More on natural image matting

Digital Visual Effects

Yung-Yu Chuang

with slides by

- A closed form solution to natural alpha matting, CVPR 2006

- With slides from Prof. Hwann-Tzong Chen

Linear relation (grayscale for now)

DigiVFX

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

Assumption: both F and B are approximately constant over a small window around each pixel

$$I_i \approx \alpha_i F + (1 - \alpha_i) B$$
$$I_i \approx \alpha_i (F - B) + B$$
$$\alpha_i \approx \frac{1}{F-B} I_i - \frac{B}{F-B}$$

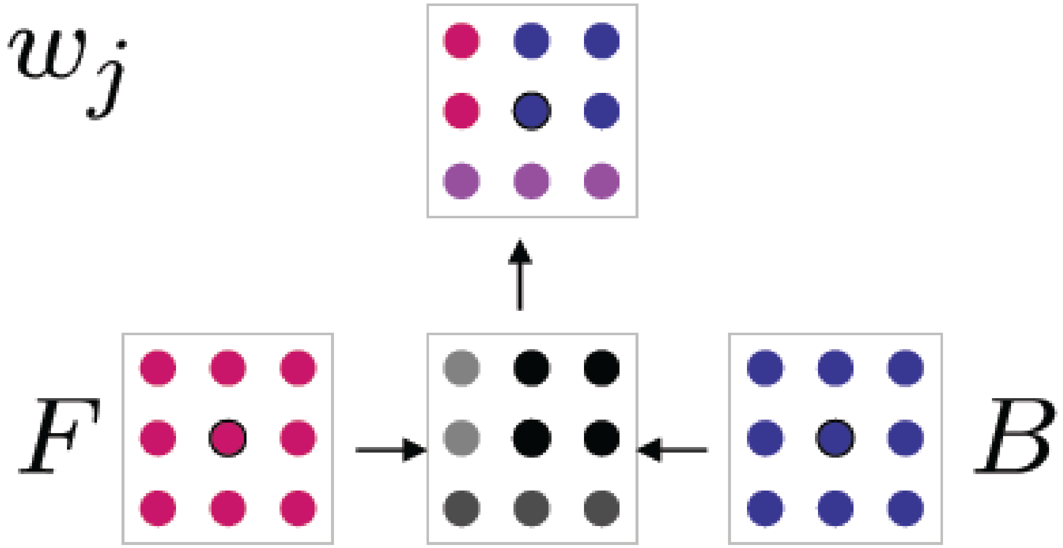
$$\alpha_i \approx a I_i + b, \quad \forall i \in w$$ ← a small window

$$a = \frac{1}{F-B} \qquad b = -\frac{B}{F-B}$$

Linear relation

$$\alpha_i \approx aI_i + b, \quad \forall i \in w$$ ← a small window

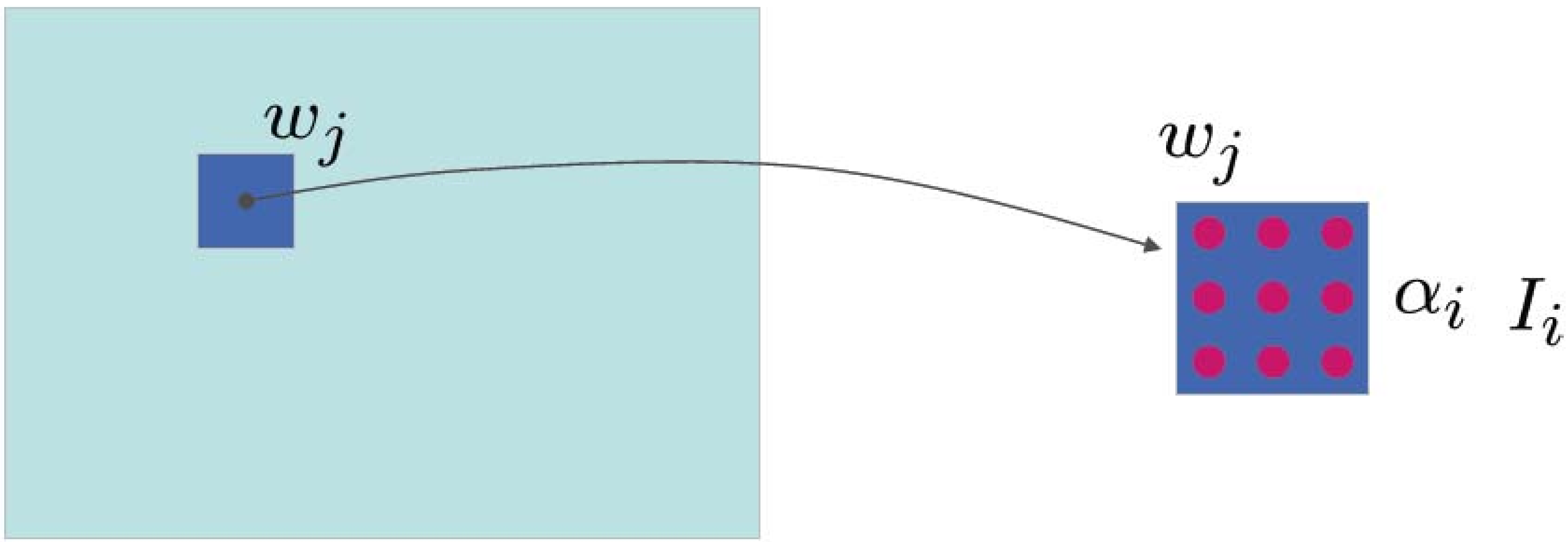
$$a = \frac{1}{F-B} \quad b = -\frac{B}{F-B}$$



Optimization

$$J(\alpha, a, b) = \sum_{j \in I} \left(\sum_{i \in w_j} (\alpha_i - a_j I_i - b_j)^2 + \epsilon a_j^2 \right)$$

where w_j is a small window around pixel j



Optimization

DigiVFX

$$J(\alpha, a, b) = \sum_{j \in I} \left(\sum_{i \in w_j} (\alpha_i - a_j I_i - b_j)^2 + \epsilon a_j^2 \right)$$

where w_j is a small window around pixel j

a regularization term on a:

minimizing the norm of a biases the solution towards smoother α mattes $\alpha_i \approx a\cancel{I_i} + b, \quad \forall i \in w$

$a \ll 0$ implies that F and B are very different

Optimization

$$J(\alpha, a, b) = \sum_{j \in I} \left(\sum_{i \in w_j} (\alpha_i - a_j I_i - b_j)^2 + \epsilon a_j^2 \right)$$

$$J(\alpha, a, b) = \sum_k \left\| \begin{pmatrix} I_1 & 1 \\ \vdots & \vdots \\ I_{|w_k|} & 1 \\ \sqrt{\epsilon} & 0 \end{pmatrix} \begin{pmatrix} a_k \\ b_k \end{pmatrix} - \begin{pmatrix} \alpha_1 \\ \vdots \\ \alpha_{|w_k|} \\ 0 \end{pmatrix} \right\|^2$$

$$J(\alpha, a, b) = \sum_k \left\| G_k \begin{bmatrix} a_k \\ b_k \end{bmatrix} - \bar{\alpha}_k \right\|^2$$

Optimization

DigiVFX

$$(a_k^*, b_k^*) = \operatorname{argmin} \left\| G_k \begin{bmatrix} a_k \\ b_k \end{bmatrix} - \bar{\alpha}_k \right\|^2$$

$$\begin{pmatrix} a_k^* \\ b_k^* \end{pmatrix} = (G_k^T G_k)^{-1} G_k^T \bar{\alpha}_k$$

Optimization

DigiVFX

$$\begin{pmatrix} a_k^* \\ b_k^* \end{pmatrix} = (G_k^T G_k)^{-1} G_k^T \bar{\alpha}_k$$

$$J(\alpha, a^*, b^*) = \sum_k \left\| G_k \begin{pmatrix} a_k^* \\ b_k^* \end{pmatrix} - \bar{\alpha}_k \right\|^2$$

$$\begin{aligned} J(\alpha) &= \sum_k \left\| \left(G_k (G_k^T G_k)^{-1} G_k^T - \mathbf{I} \right) \bar{\alpha}_k \right\|^2 \\ &= \sum_k \bar{\alpha}_k^T \bar{G}_k^T \bar{G}_k \bar{\alpha}_k \end{aligned}$$

$$\bar{G}_k = \mathbf{I} - G_k (G_k^T G_k)^{-1} G_k^T$$

Optimization

$$\bar{G}_k^T \bar{G}_k = (\mathbf{I} - G_k(G_k^T G_k)^{-1} G_k^T)^T (\mathbf{I} - G_k(G_k^T G_k)^{-1} G_k^T)$$
$$= \mathbf{I} - G_k(G_k^T G_k)^{-1} G_k^T$$

$$G_k = \begin{pmatrix} I_1 & 1 \\ \vdots & \vdots \\ I_{|w_k|} & 1 \\ \sqrt{\epsilon} & 0 \end{pmatrix}$$

the (i, j)-th element of $\mathbf{I} - G_k(G_k^T G_k)^{-1} G_k^T$ is

$$\delta_{ij} - \frac{1}{|w_k|}\left(1 + \frac{1}{\frac{\epsilon}{|w_k|} + \sigma_k^2}(I_i - \mu_k)(I_j - \mu_k)\right)$$

Optimization

DigiVFX

The (i,j) element

$$\left(\mathbf{I} - G_k \left(G_k^T G_k\right)^{-1} G_k^T\right)_{ij}$$

$$= \delta_{ij} - (I_i \;\; 1) \begin{pmatrix} \sum_n^{|w_k|} I_n^2 + \epsilon & \sum_n^{|w_k|} I_n \\ \sum_n^{|w_k|} I_n & |w_k| \end{pmatrix}^{-1} \begin{pmatrix} I_j \\ 1 \end{pmatrix}$$

$$\mathbf{I} = \begin{pmatrix} 1 & & & \\ & 1 & & \\ & & \ddots & \\ & & & 1 \end{pmatrix}$$

$$\begin{pmatrix} I_1 & 1 \\ I_2 & 1 \\ \vdots & \\ I_{|w_k|} & 1 \\ \sqrt{\epsilon} & 0 \end{pmatrix} \left\{ \begin{pmatrix} I_1 & I_2 & \dots & I_{|w_k|} & \sqrt{\epsilon} \\ 1 & 1 & & 1 & 0 \end{pmatrix} \begin{pmatrix} I_1 & 1 \\ I_2 & 1 \\ \vdots & \\ I_{|w_k|} & 1 \\ \sqrt{\epsilon} & 0 \end{pmatrix} \right\}^{-1} \begin{pmatrix} I_1 & I_2 & \dots & I_{|w_k|} & \sqrt{\epsilon} \\ 1 & 1 & & 1 & 0 \end{pmatrix}$$

The inverse

DigiVFX

$$\begin{pmatrix} \sum_n^{|w_k|} I_n^2 + \epsilon & \sum_n^{|w_k|} I_n \\ \sum_n^{|w_k|} I_n & |w_k| \end{pmatrix}^{-1}$$

$$= \frac{\begin{pmatrix} |w_k| & -\sum_n^{|w_k|} I_n \\ -\sum_n^{|w_k|} I_n & \sum_n^{|w_k|} I_n^2 + \epsilon \end{pmatrix}}{|w_k| \sum_n^{|w_k|} I_n^2 + \epsilon |w_k| - (\sum_n^{|w_k|} I_n)^2}$$

$$= \frac{|w_k| \begin{pmatrix} 1 & -\mu_k \\ -\mu_k & \sum_n^{|w_k|} I_n^2/|w_k| + \epsilon/|w_k| \end{pmatrix}}{|w_k|^2 \sigma_k^2 + \epsilon |w_k|}$$

$$= \frac{1}{|w_k| \sigma_k^2 + \epsilon} \begin{pmatrix} 1 & -\mu_k \\ -\mu_k & \sum_l^{|w_k|} I_n^2/|w_k| + \epsilon/|w_k| \end{pmatrix}$$

Optimization

The (i,j) element

$$\left(\mathbf{I} - G_k\left(G_k^T G_k\right)^{-1} G_k^T\right)_{ij}$$

$$= \delta_{ij} - (I_i \ 1)\begin{pmatrix} \sum_n^{|w_k|} I_n^2 + \epsilon & \sum_n^{|w_k|} I_n \\ \sum_n^{|w_k|} I_n & |w_k| \end{pmatrix}^{-1}\begin{pmatrix} I_j \\ 1 \end{pmatrix}$$

$$= \delta_{ij} - (I_i \ 1)\frac{1}{|w_k|\sigma_k^2 + \epsilon}\begin{pmatrix} 1 & -\mu_k \\ -\mu_k & \sum_n^{|w_k|} I_n^2/|w_k| + \epsilon/|w_k| \end{pmatrix}\begin{pmatrix} I_j \\ 1 \end{pmatrix}$$

$$= \delta_{ij} - \frac{1}{|w_k|\sigma_k^2 + \epsilon}\left(I_i I_j - I_i\mu_k - I_j\mu_k + \frac{\sum_n^{|w_k|} I_n^2 + \epsilon}{|w_k|}\right)$$

$$= \delta_{ij} - \frac{1}{|w_k|\sigma_k^2 + \epsilon}\left(I_i I_j - I_i\mu_k - I_j\mu_k + \mu_k^2 + \frac{\sum_n^{|w_k|} I_n^2}{|w_k|} - \mu_k^2 + \frac{\epsilon}{|w_k|}\right)$$

$$= \delta_{ij} - \frac{1}{|w_k|\sigma_k^2 + \epsilon}\left((I_i - \mu_k)(I_j - \mu_k) + \sigma_k^2 + \frac{\epsilon}{|w_k|}\right)$$

$$= \delta_{ij} - \frac{1}{|w_k|}\left(1 + \frac{1}{\sigma_k^2 + \epsilon/|w_k|}(I_i - \mu_k)(I_j - \mu_k)\right)$$

Optimization

$$J(\alpha) = \sum_k \bar{\alpha}_k^T \bar{G}_k^T \bar{G}_k \bar{\alpha}_k$$

$$J(\alpha) = \alpha^T L \, \alpha$$

L is a large sparse N-by-N matrix whose (i, j) element is

$$\sum_{k|(i,j)\in w_k} \left(\delta_{ij} - \frac{1}{|w_k|} \left(1 + \frac{1}{\sigma_k^2 + \epsilon/|w_k|} (I_i - \mu_k)(I_j - \mu_k) \right) \right)$$

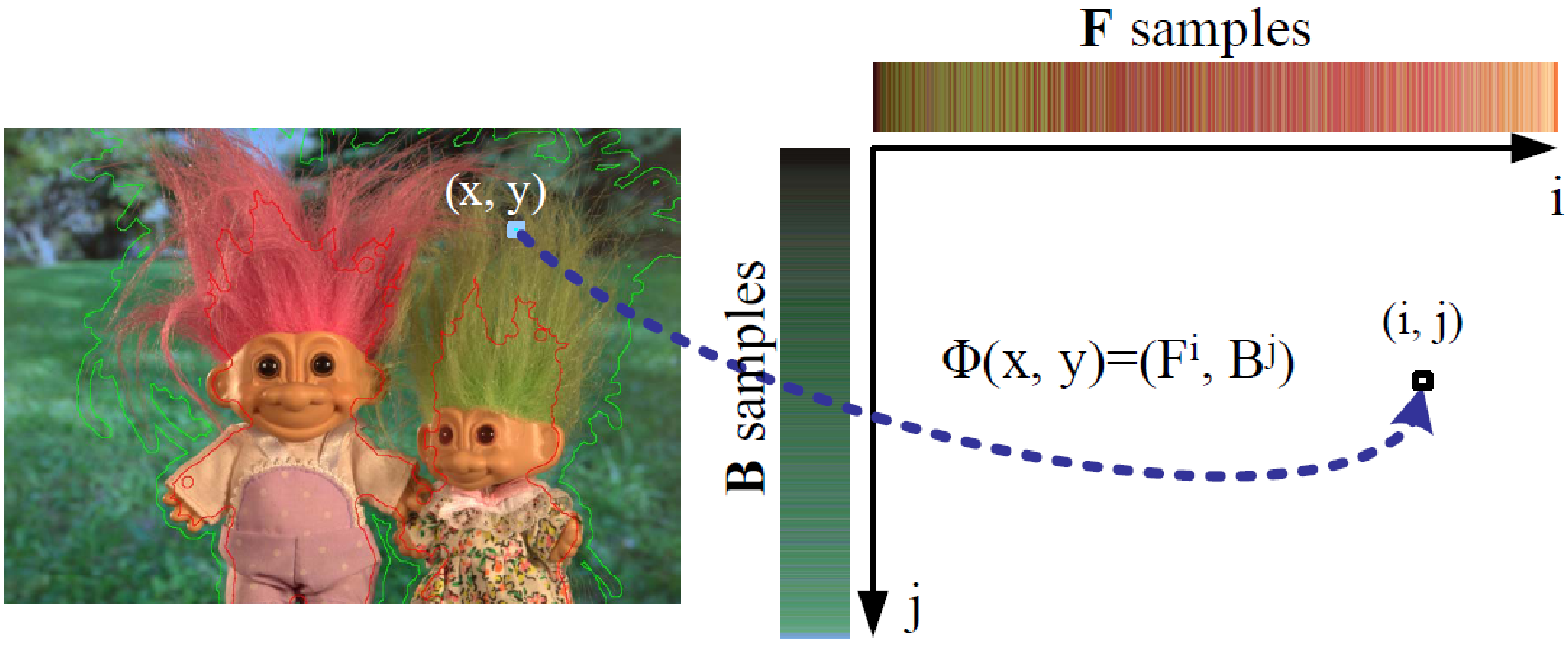
N is the number of pixels in the image

- A global sampling method for alpha matting, CVPR 2011

Idea

input | robust | improved color | shared | global

Search space



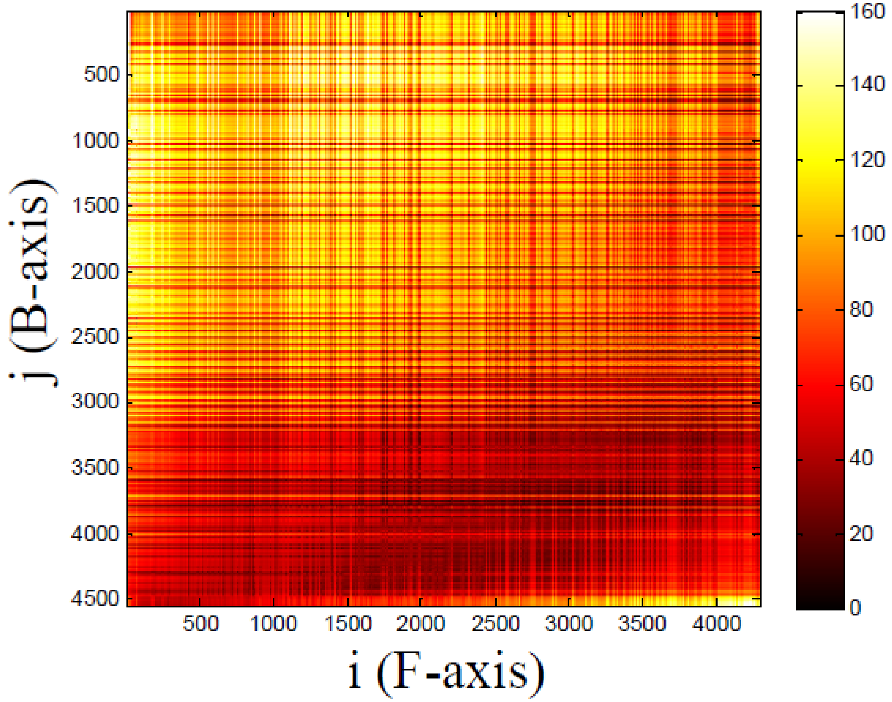
Propagation and random search

- Propagation

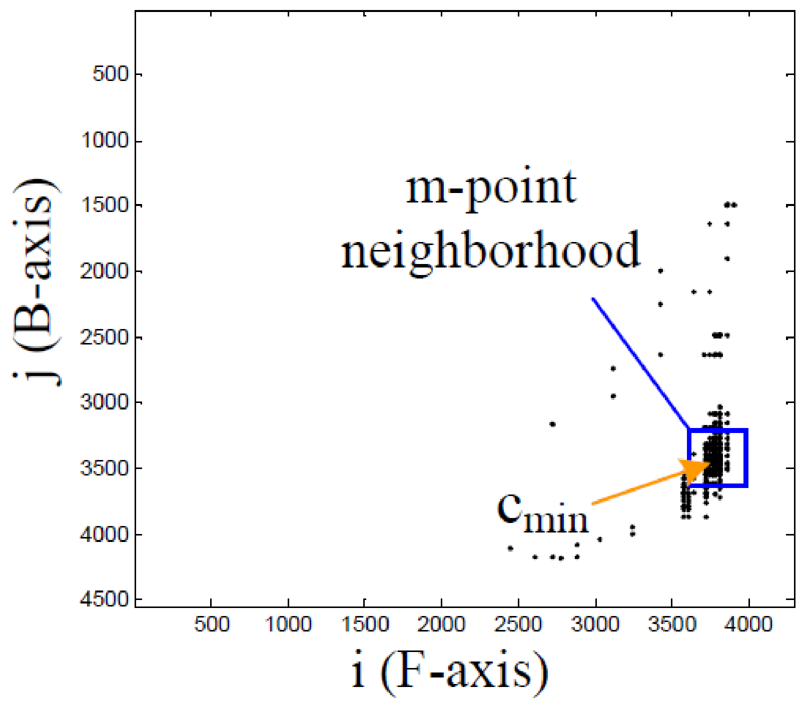
$$\Phi(x,y) \leftarrow \arg\min_{\Phi(x',y')} \mathcal{E}(\Phi(x',y'))$$

- Random search

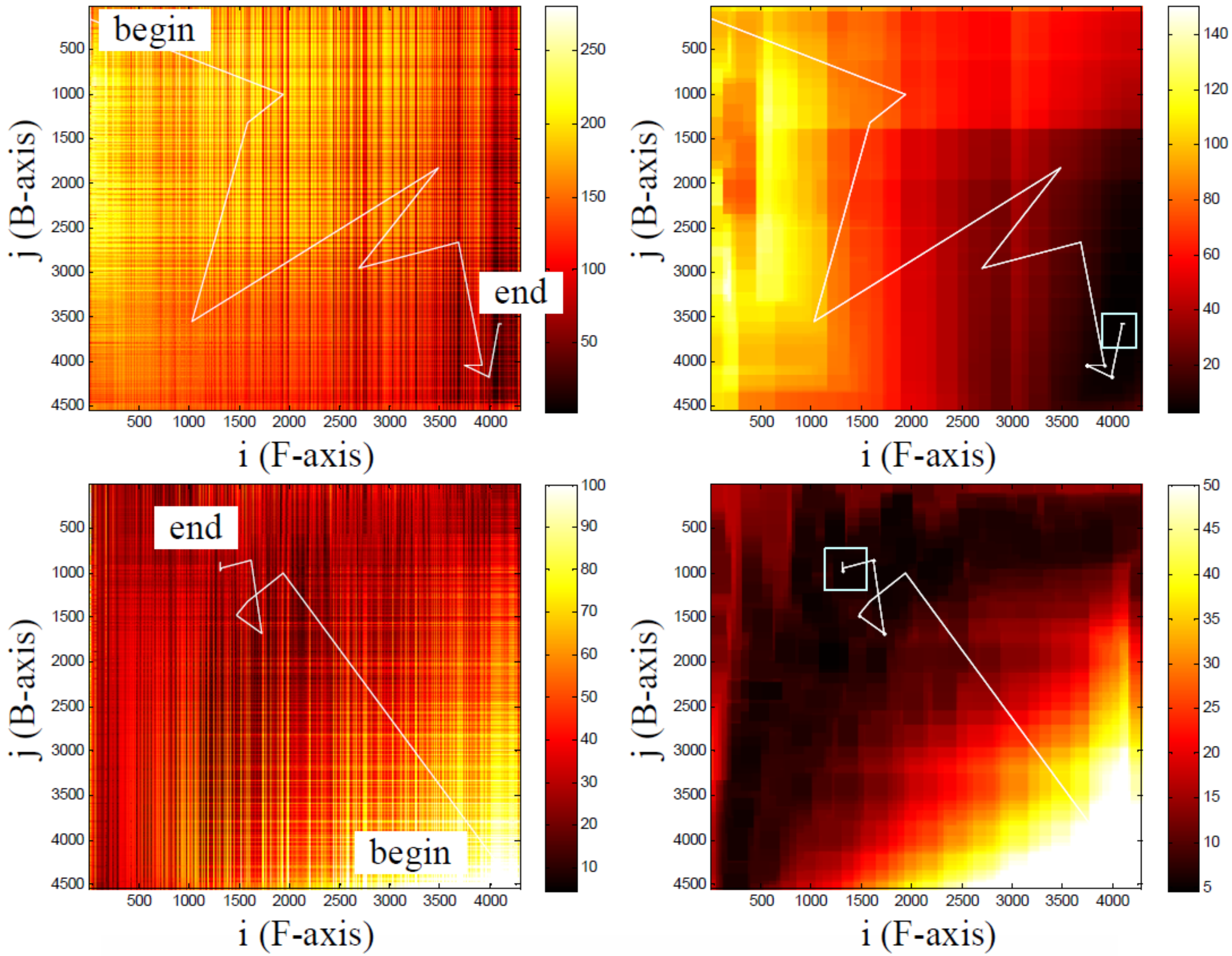
$$(i_k, j_k) = (i, j) + \omega\beta^k \mathbf{R}_k$$



160
140
120
100
80
60
40
20
0
500
1000
1500
2000
2500
3000
3500
4000
4500
j (B-axis)
i (F-axis)



m-point
neighborhood
c_{min}
500
1000
1500
2000
2500
3000
3500
4000
4500
j (B-axis)
i (F-axis)


begin
end
j (B-axis)
i (F-axis)
500 1000 1500 2000 2500 3000 3500 4000 4500
500 1000 1500 2000 2500 3000 3500 4000
250 200 150 100 50
140 120 100 80 60 40 20
100 90 80 70 60 50 40 30 20 10
50 45 40 35 30 25 20 15 10 5

- KNN Matting, CVPR 2012

Nonlocal principle

DigiVFX

$$
\begin{aligned}
E[X(i)] &\approx \sum_j X(j)k(i,j)\frac{1}{\mathcal{D}_i}, \\
k(i,j) &= \exp\left(-\frac{1}{h_1^2}\|X(i)-X(j)\|_g^2 - \frac{1}{h_2^2}d_{ij}^2\right) \\
\mathcal{D}_i &= \sum_j k(i,j).
\end{aligned}
$$

Nonlocal principle for mattes

DigiVFX

$$E[\alpha_i] \approx \sum_j \alpha_j k(i,j) \frac{1}{\mathcal{D}_i}$$

$$\mathcal{D}_i \alpha_i \approx k(i,\cdot)^T \boldsymbol{\alpha}$$

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

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

Kernels and features

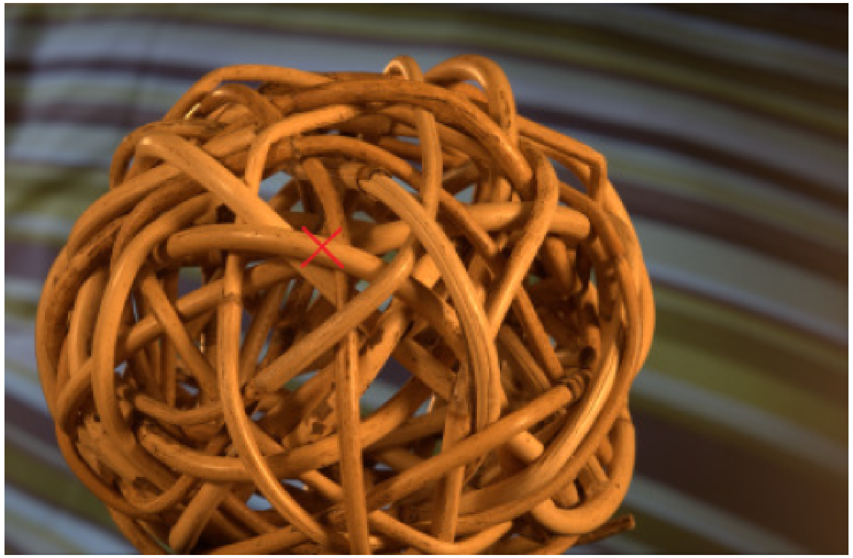
DigiVFX

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

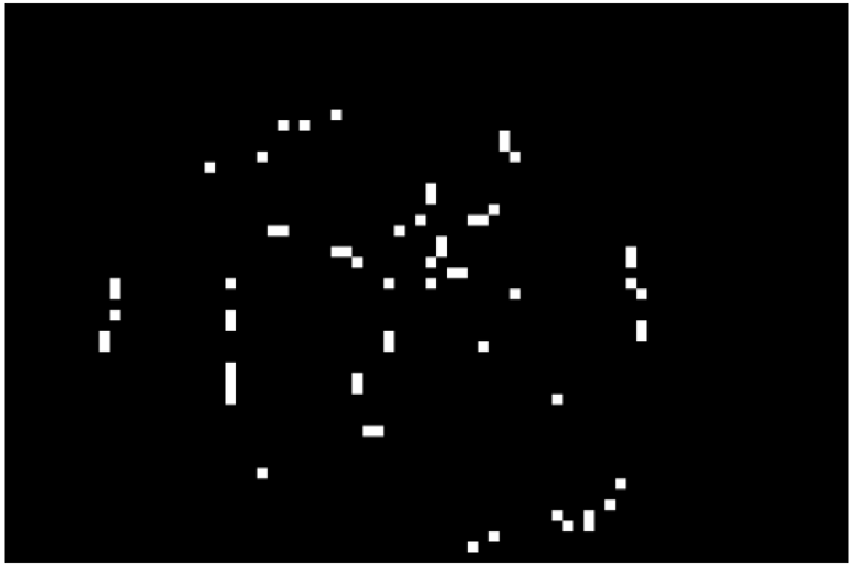
$$k(i,j) = 1 - \frac{||X(i) - X(j)||}{C}$$

C is the least upper bound of $||X(i) - X(j)||$

KNN matting



input pixel (red)

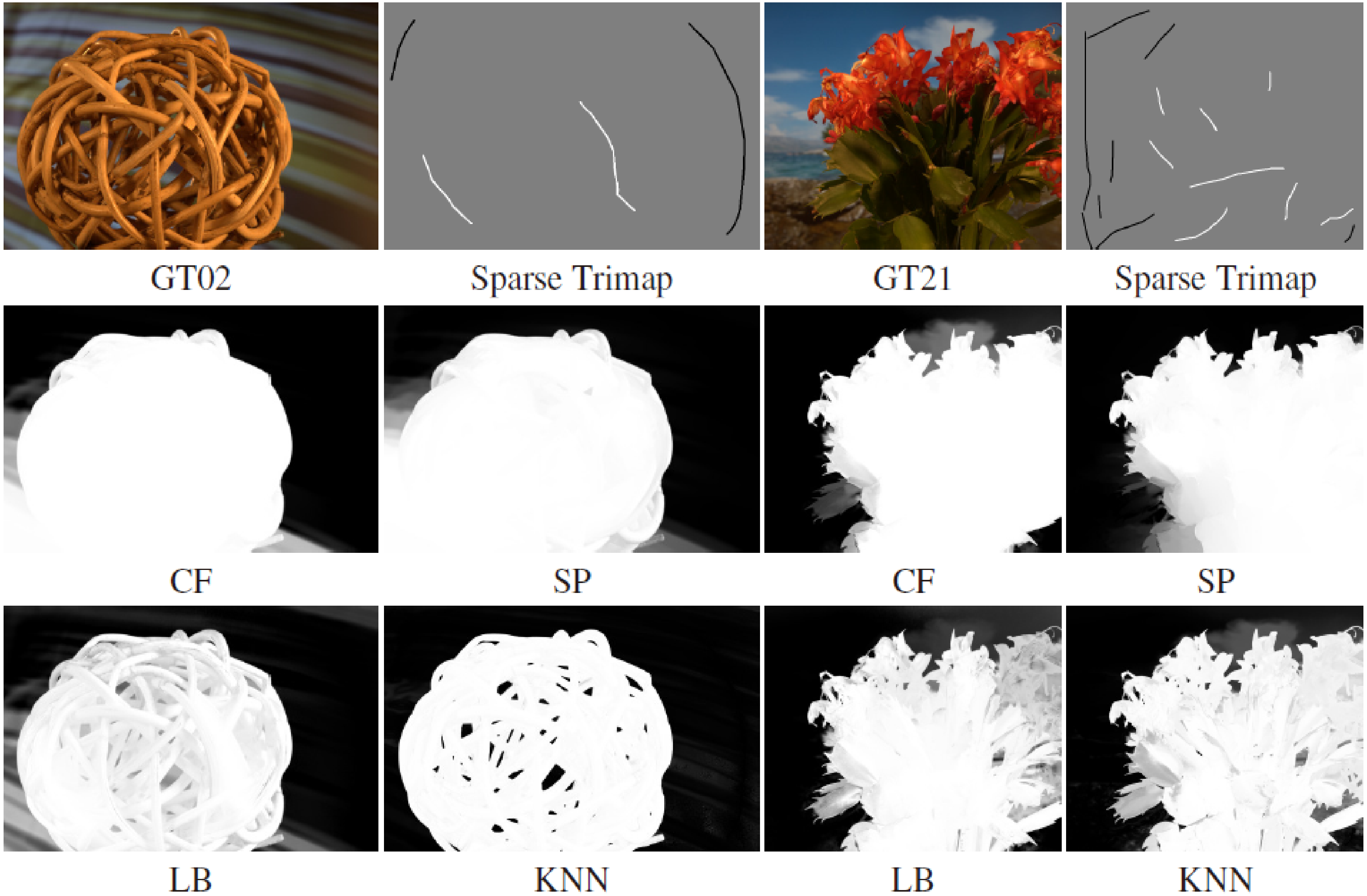


KNN (10^{-5}sec)

Results

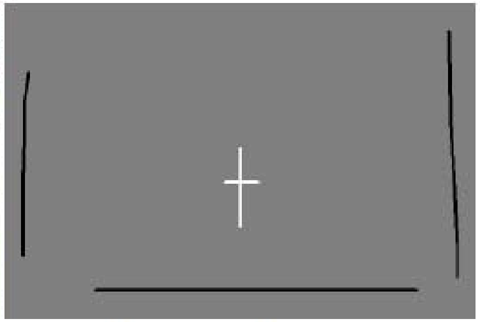
	overall	avg user	pine-apple	plastic bag	normalized score(%)
Shared	3.6	3.5	2	7	79.6
Segmentation	4.2	4	5	9	77.2
KNN	**4.3**	**3.6**	**1**	**1**	**84.6**
Improved color	4.4	4	4	3	75.7
Learning-based	5.9	6.4	12	2	67.8
Closed-Form	6	7.4	10	5	66.1
Shared (real time)	6.1	5.8	3	8	65.4
Large Kernel	6.8	6.5	6	4	62.4
Robust	7.5	8.1	8	6	55.9
High-res	8.5	8.1	9	13	51.5

Results

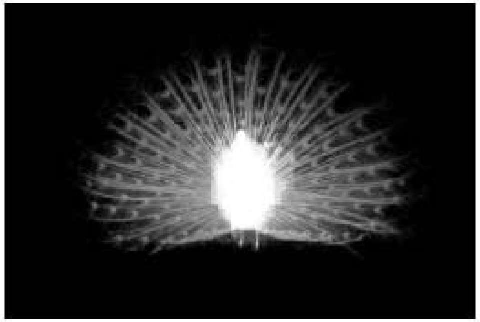

GT02
Sparse Trimap
GT21
Sparse Trimap
CF
SP
CF
SP
LB
KNN
LB
KNN

Results

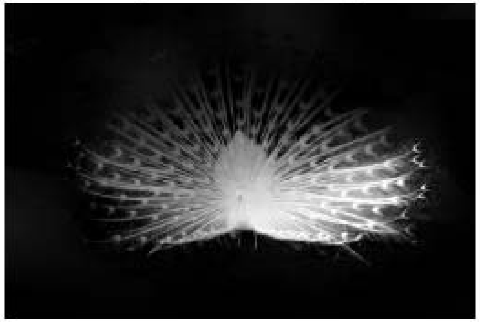
Peacock



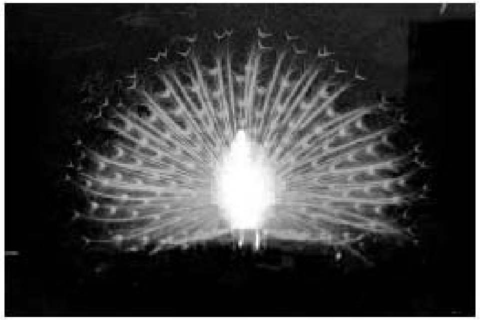
Sparse Trimap



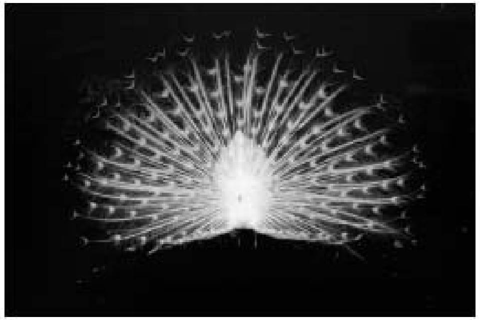
CF



SP



LB



KNN