics in Signal Processing*, vol. 4, no. 2, p. 288297, 2010.

[19] Z. Qin, D. Goldfarb, and S. Ma, "An alternating direction method for Total Variation denoising," arXiv:1108.1587v1, preprint, 2011.

[20] Y. Nesterov, "Gradient methods for minimizing composite objective function," Université catholique de Louvain, Belgium, CORE DISCUSSION PAPER 76, 2007. [Online]. Available: http://www.uclouvain.be/cps/ucl/doc/core/documents/Composit.pdf

[21] A. Beck and M. Teboulle, "A fast iterative shrinkage-thresholding algorithm for linear inverse problems," *SIAM Journal of Imaging Sciences*, vol. 2, no. 1, pp. 183–202, 2009.

[22] Z. Zhang and H. Zha, "Principal manifolds and nonlinear dimension reduction via local tangent space alignment," *SIAM Journal of Scientific Computing*, vol. 26, pp. 313–338, 2004.

[23] J. Wang and M. Cohen, "Image and video matting: A survery," *Foundations and Trends in Computer Graphics and Vision*, vol. 3, no. 2, 2007.