# A Convex Image Segmentation: Extending Graph Cuts and Closed-Form Matting

Youngjin Park and Suk I. Yoo

Seoul National University, Kwanak, Seoul, Korea
{yjpark,siyoo}@ailab.snu.ac.kr

**Abstract.** Image matting and segmentation are two closely related topics that concern extracting the foreground and background of an image. While the methods based on global optimization are popular in both fields, the cost functions and the optimization methods have been developed independently due to the different interests of the fields: graph cuts optimize combinatorial functions yielding hard segments, and closed-form matting minimizes quadratic functions yielding soft matte.
In this paper, we note that these seemingly different costs can be represented in very similar convex forms, and suggest a generalized framework based on convex optimization, which reveals a new insight. For the optimization, a primal-dual interior point method is adopted. Under the new perspective, two novel formulations are presented showing how we can improve the state-of-the-art segmentation and matting methods. We believe that this will pave the way for more sophisticated formulations in the future.

## 1 Introduction

Estimating the foreground and background of an image is of great importance in computer vision. When the boundary details are not critical, hard segmentation methods are usually employed that assign a label to every pixel. Graph cuts are among the most successful methods in this class. This might be sufficient for some tasks such as recognition, however, an accurate soft matte (foreground opacity) is often required, for instance, in image editing. Recently, the closed-form formulation using *matting Laplacian* [15] has been proven to be very effective for matting [26]. While generating accurate boundaries, however, it usually requires the unknown region to be small unlike graph cuts. This sort of complementary property has triggered the development of algorithms that take advantage of both sides [18]. Our work reveals the link between the two problems, and the better approaches are obtained through generalization. Note that we will consider the two-layer (foreground and background) segmentation and matting only, but multiple layers can be handled in a standard manner [5, 23].

Segmentation is one of the most intensively studied topics in computer vision. Among many others, graph cuts methods have gained the popularity due to their capability to infer globally consistent labels incorporating various local cues. While the original formulation is in a combinatorial form [10], it is known that its continuous relaxation results in a convex function. In particular, [2] reformulated graph cuts as an unconstrained $l_1$ minimization problem, and solved it using the barrier method, a general algorithm for solving convex problems. Similarly, we reformulated the cost function into the following form (Sec. 2.1):

$$J_1(\alpha) = \|K\alpha\|_1 + \|V(\alpha - \hat{\alpha}_1)\|_1 \tag{1}$$

where $\alpha$ is the relaxed soft segmentation labels. The two terms on the right side corresponds to smoothness and data terms.

The problem of finding an accurate matte has gained attention relatively recently. Currently, many of the state-of-the-art methods [18, 27, 17, 12, 22] are derived from the work of [15]. Generalizing the cost function of these methods yields the following expression (Sec. 2.2):

$$J_2(\alpha) = \|L^{1/2}\alpha\|_2^2 + \|W^{1/2}(\alpha - \hat{\alpha}_2)\|_2^2 \tag{2}$$

The similarities of the two expressions are apparent; they are $l_1$ and $l_2$ norms of the same form.

We recognize this close relationship between the two. Since they are both convex, the sum of them also yields a convex function allowing us to efficiently find a global minimum. Hence, we can see graph cuts and closed-form matting methods within the convex optimization framework. As a result, we propose a new convex segmentation framework whose cost function is given as

$$J(\alpha) = J_1(\alpha) + J_2(\alpha) = \|A\alpha + b\|_1 + (1/2)\,\|C\alpha + d\|_2^2 \tag{3}$$

In this expression, we further merged the smoothness and the data terms, and simply obtained $l_1$ and $l_2$ norms of the linear functions of $\alpha$. While it is possible to consider other types of convex functions, we focus on this combination since each of them is well studied.

This generalization is of more than just the theoretical interests; in fact, by properly choosing the parameters $A$, $b$, $C$, and $d$, we can improve both segmentation and matting quality. We will show two particular examples in Sec. 5. For matting, Sec. 5.1 incorporates $l_1$ data terms, and this makes the method robust to erroneous data estimates. For segmentation, Sec. 5.2 adopts matting Laplacian as smoothness terms, and shows that the shrinking phenomena of graph cuts are mitigated.

One related work is [24]. They have obtained a new segmentation algorithm based on $l_\infty$ norms by generalizing graph cuts and random walker [9]. We further generalize the formulation so that the closed-form matting can be included, and try optimizing joint cost function.

For the optimization, we used a primal-dual interior point method which is a standard technique for convex optimization. Since our problem is often very large including millions of variables and inequality constraints, most general solvers cannot handle it; we implemented new software for minimizing Eq. (3). Exploiting GPU computing technology, we could make it fast enough to be practically used.

Similar formulations and optimization methods often occur in some other fields including sparse signal reconstruction, feature selection, and statistics [13]. For image restoration, [7] also used a primal-dual interior point method minimizing mixed $l_1$ and $l_2$ norms. [1] adopted a similar cost function for the feature learning problem.

In summary, this paper gives a new perspective on segmentation and matting. We showed the relationship between graph cuts and closed-form matting. Following the observation, we developed a new method and obtained promising results. We believe that developing new convex objectives is an attractive future research direction.

## 2 Previous Works

We start by clarifying our notation. The $c$ channel color input image is denoted as $\mathcal{I}$, and $\mathcal{I}_i$ is a $c \times 1$ vector representing colors of the $i$th pixel. We want to estimate the foreground opacity $\mathbf{0} \preceq \alpha \preceq \mathbf{1}$, a column vector of length $n$ where $n$ is the number of pixels. The curled inequality denotes componentwise inequality. $\mathbf{0}$ and $\mathbf{1}$ are column vectors of 0 and 1, respectively, whose size should be apparent from the context. Also, $\|\cdot\|_i$ represents $l_i$ norm. Thus, $\|x\|_1 = \sum_i |x_i|$ and $\|x\|_2^2 = \sum_i x_i^2$ for a vector $x$.

### 2.1 Graph Cuts

Graph cuts methods treat the image as a graph, and divide it by finding a minimum cut of it [10]. The cost function is given as

$$J(\alpha) = \sum_{(i,j)\in N} f_{ij}(\alpha_i, \alpha_j) + \sum_i f_i(\alpha_i) \tag{4}$$

where $N$ is the neighborhood or edge set, and $\alpha_i \in \{0, 1\}$. It consists of two parts: the smoothness terms that are functions of two neighboring pixels, and the data terms encoding local likelihoods. Smoothness terms are designed to prefer similar pixels to be in the same segment, thus, usually defined as a dissimilarity of neighboring pixels.

The minimum cut is usually found by solving its dual problem: max flow problem. It involves the classic Ford-Fulkerson [6, 8] or Push-relabel with their specialized improvements.

While the original formulation restricts the labels to the discrete values, its continuous relaxation allows $\alpha \in [0, 1]$ instead. The cost function should also be adapted to be defined on the continuous domain. In fact, it is known that a convex continuous relaxation is possible, if $f_{ij}$ is submodular; that is, if $f_{ij}(0,0) + f_{ij}(1,1) \leq f_{ij}(0,1) + f_{ij}(1,0)$. One possible form is presented in [2], leading to an unconstrained $l_1$ minimization problem, and it is solved using the barrier method.

Similarly, we reformulate Eq. (4) as the following convex continuous form:

$$J_1(\alpha) = \|K\alpha\|_1 + \|V(\alpha - \hat{\alpha}_1)\|_1 \tag{5}$$

where each row $r$ of $K$ corresponds to an edge $(i,j) \in N$ with its two non-zero elements being defined as

$$K_{r,i} = -K_{r,j} = \left(-f_{ij}(0,0) + f_{ij}(0,1) + f_{ij}(1,0) - f_{ij}(1,1)\right)/2 \tag{6}$$

and the diagonal matrix $V$ and $\hat{\alpha}$ are given as

$$V_{i,i} = |w_i|, \quad \hat{\alpha}_{1i} = \begin{cases} 1 & \text{if } w_i \geq 0, \\ 0 & \text{if } w_i < 0, \end{cases} \tag{7}$$

where $w_i$ is defined as

$$w_i = f_i(1) - f_i(0) - \sum_{j|(i,j)\in N} \frac{f_{ij}(0,0) + f_{ij}(0,1) - f_{ij}(1,0) - f_{ij}(1,1)}{2} \tag{8}$$

It is easy to verify that any minimum of Eq. (4) is also a minimum of Eq. (5).

### 2.2 Closed-Form Matting

Matting problem is to estimate a foreground and a background of an image along with a opacity for each pixel. Most algorithms typically assume compositing equation:

$$\mathcal{I}_i = \alpha_i \mathcal{F}_i + (1 - \alpha_i)\mathcal{B}_i \tag{9}$$

where $\mathcal{F}_i$ and $\mathcal{B}_i$ are $c \times 1$ vectors representing foreground and background colors of the $i$th pixel, respectively. Since the problem is ill-posed, usually a trimap indicating definite foreground $C_F$, background $C_B$, and unknown region is given. We enforce $\alpha_i = 1$ for $i \in C_F$, and $\alpha_i = 0$ for $i \in C_B$.

Currently, matting-Laplacian-based methods produce the best results. They define a quadratic cost function of $\alpha$:

$$J(\alpha) = \alpha^T L \alpha \tag{10}$$

where $L$ is a $n \times n$ symmetric matrix referred as the matting Laplacian that we define:

$$L_{i,j} = \sum_{k|i,j\in w_k} \left(\delta_{ij} - \frac{1}{|w_k|}\left(1 + (\mathcal{I}_i - \mu_k)(\Sigma_k + \frac{\epsilon}{|w_k|} I_c)^{-1}(\mathcal{I}_j - \mu_k)\right)\right). \tag{11}$$

where $\delta_{ij}$ is Kronecker delta, $\Sigma_k$ is a $c \times c$ covariance matrix, $\mu_k$ is a $c \times 1$ mean vector of the colors in a window $w_k$, and $I_c$ is the $c \times c$ identity matrix. Typically, $c = 3$ for color images. See [15] for the original derivation.

There is another way of defining matting Laplacian. While the above is based on color line assumption, [22] assumed locally constant color model for an alternative derivation. In both cases, the cost function is in a quadratic form.

Often, quadratic data terms or priors are added [18, 27, 17, 12].

$$J(\alpha) = \alpha^T L\alpha + (\alpha - \hat{\alpha})W(\alpha - \hat{\alpha}) \quad (12)$$

where $W$ is a diagonal matrix penalizing $\alpha$ from deviating from $\hat{\alpha}$.

We may see Eq. (12) as a combination of closed-form matting and random walker segmentation [9, 27, 17]. Similarities of their cost functions allow easy integration; the same optimization method, solving the following sparse linear system, can be applied.

$$(L + W)\alpha = W\hat{\alpha} - b \quad (13)$$

One noticeable shortcoming of matting Laplacian methods is that the solution often includes mid-range values; in reality, the alpha values of 0 and 1 are more likely. Enforcing sparsity prior in quadratic forms are inherently difficult as we will see in Sec. 5.1.

Rewriting Eq. (12) yields the following expression:

$$J_2(\alpha) = (1/2)\|L^{1/2}\alpha\|_2^2 + (1/2)\|W^{1/2}(\alpha - \hat{\alpha}_2)\|_2^2 \quad (14)$$

## 3 A Convex Segmentation

Many of the best segmentation algorithms can be understood as optimizing a convex function. We refer those as convex segmentation methods. They are particulary interesting since the convexity allows the efficient computation of a global optimum. Among numerous possible convex functions, we suggest a particular form comprising $l_1$ and $l_2$ norms.

Formally, given an input image $\mathcal{I}$ with $n$ pixels, we obtain a foreground opacity $\alpha_i$ for each pixel $i$, by solving the following optimization problem:

$$\begin{array}{ll} \underset{\alpha}{\text{minimize}} & J(\alpha) = J_1(\alpha) + J_2(\alpha) = \|A\alpha + b\|_1 + (1/2)\,\|C\alpha + d\|_2^2 \\ \text{subject to} & \alpha_i = 1,\ i \in C_F, \\ & \alpha_i = 0,\ i \in C_B, \\ & 0 \le \alpha_i \le 1,\ i = 1 \ldots n \end{array} \quad (15)$$

where $C_F$ and $C_B$ are the sets of pixels that are pre-specified as definite foreground and background, respectively. Each component of $\alpha$ is constrained to be in $[0, 1]$.

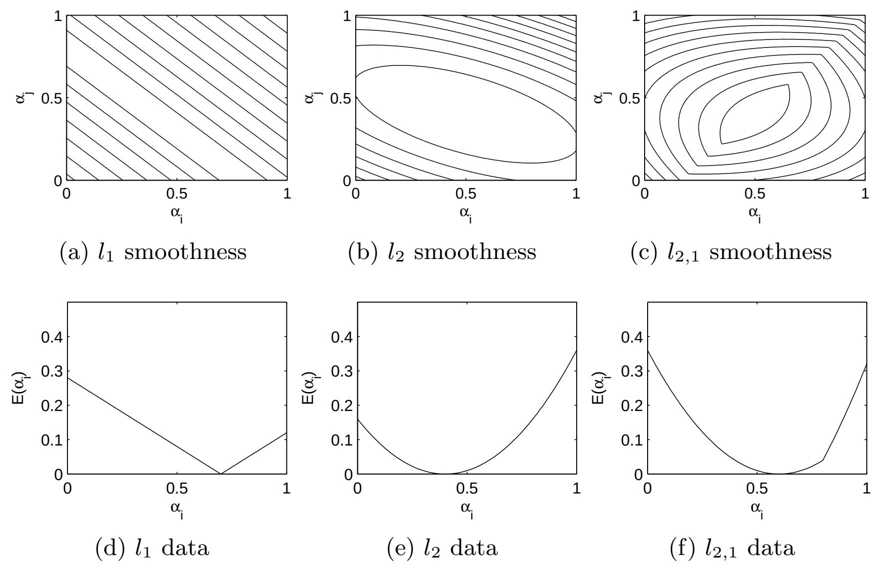
### 3.1 Specializations

The convex segmentation problem of Eq. (15) includes graph cuts and closed-form matting methods. If we let $A = \begin{bmatrix} K \\ V \end{bmatrix}$, $b = \begin{bmatrix} \mathbf{0} \\ -V\hat{\alpha}_1 \end{bmatrix}$, $C = \mathbf{0}^T$, and $d = 0$, then Equ. (15) reduces to graph cuts cost of Eq. (5). If we let $A = \mathbf{0}^T$, $b = 0$, $C = \begin{bmatrix} L^{1/2} \\ W^{1/2} \end{bmatrix}$, and $d = \begin{bmatrix} \mathbf{0} \\ -W^{1/2}\hat{\alpha}_2 \end{bmatrix}$, then Eq. (15) reduces to closed-form matting of Eq. (14). Note that since $L$ and $W$ are positive semidefinite, $L^{1/2}$ and $W^{1/2}$ are valid.

### 3.2 Remarks

In our framework, we have no restriction on $A$, $b$, $C$, and $d$. This means that we may have the expressive power beyond the simple mixture of graph cuts and closed-form matting. For example, the components of $\hat{\alpha}_1$ are constrained to be either 0 or 1 in graph cuts, but any value is possible in the new form (Fig. 1); Sec. 5.1 shows the usefulness of this extension. Also, supermodular cost functions that graph cuts cannot minimize are allowed in this formulation. In fact, Fig. 1 (a) depicts such a case: $f_{ij}(0,0) + f_{ij}(1,1) > f_{ij}(0,1) + f_{ij}(1,0)$.

The $l_1$ and $l_2$ norm minimization problems are well studied in convex optimization. $l_1$ norm minimization is robust to data noise and usually leads to a sparse solution; $l_2$ norm minimization is not robust but has stable solution. These properties also apply to our segmentation problem as we will see in Sec. 5.




(a) $l_1$ smoothness (b) $l_2$ smoothness (c) $l_{2,1}$ smoothness

(d) $l_1$ data (e) $l_2$ data (f) $l_{2,1}$ data

Fig. 1: Possible cost functions: the top row shows the level contours of the cost with respect to two labels, and the bottom row plots the cost as a function of a single label. Our new formulation allows complicated relationships such as (c) and (f) previously impossible.

## 4 Optimization

This section gives a summary on solving the problem of Eq. (15) using a primal-dual interior point method. Although the derivation here is tailored for the $l_1$

and $l_2$ norms, other convex forms might be added harmlessly. We assumed that readers have some knowledge of convex optimization due to the page limit. Refer the text of [3] for deeper understanding.

### 4.1 Reformulation

First, we reformulate the problem so that it can be better handled. First, we remove the equality constraints by substitution. Though we may incorporate them into data terms using large coefficients as in [12], or may just leave it since the primal-dual interior point method can handle them [3], substitution is a better strategy since it reduces the number of variables significantly when dealing with a large trimap. Substituting $\alpha_i = 1$ for $i \in C_F$ and $\alpha_i = 0$ for $i \in C_B$ yields:

$$\begin{aligned} &\underset{\alpha_U}{\text{minimize}} \quad J(\alpha_U) = \left\|A_{(:,U)}\alpha_U + A_{(:,F)}\mathbf{1} + b\right\|_1 + (1/2)\left\|C_{(:,U)}\alpha_U + C_{(:,F)}\mathbf{1} + d\right\|_1 \\ &\text{subject to} \quad \mathbf{0} \preceq \alpha_U \preceq \mathbf{1} \end{aligned} \tag{16}$$

where $\cdot_{(:,U)}$ and $\cdot_{(:,F)}$ give new matrices only with the columns corresponding unconstrained and foreground pixels, respectively, and $\alpha_U$ is $\alpha$ only with unconstrained pixels. Since $A_{(:,F)}\mathbf{1} + b$ and $C_{(:,F)}\mathbf{1} + d$ are again constant column vectors, this substitution does not change the form of the equation; it just removes some columns and rows. Hence, without loss of generality, we will assume the case with no equality constraints from Eq. (15):

$$\begin{aligned} &\underset{\alpha}{\text{minimize}} \quad J(\alpha) = \|A\alpha + b\|_1 + (1/2)\|C\alpha + d\|_2^2 \\ &\text{subject to} \quad 0 \le \alpha_i \le 1, \; i = 1 \ldots n \end{aligned} \tag{17}$$

Second, we introduce an auxiliary variable $w$ to deal with the nondifferentiability of $l_1$ norms. It is a standard technique, and is also used in [2].

$$\begin{aligned} &\underset{\alpha,w}{\text{minimize}} \quad J(\alpha, w) = \mathbf{1}^T w + (1/2)\|C\alpha + d\|_2^2 \\ &\text{subject to} \quad -w \preceq A\alpha + b \preceq w \\ &\qquad\qquad\quad \mathbf{0} \preceq \alpha \preceq \mathbf{1} \end{aligned} \tag{18}$$

Eq. (18) is equivalent to Eq. (17), but has a differentiable objective with additional inequality constraints. Rewriting once more the inequality constraints gives the final smooth form that we will handle:

$$\begin{aligned} &\underset{\alpha,w}{\text{minimize}} \quad J(\alpha, w) = \mathbf{1}^T w + (1/2)\|C\alpha + d\|_2^2 \\ &\text{subject to} \quad Z\begin{bmatrix}\alpha \\ w\end{bmatrix} + Y \preceq \mathbf{0}, \text{ where } Z = \begin{bmatrix} -A & -I \\ A & -I \\ -I & \mathbf{0} \\ I & \mathbf{0} \end{bmatrix} \; \textit{and } Y = \begin{bmatrix} -b \\ b \\ \mathbf{0} \\ -\mathbf{1} \end{bmatrix} \end{aligned} \tag{19}$$

### 4.2 A Primal-Dual Interior Point Method

We start from the Karush-Kuhn-Tucher (KKT) optimality conditions for Eq. (19):

$$\begin{aligned} Z \begin{bmatrix} \alpha^* \\ w^* \end{bmatrix} + Y &\preceq \mathbf{0}, \\ \lambda^* &\succeq \mathbf{0}, \\ \mathrm{diag}(\lambda^*) \left( Z \begin{bmatrix} \alpha* \\ w^* \end{bmatrix} + Y \right) &= \mathbf{0}, \\ \begin{bmatrix} C^T C \alpha^* + C^T d \\ \mathbf{1} \end{bmatrix} + Z^T \lambda^* &= \mathbf{0} \end{aligned} \quad (20)$$

where the column vector $\lambda$ is a dual variable of length $m$ associated with $m$ inequality constraints. These conditions must be satisfied by any pair of primal and dual optimal points $\alpha^*$, $w^*$, and $\lambda^*$.

We cannot find such a point analytically, so we resort to a sequential numerical algorithm. We update the current point $(\alpha, w, \lambda)$ following the primal-dual search direction $(\Delta\alpha, \Delta w, \Delta\lambda)$. Until the convergence, the next point $(\alpha^+, w^+, \lambda^+)$ is obtained:

$$\alpha^+ = \alpha + \tau\Delta\alpha, \quad w^+ = w + \tau\Delta w, \quad \lambda^+ = \lambda + \tau\Delta\lambda \quad (21)$$

where $\tau$ is called a search step.

Search directions are obtained by Newton's method applied to a series of modified KKT equations expressed as $r_t(\alpha, w, \lambda) = \mathbf{0}$, where we define: (cf. Eq. (20))

$$r_t(\alpha, w, \lambda) = \begin{bmatrix} \begin{bmatrix} C^T C\alpha + C^T d \\ \mathbf{1} \end{bmatrix} + Z^T \lambda \\ -\mathrm{diag}(\lambda) \left( Z \begin{bmatrix} \alpha \\ w \end{bmatrix} + Y \right) - (1/t)\mathbf{1} \end{bmatrix} \quad (22)$$

The search step $\tau$ is decided so that the updated values still satisfy the two inequality constraints of Eq. (20):

$$\tau = \sup\{\tau \in [0,1] \mid Z \begin{bmatrix} \alpha + \tau\Delta\alpha \\ w + \tau\Delta w \end{bmatrix} + Y \preceq \mathbf{0}, \lambda + \tau\Delta\lambda \succeq \mathbf{0}\} \quad (23)$$

After one update, $t$ is recalculated using the surrogate duality gap [3]:

$$t = \mu m / (-(Z \begin{bmatrix} \alpha \\ w \end{bmatrix} + Y)^T \lambda) \quad (24)$$

where $\mu$ is a parameter that works well on the order of 10.

Note that as $t$ increases, the modified KKT equation better approximates the equality conditions of Eq. (20), while the inequality conditions are satisfied by Eq. (23). Hence, the solution converges to a global minimum satisfying all the KKT conditions.

### 4.3 A Newton Step

The Newton step solves the nonlinear equation $r_t(\alpha, w, \lambda) = \mathbf{0}$ for a fixed $t$ by forming Taylor approximation at the current point $x = (\alpha, w, \lambda)$ yielding a search direction $\Delta x = (\Delta\alpha, \Delta w, \Delta\lambda)$:

$$r_t(x + \Delta x) \approx r_t(x) + Dr_t(x)\Delta x = \mathbf{0} \tag{25}$$

In terms of $\alpha$, $w$, and $\lambda$,

$$\begin{bmatrix} H & Z^T \\ -\mathrm{diag}(\lambda)Z & \mathrm{diag}(-(Z\begin{bmatrix}\alpha \\ w\end{bmatrix} + Y)) \end{bmatrix} \begin{bmatrix}\Delta\alpha \\ \Delta w \\ \Delta\lambda\end{bmatrix} = -r_t(\alpha, w, \lambda) \tag{26}$$

where $H = \begin{bmatrix} C^TC & 0 \\ 0 & 0\end{bmatrix}$. A block elimination yields:

$$\left[H + Z^T\mathrm{diag}(\lambda)\mathrm{diag}(s)^{-1}Z\right]\begin{bmatrix}\Delta\alpha \\ \Delta w\end{bmatrix} = -\begin{bmatrix} C^TC\alpha \\ \mathbf{1}\end{bmatrix} - (1/t)Z^T\mathrm{diag}(s)^{-1}\mathbf{1} \tag{27}$$

where $s = -(Z\begin{bmatrix}\alpha \\ w\end{bmatrix} + Y)$.

We solve the sparse linear system of Eq. (27) using the conjugate gradient method. Our GPU implementation is influenced by [4]; we also used Jacobian preconditioner. This leaves rooms for further significant speedup using sophisticated preconditioners.

Once the primal search direction $(\Delta\alpha, \Delta w)$ is obtained, the dual search direction $\Delta\lambda$ is given as

$$\Delta\lambda = \mathrm{diag}(s)^{-1}\mathrm{diag}(\lambda)Z\begin{bmatrix}\Delta\alpha \\ \Delta w\end{bmatrix} - \lambda + \mathrm{diag}(s)^{-1}(1/t)\mathbf{1} \tag{28}$$

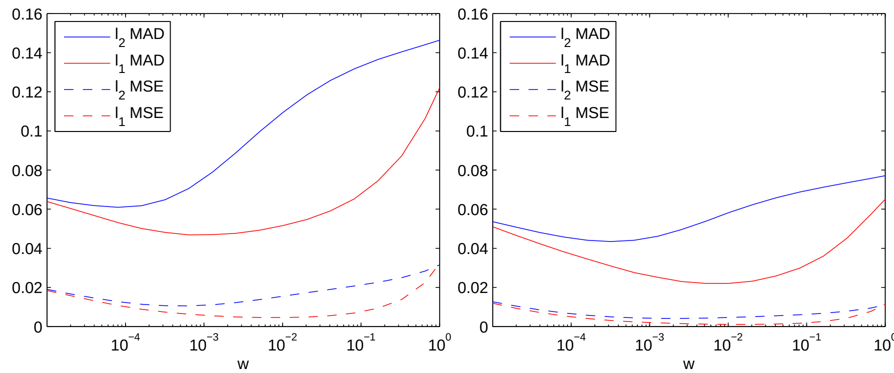
## 5 Applications

Having defined the new convex form and knowing that we can optimize it, this section shows the examples that actually benefit from that. Note that the cost functions of this section need to be transformed to fit in Eq. (15) before being optimized. This should be easy following Sec. 2. For every experiment in this section, the computation time for each image was less than 10 seconds.

### 5.1 Matting

Many of the current state-of-the-art matting methods incorporate $l_2$ data terms based on certain global color models resulting in the following cost function (see Sec. 2.2):

$$J(\alpha) = \alpha^T L\alpha + (\alpha - \hat{\alpha})W(\alpha - \hat{\alpha}) \tag{29}$$




(a) $\epsilon = 0.2, \sigma = 0.2$ (b) $\epsilon = 0.1, \sigma = 0.1$

Fig. 2: Robust data terms for matting: using $l_1$ data terms almost always give lower errors under the assumption of the estimation model Eq. (31): varying $\epsilon$ and $\sigma$ does not change the general shape of the plot.

where $\hat{\alpha}_i$ is the most likely opacity value for the pixel $i$. However, often the estimation of $\hat{\alpha}_i$ is erroneous due to imperfect color models, and the final result is easily affected by small amount of outliers. In such cases, it is known that using $l_1$ norms, instead, gives robust results. Hence, we designed a new formulation:

$$J(\alpha) = \alpha^T L \alpha + \|W(\alpha - \hat{\alpha})\|_1 \tag{30}$$

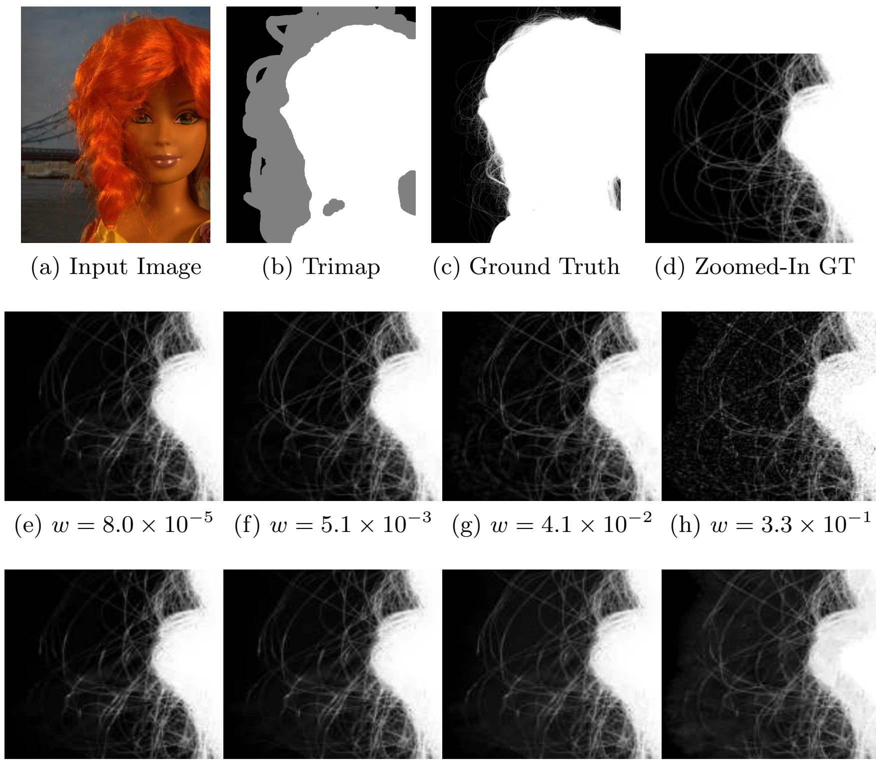
Our expectation is that minimizing Eq. (30) gives more accurate matte than minimizing Eq. (29), and we have experimentally confirmed this. For the fair comparison of the two, we have assumed the following hypothetical estimation model rather than resorting to a particular method:

$$\hat{\alpha}_i \sim \begin{cases} U(0,1) & \text{with probability } \epsilon, \\ \gamma_i + N(0, \sigma^2) & \text{with probability } (1-\epsilon) \end{cases} \tag{31}$$

where $U(0,1)$ is a uniform distribution, $\gamma_i$ is the ground truth opacity, and $N(0, \sigma^2)$ is a Gaussian distribution with variance $\sigma^2$; we simulate the measurement of the term $\hat{\alpha}_i$. Hence, the measurement is incorrect with the probability $\epsilon$. Also, $W$ is assumed to be a diagonal matrix whose diagonal elements are all $w$.

This experiment requires the ground truth matte, so we used the training dataset of [19]. We have measured the error with respect to $w$, which is the weight of the data term. Fig. 2 shows that using $l_1$ norm significantly reduces the errors. This is the result averaged over all 27 images.

Since the two error measures, mean absolute difference (MAD) and mean squared error (MSE), are sometimes inconsistent with the perceptual quality [19], we also examined the results qualitatively, but we found no inconsistency in this case. Fig. 3 shows the close-up look at one of the results. As expected, we could find the problem of non-sparse solution is relieved when the new $l_1$ data terms are used.



(a) Input Image (b) Trimap (c) Ground Truth (d) Zoomed-In GT

(e) $w = 8.0 \times 10^{-5}$ (f) $w = 5.1 \times 10^{-3}$ (g) $w = 4.1 \times 10^{-2}$ (h) $w = 3.3 \times 10^{-1}$

(i) $w = 4.0 \times 10^{-5}$ (j) $w = 1.6 \times 10^{-4}$ (k) $w = 6.4 \times 10^{-4}$ (l) $w = 4.1 \times 10^{-2}$

Fig. 3: Qualitative comparison between different data terms for matting. It shows the zoomed-in results for a particular region. The second row: Eq. (30) with $l_1$ data terms. The third row: Eq. (29) with $l_2$ data terms. (f) and (j) are the best cases. We can see that the hairs stand out clear in the $l_1$ case.

### 5.2 Segmentation

We may also improve the segmentation quality by replacing the smoothness term of graph cuts with the well-defined matting Laplacian term. Our new formulation has the following form:

$$J(\alpha) = \lambda \alpha^T L \alpha + \sum_i f_i(\alpha_i) \tag{32}$$

Of course, since $f_i$ is only defined when $\alpha_i \in \{0,1\}$, the continuous relaxation (in Sec. 2.1) is required. In this way, we can avoid the heuristic step often involved in defining the smoothness terms. Also, a well known shrinking bias of graph cuts can be mitigated. However, since this results in a soft segmentation, the final thresholding step is required; we used the fixed threshold of 0.5.

We first implemented GrabCut method [21] and tried substituting the matting Laplacian term for the smoothness term. The weighting constant $\lambda$ is set to 10 upon the normalization of data terms: $\exp(-f_i(0)) + \exp(-f_i(1)) = 1$. For the calculation of $L$, $5 \times 5$ windows are used. Then, we tested two methods on the dataset of [20].

As expected, we could see the performance gain especially in the presence of fuzzy or blurry boundaries. However, the overall error stayed almost same, because the new formulation is sensitive to non-Gaussian noise; the image compression noise often affects the result largely because GrabCut framework relies on iterative estimation. The qualitative comparison confirmed that the new method is very promising. Fig. 4 presents some of the results on the segmentation dataset of [20, 16].

## 6 Conclusion

We presented a novel segmentation framework based on convex optimization by extending graph cuts and closed form matting methods: we obtained a unified viewpoint to see segmentation and matting. While many of the previous works have focused on how to refine the cost function within the fixed forms, e.g. trying various data terms and smoothness terms, the new formulation suggests that we may consider altering the form itself. By doing so, we can overcome the inherent limitation imposed by a certain form.

Encoding new types of prior would be good future research, since some of the recent works showed that incorporating proper prior often boosts the performance: e.g. bounding box prior [14], geodesic star convexity [11], and connectivity prior [25]. It is interesting that many of them have convex objectives.

Hoping it to be useful for solving large scale problems, we release the reference implementation[1] of the primal-dual interior point method that exploits GPU computing technology.

**Acknowledgement**. The ICT at Seoul National University provides research facilities for this study.

## References

1. Argyriou, A., Evgeniou, T., Pontil, M.: Multi-task feature learning. Advances in Neural Information Processing Systems 19 (2007)
2. Bhusnurmath, A., Taylor, C. J.: Graph cuts via $l_1$ norm minimization. IEEE Trans. PAMI **30** (2008) 1866-1871

[1] http://ailab.snu.ac.kr/~yjpark

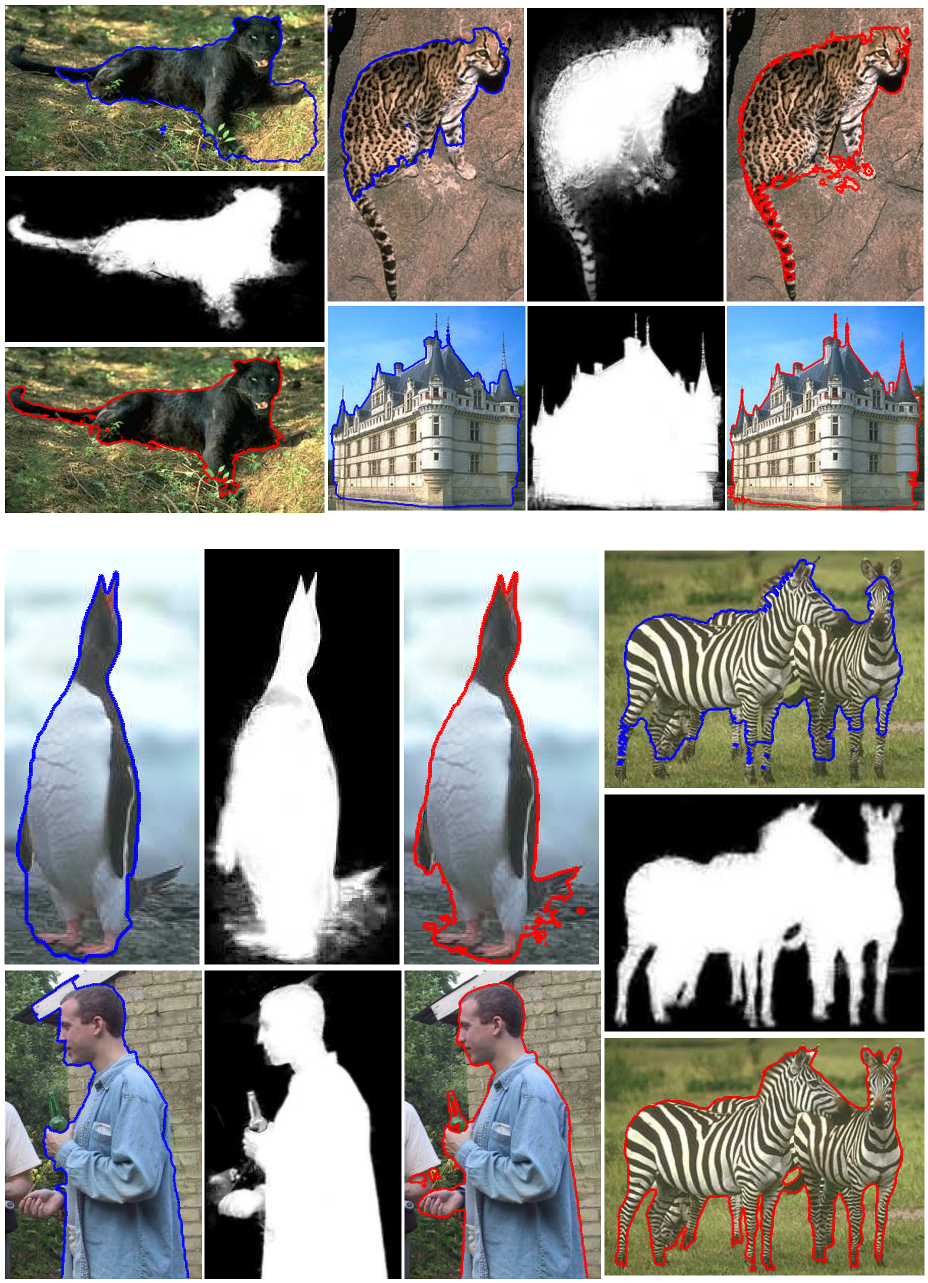


Fig. 4: Segmentation results for GrabCut[21] (*blue contour*) and our new formulation (*red contour*). The grayscale image shows the raw results (the minimum of Eq. (32)) before thresholding. The new formulation shows accurate results for thin parts and blurry boundaries.

3. Boyd, S., Vandenberghe, L.: Convex Optimization. Cambridge University Press, New York (2004)
4. Buatois, L., Caumon, G., Levy, B.: Concurrent number cruncher: an efficient sparse linear solver on the GPU. In: High Performance Computation Conference (2007)
5. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. IEEE Trans. PAMI **23** (1999) 2001
6. Ford, L. R., Fulkerson, D. R.: Maximal flow through a network. Canad. J. Math. **8** (1956) 399-404
7. Fu, H., Ng, M. K., Nikolova, M., Barlow, J. L.: Efficient minimization methods of mixed l2-l1 and l1-l1 norms for image restoration. SIAM J. Sci. Comput. **27** (2006)
8. Goldberg, A. V., Tarjan, R. E.: A new approach to the maximum flow problem. Eighteenth Annual ACM Symposium on Theory of Computing (1986) 136–146
9. Grady, L.: Random walks for image segmentation. IEEE Trans. PAMI **28**(11) (2006) 1768-1783
10. Greig, D. M., Porteous, B. T., Seheult, A. H.: Exact maximum a posteriori estimation for binary images. Journal of the Royal Statistical Society (1989)
11. Gulshan, V., Rother, C., Criminisi, A., Blake, A., Zisserman, A. Geodesic star convexity for interactive image segmentation. In: CVPR (2010)
12. He, K., Sun, J., Tang, X.: Fast matting using large kernel matting laplacian matrices. In: CVPR (2010)
13. Kim, S., Koh, K., Lustig, M., Boyd, S., Gorinevsky, D.: An interior-point method for large-scale l1-regularized least squares. IEEE Journal of Selected Topics in Signal Processing **1** (2007) 606–617
14. Lempitsky, V., Kohli, P., Rother, C., Sharp, T.: Image segmentation with a bounding box prior. In: CVPR (2009)
15. Levin, A., Lischinski, D., Weiss, Y.: A closed-form solution to natural image matting. IEEE Trans. PAMI (2008)
16. Martin, D., Fowlkes, C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: ICCV (2001)
17. Rhemann, C., Rother, C., Gelautz, M.: Improving color modeling for alpha matting. In: BMVC (2008)
18. Rhemann, C., Rother, C., Rav-Acha, A., Sharp, T.: High resolution matting via interactive trimap segmentation. In: CVPR (2008) 1–8
19. Rhemann, C., Rother, C., Wang, J., Gelautz, M., Kohli, P., Rott, P.: A perceptually motivated online benchmark for image matting. In: CVPR (2009) 1826–1833
20. Rother, C.: Grabcut dataset. http://tinyurl.com/grabcut
21. Rother, C., Kolmogorov, V. Blake, A.: "GrabCut": interactive foreground extraction using iterated graph cuts. ACM Trans. Graphics **23** (2004) 309–314
22. Singaraju, D., Rother, C., Rhemann, C.: New appearance models for natural image matting. In: CVPR (2009) 659–666
23. Singaraju, D., Vidal, R.: Interactive image matting for multiple layers. In: CVPR (2008)
24. Sinop, A. K., Grady, L.: A seeded image segmentation framework unifying graph cuts and random walker which yields a new algorithm. In: ICCV (2007) 1–8
25. Vicente, S., Kolmogorov, V., Rother, C.: Graph cut based image segmentation with connectivity priors. In: ICCV (2008)
26. Wang, J., Cohen, M. F.: Image and video matting: a survey. Foundations and Trends in Computer Graphics and Vision **3** (2007) 97–175
27. Wang, J., Cohen, M. F.: Optimized color sampling for robust matting. In: CVPR (2007) 1–8