Natural Image Matting using Deep Convolutional Neural Networks

Donghyeon Cho[1], Yu-Wing Tai[2], In So Kweon[1]

KAIST[1], SenseTime[2]

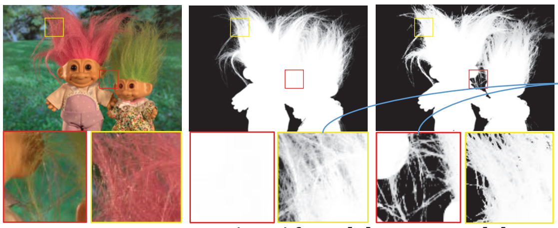
Introduction

Motivation

- There is a synergistic effect between **local** and **nonlocal** matting methods.
- So far, there are no effective ways to **combine** there two kinds of methods without lo sing the advantages of both methods.

Objective

- Effectively combine alpha mattes of local and nonlocal methods using deep CNN model to reconstruct higher quality alpha mattes than both of its inputs.
- We choose the closed form matting [1] and KNN matting [2] as representative methods for local and nonlocal methods, respectively.

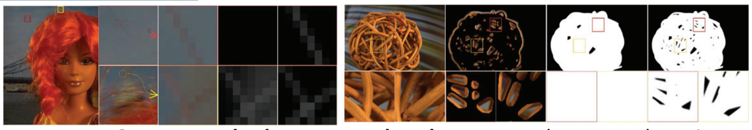


RGB image Closed form [1] KNN [2]

Better!!
How can we combine both results properly?

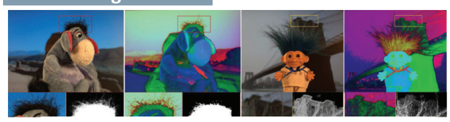
Reviews of closed form (local) and KNN matting (nonlocal)

Closed form matting



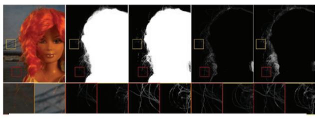
Over-smoothed **Isolated** can never be correctly estimated

KNN matting



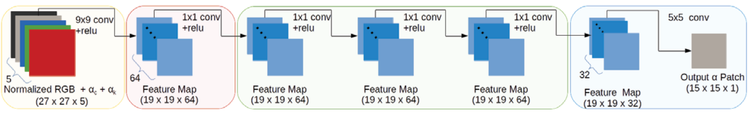
It is difficult to define a **universal feature space** to find nonlocal neighbors.

Closed form and KNN matting



-**Closed form matting[1]** performs better in preserving **local smoothness** which has smaller errors in sharp, and short hair regions.
-**KNN matting[2]** performs better in protecting long **hair regions** as shown in the zoom-in regions.

Deep CNN Matting

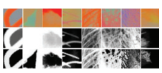


$$\alpha = \mathcal{F}(\bar{I}, \alpha_c, \alpha_k)$$

Normalized RGB

$$\bar{I} = \frac{I}{\sqrt{r^2+g^2+b^2}}$$

	Closed form, α_c [1]	KNN, α_k [2]
Key properties	Local Few parameter Visually pleasing RGB space	Non-local Few parameter Fine structure HSV space

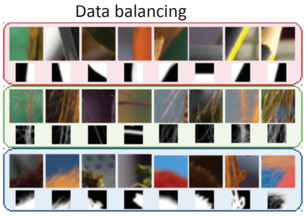


Examples of input

Loss function (Euclidian loss)

$$L = \frac{1}{n}\sum_{i=1}^{n}\sqrt{(F(\bar{I}, \alpha_c, \alpha_k) - \bar{\alpha}_i)^2}$$

Data balancing

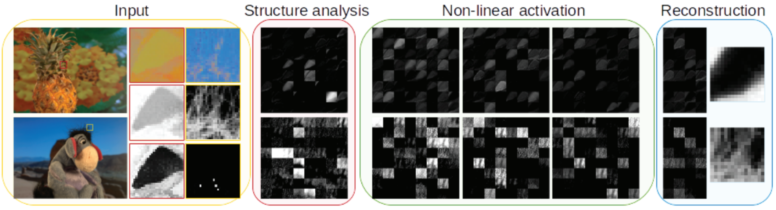


Data augmentation

Learning details

Training time	2~3 days
Number of iterations	10^6
Learning rate	10^{-5}
Momentum	0.9
Batch size	128
CPU	I7 3.4GHz CPU
GPU	GTX 760
Testing time (800x640)	15~25 seconds

Input Structure analysis Non-linear activation Reconstruction



Experiments

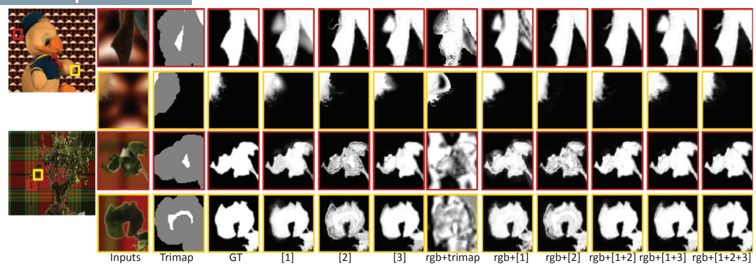
Evaluation

Sum of Absolute Differences	overall rank	avg. small rank	avg. large rank	avg. user rank
DCNN Matting	**2.7**	4.1	1.3	2.8
CSC Matting	**9.3**	12.9	5.5	9.4
LNSP Matting	**10**	6.5	9.6	13.9
Graph-based sparse matting	**10.3**	10.6	10.8	9.4
Patch-based Matting	**10.3**	5.8	11.9	13.3
KL-Divergence Based Sparse Sampling	**10.9**	10.3	10.1	12.4
TSPS-RV Matting	**11.9**	10.5	10.8	14.5
Iterative Transductive Matting	**12.6**	13.8	12.3	11.8
Comprehensive sampling	**12.8**	10.8	12.8	14.8
SVR Matting	**13.2**	16	12.5	11.1

Mean Squared Error	overall rank	avg. small rank	avg. large rank	avg. user rank
DCNN Matting	**3.3**	4.3	1.5	4
LNSP Matting	**8.6**	6.3	8	11.5
Patch-based Matting	**9.4**	6.1	9.5	12.6
KL-Divergence Based Sparse Sampling	**10.8**	10.3	9.6	12.5
CCM	**11.1**	13.9	11.5	8
Graph-based sparse matting	**11.5**	11.9	11.8	10.9
TSPS-RV Matting	**12**	12.3	8.9	14.9
Comprehensive sampling	**12.2**	11	12.3	13.4
SVR Matting	**12.5**	16	11.1	10.5
Comprehensive Weighted Color and Texture	**13.3**	13.1	14.4	12.4

Gradient error	overall rank	avg. small rank	avg. large rank	avg. user rank
DCNN Matting	**6.5**	8.4	5.5	5.5
Graph-based sparse matting	**9.7**	7.9	8.1	13.1
Patch-based Matting	**9.9**	7.1	9.1	13.5
KL-Divergence Based Sparse Sampling	**10**	8.1	8.6	13.1
LNSP Matting	**10.3**	7.8	10.1	13
Comprehensive sampling	**11.1**	11.3	10.8	11.4
CCM	**13.2**	15.3	13.1	11.3
SVR Matting	**13.2**	15.3	14.6	9.8
Sparse coded matting	**13.3**	14.6	12.4	13
Segmentation-based matting	**13.8**	16.8	12.3	12.4

Various inputs

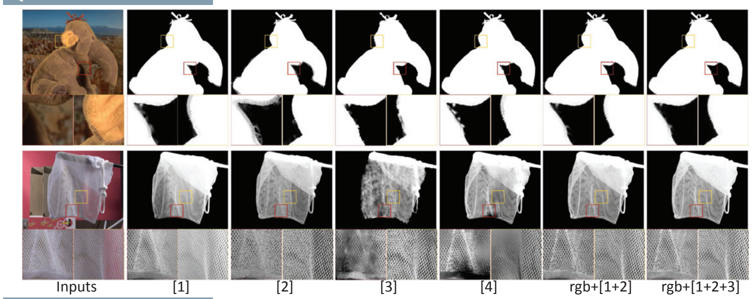


Inputs Trimap GT [1] [2] [3] rgb+trimap rgb+[1] rgb+[2] rgb+[1+2] rgb+[1+3] rgb+[1+2+3]

The number of layers

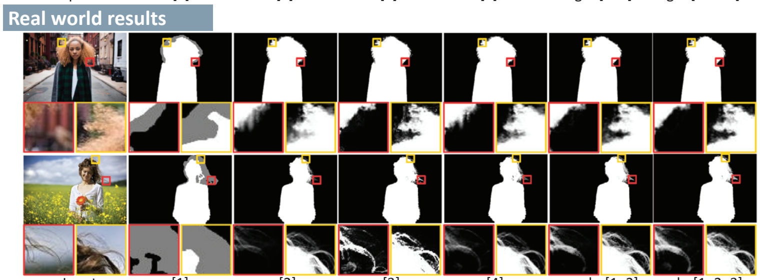
[Avg.]	Sum of Absolute Difference	Mean Squared Error	Gradient Error	Time (Sec.)
Single layer	11.10	0.596	0.842	4.617
Four layers	10.30	0.563	0.804	5.126
Eight layers	10.17	0.546	0.792	6.528

Qualitative results



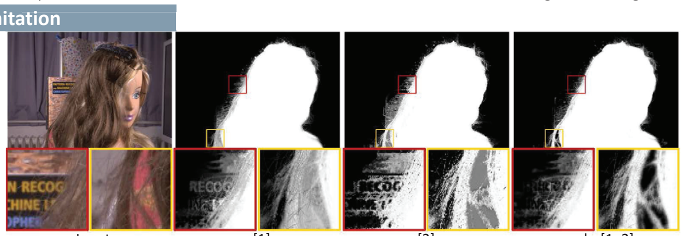
Inputs [1] [2] [3] [4] rgb+[1+2] rgb+[1+2+3]

Real world results



Inputs [1] [2] [3] [4] rgb+[1+2] rgb+[1+2+3]

Limitation



Inputs [1] [2] rgb+[1+2]

Conclusion

Contribution

- We introduce a deep CNN model for natural image matting.
- Our deep CNN model can effectively combine alpha mattes of local and nonlocal methods to reconstruct higher quality alpha mattes than both of its inputs.
- Our deep CNN method demonstrates outstanding performance.

[1] A. Levin, D. Lischinski, and Y. Weiss. A closed-form solution to natural image matting. IEEE Trans. Pattern Anal. Mach. Intell. (TPAMI), 30(2):0162–8828, 2008.
[2] Q. Chen, D. Li, and C.-K. Tang. Knn matting. In Proc. of Computer Vision and Pattern Recognition (CVPR), 2012.
[3] E. Shahrian and D. Rajan. Weighted color and texture sample selection for image matting. In Proc. of Computer Vision and Pattern Recognition (CVPR), 2012.
[4] E. Shahrian, D. Rajan, B. Price, and S. Cohen. Improving image matting using comprehensive sampling sets. In Proc. of Computer Vision and Pattern Recognition (CVPR), 2013.