Matting Update

CVFX @ NTHU

5 May 2015

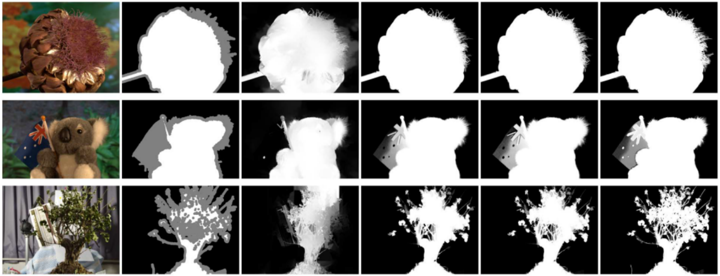
Outline

Section

KNN matting

"KNN Matting", Chen, Li, and Tang. TPAMI, vol. 35, no. 9, September 2013.

- nonlocal principle
- k nearest neighbors in the feature space for matching
- works well with click inputs (more convenient than scribbles)



Input Trimap Spectral Matting Closed-Form Learning Based KNN

KNN matting

- Code available

```
http://dingzeyu.li/projects/knn/
```

The compositing equation

$$I = \sum_{i=1}^{n} \alpha_i F_i, \; \sum_{i=1}^{n} \alpha_i = 1\,.$$

Nonlocal principle for alpha matting

- Assumption: a denoised pixel i is a weighted sum of the pixels with similar appearace to the weights given by a kernel functon $K(i,j)$

$$E[X(i)] \approx \sum_j X(j)K(i,j)\frac{1}{D_i}$$

$$K(i,j) = \exp\left(-\frac{1}{h_1^2}\|X(i)-X(j)\|_g^2 - \frac{1}{h_2^2}d_{ij}^2\right),$$

$$D_i = \sum_j K(i,j).$$

$X(i)$ is a feature vector computed using the information at pixel i, and d_{ij} is the pixel distance between pixels i and j.

Expected value of alpha matte

$$E[\alpha_i] = \sum_j \alpha_j K(i,j) \frac{1}{D_i}$$

or

$$D_i \alpha_i \approx K(i,\cdot)^T \boldsymbol{\alpha}$$

$\boldsymbol{\alpha}$ is the vector of all α values over the input image.

1. the nonlocal principle applies to $\boldsymbol{\alpha}$
2. the conditional distribution $\boldsymbol{\alpha}$ given X is $E[\alpha_i | X(i) = X(j)] = \alpha_j$ the pixels having the same appearance are expected to share the same alpha value
3. replacing the local color-line assumption

Clustering Laplacian

$$D\boldsymbol{\alpha} \approx A\boldsymbol{\alpha}$$

$A = [K(i,j)]$ is an $N \times N$ affinity matrix
$D = \text{diag}(D_i)$ is an $N \times N$ diagonal matrix
N is the total number of pixels

$$(D - A)\boldsymbol{\alpha} \approx \mathbf{0}$$

$$\boldsymbol{\alpha}^T L_c \boldsymbol{\alpha} \approx 0$$

$$L_c = (D - A)^T (D - A)$$

The corresponding quadratic minimization problem

$$\min_{\boldsymbol{\alpha}} \sum A_{ij}(\alpha_i - \alpha_j)^2$$

Computing A using KNN

Collecting nonlocal neighborhoods j of a pixel i before their feature vectors $X(\cdot)$ are matched using $K(i,j)$.

- Existing efficient package for computing KNN: FLANN

Feature vector X with spatial coordinates

$$X(i) = (\cos(h), \sin(h), s, v, x, y)_i$$

h, s, v are the HSV coordinates and x, y are the spatial coordinates of pixel i

Kernel function

$$K(i,j) = 1 - \frac{\|X(i) - X(j)\|}{C}$$

favors soft segmentation (cf.e^{-x})

Closed-form solution

The Laplacian $L = D - A$ is sparser than the clustering Laplacian $L_c = (D - A)^T(D - A)$.

With user input, extracting $n \geq 2$ layers

$$(L + \lambda D)\sum_{i}^{n} \boldsymbol{\alpha}_i = \lambda \mathbf{m}$$

$\mathbf{m}$ is a binary vector of indices of all the marked-up pixels and $D = \text{diag}(\mathbf{m})$.
Closed-form solution:

$$g(x) = x^T L x + \lambda \sum_{i \in \mathbf{m} - \mathbf{v}} x_i^2 + \lambda \sum_{i \in \mathbf{v}} (x_i - 1)^2$$

$\mathbf{v}$ is a binary vector of pixel indices corresponding to user markups for a given layer

Derivation of closed form solution

$$
\begin{aligned}
g(x) &= x^T L x + \lambda \sum_{i \in \mathbf{m}-\mathbf{v}} x_i^2 + \lambda \sum_{i \in \mathbf{v}} x_i^2 - 2\lambda \mathbf{v}^T x + \lambda |\mathbf{v}| \\
&= x^T L x + \lambda \sum_{i \in \mathbf{m}} x_i^2 - 2\lambda \mathbf{v}^T x + \lambda |\mathbf{v}| \\
&= \frac{1}{2} x^T 2(L + \lambda D) x - 2\lambda \mathbf{v}^T x + \lambda |\mathbf{v}| \\
&= \frac{1}{2} x^T H x - c^T x + \lambda |\mathbf{v}| \, .
\end{aligned}
$$

Let

$$\frac{\partial g}{\partial x} = Hx - c = 0.$$

The optimal solution:

$$x = H^{-1} c = (L + \lambda D)^{-1} (\lambda \mathbf{v}) \, .$$

Video matting

"Motion-Aware KNN Laplacian for Video Matting", Li, Chen, and Tang. ICCV 2013.

- Key idea: producing spatio-temporally coherent pixel clusters of moving pixels
- Pixels sharing Similar appearance and similar motion should have similar α
- How to define the feature vector $X_t(i)$ at pixel i in frame t?

Feature Vector X

$$X_t(i) = (\lambda_s(x, y)\ \lambda_f(u_f, v_f, u_b, v_b)\ P(i, \lambda_p))_t$$

(x, y) are spatial coordinates of pixel i.
(u_f, v_f) and (u_b, v_b) are the forward and backward motion vectors.
$P(i, s)$ is an RGB image patch of size s centered at i

Asymmetric two-frame affinity matrix A

$$A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}_{2N \times 2N}$$

A_{11} and A_{22} are intra-frame affinity matrices.
A_{12} and A_{21} are inter-frame affinity matrices.

Alpha constraints

Using $\boldsymbol{\alpha}_t$ as soft constraints to optimize $\boldsymbol{\alpha}_{t+1}$
Avoiding trimap propagation

$$\mathbf{m} = \mathbf{m}_f + \mathbf{m}_b, \mathbf{m}_f = \begin{bmatrix} \mathbf{v}_f \\ \mathbf{0} \end{bmatrix}, \mathbf{m}_b = \begin{bmatrix} \mathbf{v}_b \\ \mathbf{0} \end{bmatrix}$$

$\mathbf{v}_b$ and $\mathbf{v}_f$ are $N \times 1$ indication vectors
$\mathbf{0}$ is an $N \times 1$ zero vector for $\boldsymbol{\alpha}_{t+1}$

Energy function

$$
\begin{aligned}
g(x) &= x^T L x + \lambda \left[\sum_{i \in \mathbf{m}_b} x_i^2 + \sum_{i \in \mathbf{m}_f} (1 - x_i)^2 \right] \\
&= x^T L x + \lambda \left[\sum_{i \in \mathbf{m}} x_i^2 - 2\mathbf{m}_f^T x + |\mathbf{m}_f| \right] \\
&= x^T (L + \lambda D) x - 2\lambda \mathbf{m}_f^T x + \lambda |\mathbf{m}_f|
\end{aligned}
$$

$$
x = \begin{bmatrix} \boldsymbol{\alpha}_t \\ \boldsymbol{\alpha}_{t+1} \end{bmatrix}, D = \mathrm{diag}(\mathbf{m})
$$

Solution

$$\frac{\partial g}{\partial x} = 2(L + \lambda D)x - 2\lambda \mathbf{m}_f^T = 0$$

$$x = (L + \lambda D)^{-1}(\lambda \mathbf{m}_f)$$

After solving the linear system, we obtain $\boldsymbol{\alpha}_{t+1}$ and the refined $\boldsymbol{\alpha}_t$.

Solved in Matlab using the biconjugate gradients stabilized method `bicgstab`

Further information

Evaluation website

http://www.alphamatting.com/index.html

Tutorial

http://www.alphamatting.com/ICCV2013_tutorial/

Sampling based matting

"Optimized Color Sampling for Robust Matting", Wang and Cohen. CVPR 2007.

"Image Matting with Local and Nonlocal Smooth Priors", Chen et al. CVPR 2013.

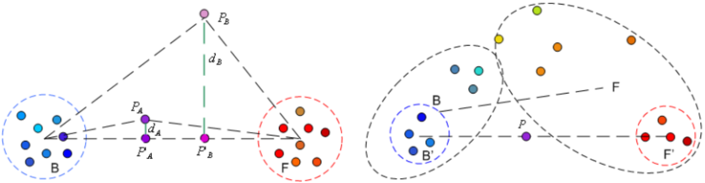
"Improving Image Matting Using Comprehensive Sampling Sets", Shahrian et al. CVPR 2013.

Color sampling [Wang and Cohen, CVPR 2007]

- Recall

$$\tilde{\alpha} = \frac{(C - B_j)(F_i - B_j)}{\|F_i - B_j\|^2}$$

- Failure cases



Confidence [Wang and Cohen, CVPR 2007]

- Distance ratio

$$R_d(F_i, B_j) = \frac{\|C - (\tilde{\alpha} F_i + (1 - \tilde{\alpha}) B_j\|}{\|F_i - B_j\|}$$

- Weights

$$w(F_i) = \exp\{-\|F_i - C\|^2 / \min_{i'}(\|F_{i'} - C\|)\}$$

$$w(B_j) = \exp\{-\|B_j - C\|^2 / \min_{j'}(\|B_{j'} - C\|)\}$$

- Confidence value

$$f(F_i, B_j) = \exp\left\{-R_d(F_i, B_j)^2 \cdot w(F_i) \cdot w(B_j) / \sigma^2\right\}$$

Local and nonlocal smooth priors [Chen et al., CVPR 2013]

- Data terms

$$W_{(i,F)} = \gamma\tilde{\alpha}\,, \quad W_{(i,B)} = \gamma(1-\tilde{\alpha})$$

- Local smooth term (i and j in a 3×3 window w_k

$$W_{ij}^{lap} = \eta \sum_{k}^{(i,j)\in w_k} \frac{1 + (C_i - \mu_k)(\Sigma_k + \epsilon I/9)^{-1}(C_j - \mu_k)}{9}$$

- Nonlocal smooth term

$$\sum_{i=1}^{N} \|X_i - \sum_{m=1}^{K} W_{im}^{lle} X_{im}\|^2\,, \text{subject to } \sum_{m=1}^{K} W_{im}^{lle} = 1\,.$$

$X_i = (r_i, g_i, b_i, x_i, y_i)$

LNSP optimization [Chen et al., CVPR 2013]

- Energy function

$$E = \lambda \sum_{i \in \mathcal{S}} (\alpha_i - g_i)^2 + \sum_{i=1}^{N} \left(\sum_{j \in N_i} W_{ij} (\alpha_i - \alpha_j) \right)^2$$

W_{ij} represents three kinds of weights, W_{ij}^{lap}, W_{ij}^{lle}, and $W_{(i,F)}$ and $W_{(i,B)}$.

$$E = (\alpha - G)^T \Lambda (\alpha - G) + \alpha^T L^T L \alpha$$

$$L_{ij} = \begin{cases} W_{ii}, & \text{if } i = j\,, \\ -W_{ij}, & \text{if } i \text{ and } j \text{ are neighbors}\,, \\ 0, & \text{otherwise}\,. \end{cases}$$

- Closed-form solution

$$(\Lambda + L^T L)\alpha = \Lambda G$$