A Learning-Based Approach to Reduce JPEG Artifacts in Image Matting

Inchang Choi[1]   Sunyeong Kim[1]   Michael S. Brown[2]   Yu-Wing Tai[1]
Korea Advanced Institute of Science and Technology (KAIST)[1]
National University of Singapore (NUS)[2]

Abstract

Single image matting techniques assume high-quality input images. The vast majority of images on the web and in personal photo collections are encoded using JPEG compression. JPEG images exhibit quantization artifacts that adversely affect the performance of matting algorithms.

To address this situation, we propose a learning-based post-processing method to improve the alpha mattes extracted from JPEG images. Our approach learns a set of sparse dictionaries from training examples that are used to transfer details from high-quality alpha mattes to alpha mattes corrupted by JPEG compression. Three different dictionaries are defined to accommodate different object structure (long hair, short hair, and sharp boundaries). A back-projection criteria combined within an MRF framework is used to automatically select the best dictionary to apply on the object's local boundary. We demonstrate that our method can produces superior results over existing state-of-the-art matting algorithms on a variety of inputs and compression levels.

1. Introduction

Single image matting is a problem that has been extensively studied in computer vision and graphics. The goal is to segment a foreground object from the background, where each foreground pixel is assigned a value between $[0, 1]$. Virtually all image matting techniques assume that the input image is free of notable compression artifacts. This assumption, however, is no longer valid as image editing is becoming a common tool for everyday users who now wish to edit internet images, web albums, and photos captured from cell phone cameras. The vast majority of these images are encoded using JPEG compression that exhibits compression artifacts that adversely affect the quality of the extracted alpha matte. Figure 1 demonstrates how various levels of JPEG compression (expressed as quality levels) affect the performance of several state-of-the-art matting methods.
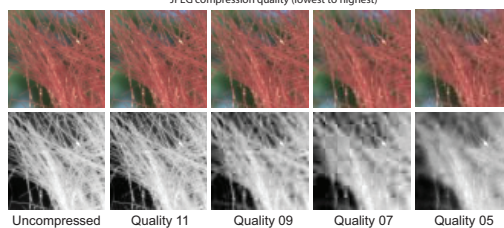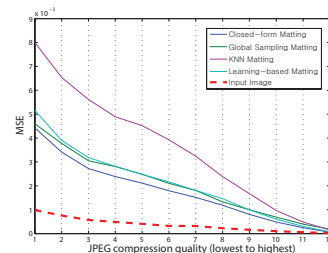

Figure 1. The RMSE of alpha mattes extracted from several state-of-the-art methods are plotted for different JPEG compression qualities (1 = lowest; 12 = highest)[1]. Compared to the RMSE of the pixel values, the RMSE of the corresponding alpha mattes increases dramatically as the compression ratio increases.

Note that the root mean square error (RMSE) of the alpha mattes suffers worse than the RMSE of the compressed image itself. Even "high quality" JPEG images (e.g quality 9[2]), which represent common compression level used on camera phones and social media sites, have subtle artifacts that can adversely affect the resulting alpha mattes.

The contribution of this paper is to propose a learning-based post-processing method for improving alpha mattes in the face of JPEG compression artifacts. Specifically, we propose a technique that learns a set of sparse dictionaries to transfer high-quality details to low-quality alpha mattes extracted from compressed images. Because matting boundaries are far more complicated than natural image boundaries, our methods uses three different dictionaries (i.e. long hair, short hair, and sharp boundaries) for various types of object structure and selects the best one locally about the object's boundary. While our approach cannot compete with mattes extracted from uncompressed input images, it is useful in improving the quality of alpha mattes over the

---

[1]The 1-12 quality scale is common for photo editing software (e.g. Photoshop), however, other schemes use a finer scale from 0-100.

[2]Photoshop categorized quality 10-12 to be *maximum quality*, 8-9 to be *high quality*, 5-7 to be *medium quality*, and 1-4 to be it low quality.

state-of-the-art in the likely event that the input images have been compressed. We tested our approach extensively on a variety of inputs and compression levels which demonstrates superior remedy results over state-of-the-art matting algorithms in comparisons with their initial corrupted alpha mattes.

2. Related Work

Works related to JPEG compression, image matting, and image super resolution are discussed. Since each of these represent well-studied areas on their own, only relevant and representative works are discussed.

JPEG compression The Joint Photographic Experts Group (JPEG) compression standard developed over two decades ago is the most widely adopted image compression method to date (for further details see [25]). Because JPEG is lossy, the uncompressed image contains errors that are in the form of frequency domain ringing and blocking artifacts that are collectively referred to as compression artifacts. For natural images it is often difficult to perceptually see these errors even for images with medium compression qualities around 5-7. While the compressed images may be perceptually acceptable, the resulting compression artifacts are well known to adversely affect low-level image processing routines. As shown in Figure 1, image matting is no exception.

There are several post-processing methods (e.g. [28, 1, 2, 6]) aimed at reducing JPEG artifacts. These approaches target low-quality compressed images and perform various types of filtering to reduce blocking and ringing artifacts. While such approaches can improve the visual quality of the input image, they tend to smooth out high frequency details. As shown in our results, we achieve better results than deblocking applied either as pre-processing to the input image or as post-processing to the alpha matte.

Image matting Image matting approaches (for a nice overview see [26]) can be roughly classified into two categories: affinity-based methods and sampling-based methods. Affinity-based methods (e.g. [19, 12, 11, 3]) estimated alpha values for the unknown region by propagating the known alpha values in accordance with the pixel affinities. Affinity-based approaches propagate alpha values well in uncompressed and maximum-quality JPEG images (e.g. quality level 12). However, these methods fail to effectively propagate the alpha values across block artifacts when an image is compressed. Sampling-based methods (e.g. [4, 27, 7, 8]) estimate alpha mattes by sampling the foreground and background color. For each pixel with an unassigned alpha value, these approaches find the most plausible pair of the foreground and the background pixels around it and solve the matting compositing equation with the sampled color pairs. As with affinity-based methods, sample-based methods are adversely affected by the ring-

ing artifacts and quantization across different blocks. For both approaches, obtaining mattes with detailed structure is difficult due to the blurring effect introduced by the DCT quantization.

Learning-based super resolution Our work is closely related to learning-based image super resolution (SR) methods (for a nice overview see [20]). These approaches use a training-set to learn the relationship between high-resolution image patches and their corresponding down-sampled (i.e. low-resolution) image patches [29, 9]. These learning-based super resolution methods can be applied to first to enhance a JPEG image. However, we found that these tend to produce smoothed results. This is often because these methods first apply a smoothing to the input image to reduce ringing and blocking artifacts (e.g. as done in Kwon *et al.* [9]) which can remove high-frequency information before the matting is applied. Moreover, these approaches target natural images. Our work, however, directly operates on alpha mattes instead of pixel intensity. Since matte boundaries are more complicated than natural image boundaries (e.g. mostly sharp boundaries), we find it necessary to learn different dictionary for different boundary structures. As a result, our methods requires an additional step to select the correct dictionary to use in a local manner.

3. Matting for JPEG images

As previously discussed, we adopt a learning-based approach used in single image super-resolution for our matting problem. Our input is a low-quality alpha matte extracted from conventional matting algorithms, e.g. [12, 8, 3, 30]. Our goal is to estimate a high-quality alpha matte by transferring details via a dictionary learned from high-quality and low-quality alpha matte patch pairs.

In the following, we denote y be the input low-quality alpha matte, x be the output high-quality alpha matte.

3.1. Problem definition and overview

Following the work from [29], we assume the alpha matte within each 8×8 block can be sparsely represented as a linear combination of a set of basis functions:

$$y = D_l\phi, \tag{1}$$

where $\phi \in \mathbb{R}^k$ is a vector of sparse coefficients, e.g. $\|\phi\|_0 \ll k$, and D_l is a dictionary containing basis functions learned from low-quality alpha mattes extracted from JPEG compressed images.

In a similar context, we define the high-quality alpha matte patch as:

$$x = D_h\phi, \tag{2}$$

where D_h is a dictionary learned from high-quality alpha mattes extracted from images without any compression.
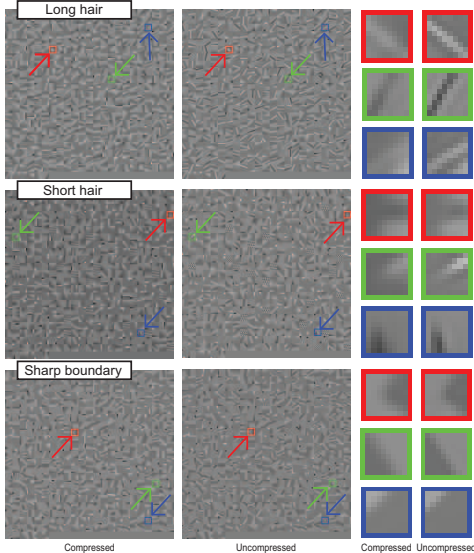
Figure 2. The learned dictionaries computed from different training examples group as *long hair*, *short hair*, and *sharp boundary*. Basis from each dictionary adopts to the different structures of training examples.

The two dictionaries, D_l and D_h, are co-trained using a set of alpha matte pairs. Where each pair contains an alpha matte extracted from the high-quality input image (either uncompressed or compressed at a high-quality, e.g. JPEG quality 12) and the alpha matte extracted from the same image with lower-quality compression, e.g. JPEG quality 7. Consequently, D_l and D_h have one-to-one correspondences and share the same sparse coefficients ϕ. Hence, for a given low-quality alpha matte input, we can first estimate the sparse coefficients ϕ and then replace the low-quality dictionary D_l with the high-quality dictionary D_h to reconstruct a high-quality matte.

3.2. Joint dictionary training

Sparse coding is used to learn the dictionary as follows:

$$D_c = \arg\min_{D_c, Z} = \|X_c - D_c Z\|_2^2 + \lambda\|Z\|_1, \qquad (3)$$

where D_c is the target dictionary with l_2-norm to measure the fidelity to represent training data X_c, Z is the linear combination coefficients with l_1-norm to enforce sparsity, and λ is the parameter to balance the two terms.

In our case, we follow [29] to substitute:

$$X_c = \begin{bmatrix} X^h \\ Y^l \end{bmatrix}, D_c = \begin{bmatrix} D_h \\ D_l \end{bmatrix}, \qquad (4)$$

where $X^h = \{x_1, x_2, \ldots, x_n\}$ is the set of high-quality 8×8 alpha mattes and $Y^l = \{y_1, y_2, \ldots, y_n\}$ is the set of the corresponding low-quality 8×8 alpha mattes. Since the size of the high- and low-quality alpha matte pairs are the same,
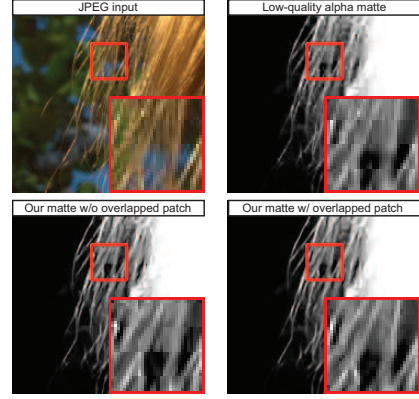

Figure 3. Example of the improvement gained by imposing a neighborhood consistency by using overlapping patches.

we can use the same normalization factor to learn D_l and D_h. Thus, we have:

$$\{D_h, D_l\} = \arg\min_{\{D_h, D_l, Z\}} \|X_c - D_c Z\|_2^2 + \lambda\|Z\|_1. \quad (5)$$

Equation (5) can be minimized using an iterative scheme that minimizes D_c and Z alternatively. For details see [15, 14, 10, 29]. Training images are grouped into three categories that best describe their object boundaries: *long hair*, *short hair*, and *sharp boundary*. Figure 2 shows examples of the trained dictionaries.

3.3. Matte reconstruction from dictionary

Our next step is to reconstruct a high-quality alpha matte given a low-quality alpha matte input. For each 8×8 block in the input alpha matte, we estimate the sparse coefficients by minimizing:

$$\phi^* = \arg\min_{\phi} \|D_l \phi - y\|_2^2 + \lambda\|\phi\|_1, \qquad (6)$$

where D_l is the dictionary learned from Equation (5). Equation (6) is solved using linear regression regularized with l_1-norm on the coefficients [23].

In order to guarantee compatibility between neighboring blocks, we follow [29] and use an overlapping window in the reconstructed high-quality alpha matte to constrain the coefficients estimation in Equation (6). This is formulated as:

$$\phi^* = \arg\min_{\phi} \|\tilde{D}\phi - \tilde{y}\|_2^2 + \lambda\|\phi\|_1, \qquad (7)$$

where

$$\tilde{D} = \begin{bmatrix} D_l \\ PD_h \end{bmatrix}, \quad \tilde{y} = \begin{bmatrix} y \\ w \end{bmatrix}, \qquad (8)$$

where P is a matrix that extracts the overlapping region between the current patch and the previous patches, and w is the alpha values of the previously reconstructed alpha mattes in the overlapping areas. Figure 3 shows the comparisons with and without the neighborhood compatibility constraint.
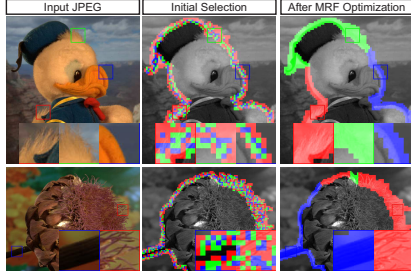
Figure 4. Optimal dictionary selection for different regions along matting boundary. Red: Long hair dictionary; Green: Short hair dictionary; Blue: Sharp boundary dictionary.

3.4. Implementation: dictionary selection

As previously mentioned, we found that using a single dictionary learned from generic images did not produce the best quality results. We instead trained three separate dictionaries to adopt to different structures on the object's boundary. In the following, we describe how to select the optimal dictionary for different boundary structures.

A reconstruction constraint is used to select the optimal dictionary for each 8×8 block as follows. First, a high-quality alpha matte is first reconstructed using all three dictionaries individually. The foreground and background colors are then estimated within each 8×8 block using a color smoothness assumption [12]. This allows us to reconstruct a high-quality image patch using the matting composite equation [16]. The reconstructed image patches are then compressed to the same quality factor of the input JPEG image. Finally, the appropriateness of each dictionary is evaluated by measuring the RMSE between the compressed reconstructed patch and the original input patch. A Markov Random Field(MRF) is used to encode the neighboring compatibility [21, 22]:

$$E(L) = \arg\min_L \sum_{i \in \mathcal{V}} E_D(l_i) + \lambda \sum_{(i,j) \in \mathcal{E}} E_S(l_i, l_j), \quad (9)$$

where $E_D(l_i)$ is the data term which is evaluated by the RMSE between the reconstructed image patch with different dictionary label and the input image:

$$
\begin{aligned}
E_D(l_i = long) &= RMSE_{long}(i)/W, \\
E_D(l_i = short) &= RMSE_{short}(i)/W, \quad (10) \\
E_D(l_i = sharp) &= RMSE_{sharp}(i)/W,
\end{aligned}
$$

where $W = RMSE_{long}(i) + RMSE_{short}(i) + RMSE_{sharp}(i)$ is the normalization factor, and $E_S(l_i, l_j)$ is the neighboring term. We empirically set:

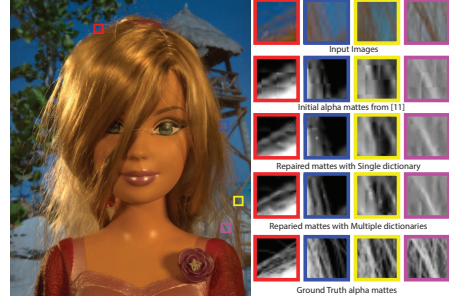|	Long	Short	Sharp
Long	0	0.35	0.8
Short	0.35	0	0.65
Sharp	0.8	0.65	0

$$(11)$$

Figure 5. Comparisons between single dictionary and multiple dictionaries. With our multiple dictionaries approach, hairy structures of matte can be better reconstructed.

Equation (9) can be solved using any standard MRF solver [21]. Figure 4 shows two examples of the dictionary selection results. Interestingly, although we do not explicitly model the statistics of different boundary structures, the reconstruction constraint selects the dictionary that agrees with the local structures of the image patch. Figure 5 shows the comparisons of the reconstructed alpha matte between single dictionary approach and multiple dictionaries approach.

4. Experimental results

Our approach is tested extensively using the following matting algorithms: closed-form matting [12], KNN matting [3], global sampling matting [8], and learning-based matting [30]. The implementation of [12, 3, 30] were provided by the authors. We implemented [8] to produce results of sampling based method. Due to space limitation, only representative results are shown; additional results are provided in the supplemental materials. Training and testing examples are downloaded from *www.alphamatting.com* [18] with ground truth alpha mattes available. Closed-form matting [12] was applied to the uncompressed image and compressed image to produce the training example pairs[3]. Different dictionaries were learnt for different level of compression level. Examples were separated such that there was no overlap between training and testing examples.

Figure 6 shows several results using different quality factors and matting methods. In each example, the initial extracted matte is used as input in our method. Details are then transferred based on our method. For each example, a composite is shown on a green background as well as the extracted matte. Selected zoomed-in regions are also shown, with the original result on top and our result on bottom. Zoomed-in regions show different object boundary structure, including sharp boundaries, and long and short hairs. The initial mattes all suffer from blurriness and blocky artifacts due to the input's JPEG compression. Applying our

---

[3]We found little difference among the results when using other matting methods to prepare the training examples.

	GT02	GT04	GT05	GT07	GT08	GT11	GT13	GT18	GT20	GT21	GT25	GT26
CFM	7.153	10.136	1.285	0.960	5.239	1.803	13.223	1.491	0.448	7.970	10.699	16.801
OUR	7.068	10.322	1.249	0.928	5.475	1.689	12.705	1.372	0.451	7.756	10.608	16.173
KNN	1.030	6.740	1.202	1.310	11.373	3.207	3.350	1.390	0.583	4.284	10.268	12.625
OUR	0.991	7.586	1.135	1.413	12.006	3.109	3.245	1.386	0.706	3.964	9.770	12.006
GSM	0.661	3.757	0.660	0.544	6.076	1.848	3.108	0.825	0.543	3.624	13.198	9.227
OUR	0.542	3.960	0.615	0.505	6.411	1.779	2.785	0.731	0.520	3.383	13.338	8.763
LBM	7.213	9.385	1.259	0.954	5.121	1.777	12.167	1.518	0.451	7.966	10.736	16.673
OUR	7.132	9.528	1.223	0.917	5.393	1.658	11.669	1.394	0.451	7.753	10.654	16.063

GT02 GT04 GT05 GT07 GT08 GT11 GT13 GT18 GT20 GT21 GT25 GT26

Table 1. We compared RMSE between results of matting algorithms and their reconstructed alpha mattes by our algorithm. The unit is $\times 10^{-3}$. CFM, KNN, GSM, and LBM stand for closed-form matting [12], KNN matting [3], global sampling matting [8], and learning-based matting [30], respectively. Results are from images compressed using JPEG quality 9.

method clearly produces better defined boundaries and noticeably reduces the blocky artifacts.

In Figure 7, the results are also compared to those obtained using a popular deblocking software, *DeJPEG* [24] and a popular denoising method, *BM3D* [5]. These are applied in two ways: in one case, we deblocked the JPEG image and then applied matting; in the other case, we applied matting to the JPEG image and applied deblocking to the alpha matte. The alpha mattes extracted from the pre-processed deblocked input images are blurry and have poorly defined boundaries. Interestingly, post-processing the extracted alpha mattes with the deblocking algorithm gives better results. It preserves the shape well and succeeds in removing the blocky compression artifacts to some degree. When compare the results quantitatively, our approach produces the best alpha mattes with the minimum RMSE as compared to the ground truth alpha mattes. Not only does our algorithm eliminates the JPEG compression artifacts, but also results in well defined boundaries of the target object.

We compute the RMSE from the initial alpha mattes and those after applying our detail transfer. Table 1 shows that our algorithm usually yields smaller RMSE than others except for the test cases GT4 and GT8. Upon carefully investigation, we found that both GT4 and GT8 contains very complicated long hair structures. Our results reconstructed the hairs with sharper details but does not fully align with the ground truth alpha mattes. Consequently, our results have higher RMSE due to the misalignment of the reconstructed details. This is not too surprising as RMSE does not always correspond to the best subjective result.

Finally, we apply our algorithm to images captured from cell phone cameras. A Samsung Galaxy S3 and an iPhone4S are chosen due to their popularity. Images are saved in the high quality using their default camera setting. We compare the original file size with the file size of images saved from Photoshop using different quality setting.

Images from both cell phone cameras have compressions quality between quality 10 and quality 11 but closer to quality 10. We therefore built the dictionaries using the quality 10 training examples. Figure 8 shows our results which apply to the alpha mattes extracted from the global sampling matting [8] and closed-form matting [12] with the original files from the cell phone cameras as input. Note that compression artifacts show up in the estimated alpha mattes but they are almost invisible in the original input images. Our approach can successfully refines the alpha mattes with better visual quality.

5. Discussion and conclusion

We have presented a method to refine alpha mattes from JPEG compressed images. While there is a previous work that targets matting for degraded image [13, 17], as far as we are aware, this is the first work to seriously address the problem of compression artifacts on image matting. Our approach is akin to single image learning-based SR approaches that synthesize image detail based on a training set of high-quality and low-quality images. Our method works directly on the alpha mattes using three separate dictionaries to accommodate various boundary structures as well as a back-projection method to select the appropriate dictionary for detail transfer. Our method is able to improve the current state-of-the-art image matting results and preforms better than applying JPEG deblocking to the input or extracted mattes.

Our method focuses on DCT-based JPEG compression due to its popularity. We are interested in extending this scheme to the wavelet-based JPEG2000 compression scheme, however, implementation details such as patch size and patch overlap will need to be empirically evaluated. Another area of future work is to consider expanding the number of dictionaries to accommodate a wider variety of object structures in effort to further improve the results.
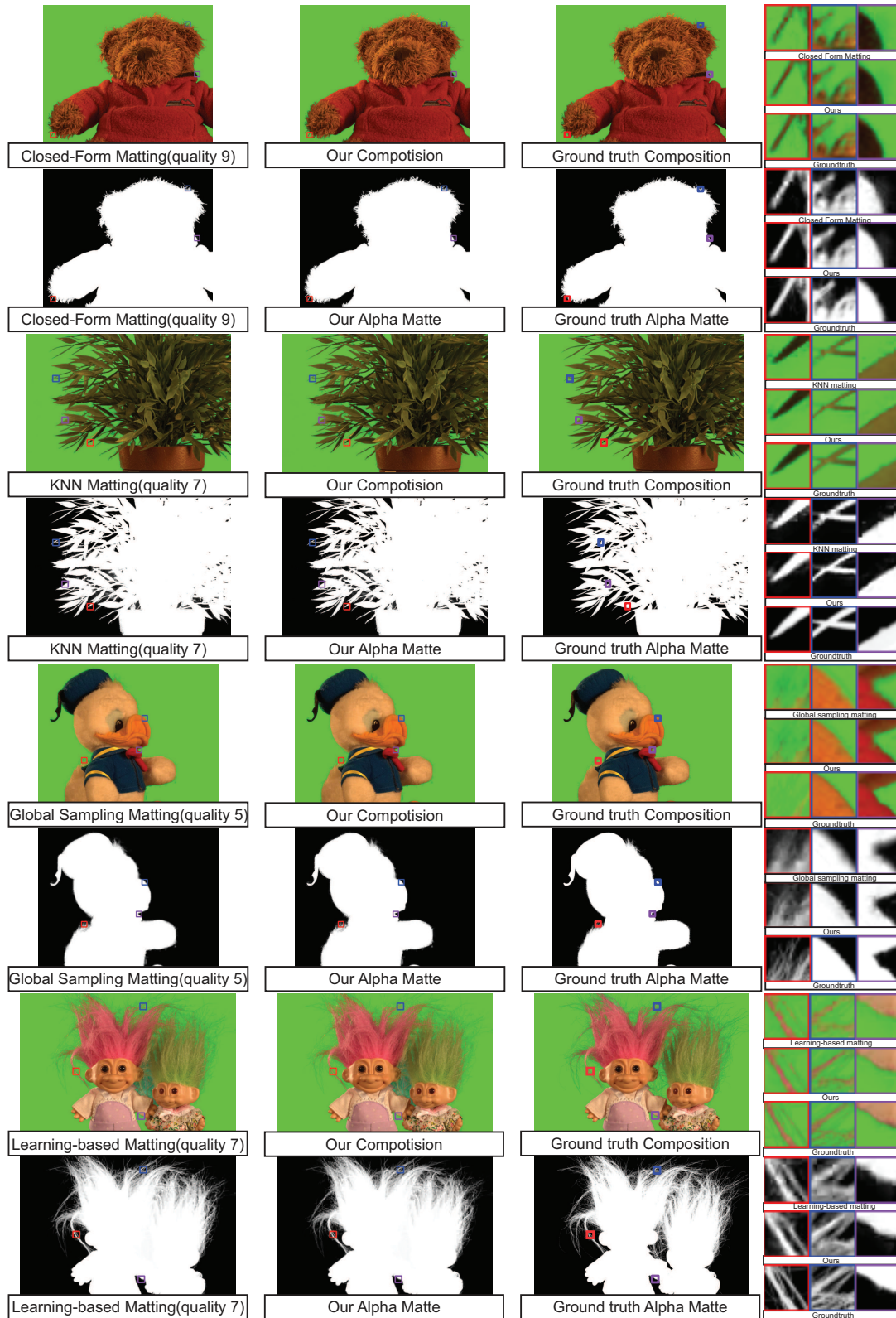
5

Figure 6. The first example is extracted by closed-form matting [12] (JPEG quality 9), the second example is extracted by KNN matting [3] (JPEG quality 7), the third example is extracted by Global sampling matting [8] (quality 5), and the final example is extracted by learning-based matting [30] (JPEG quality 7). In the zoomed-in areas, images on the top were produced using JPEG alpha mattes, and the middles are our reconstructed alpha mattes. The bottoms are the ground truth.
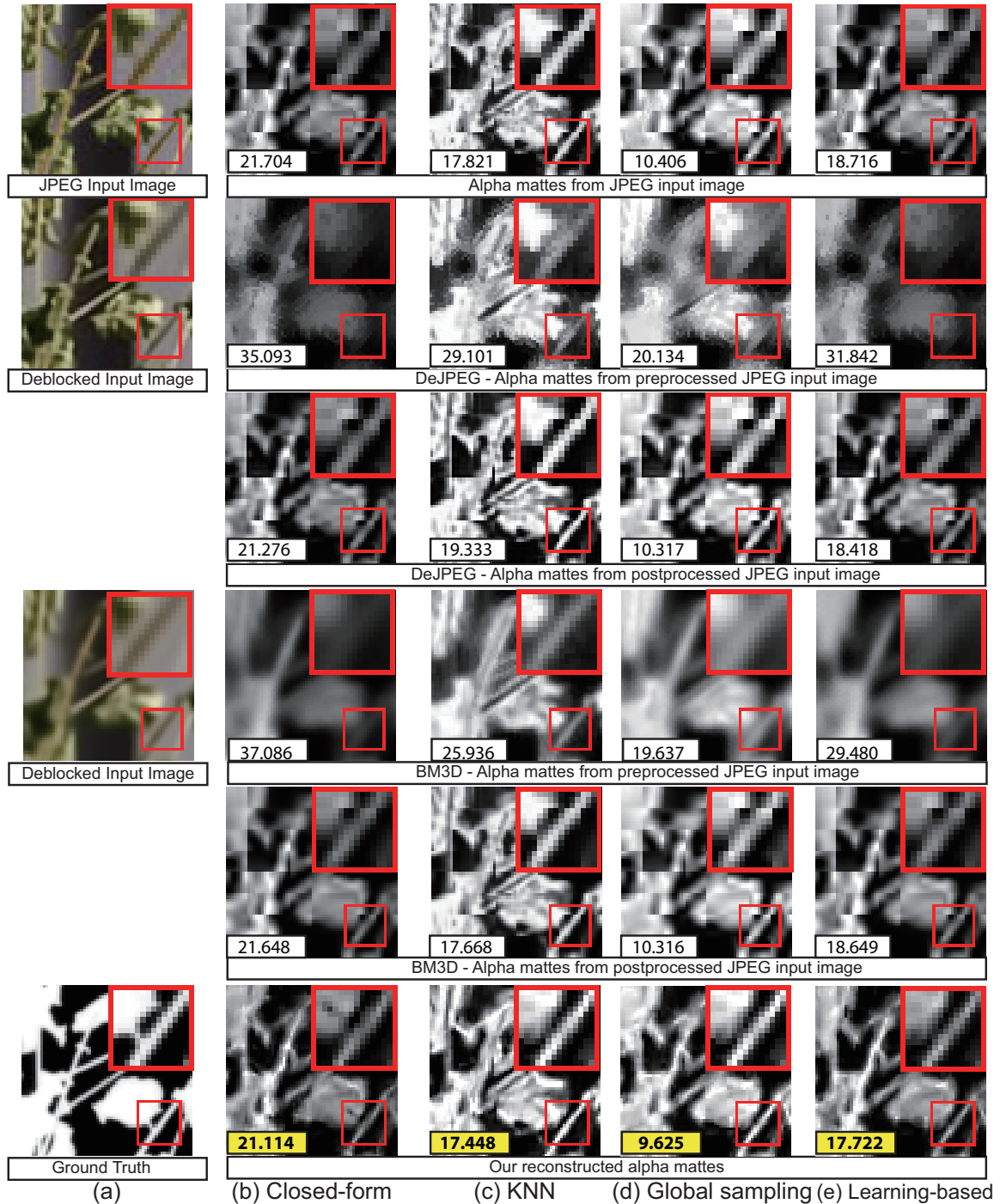
Figure 7. (a) JPEG (quality 7) input images, deblocked input images by DeJPEG [24] and BM3D [5], and ground truth alpha mattes. (b) Closed-form matting [12], (c) KNN matting [3], (d) Global sampling matting [8], (e) learning-based matting [30]. In the example, from the top, alpha mattes of JPEG input images (the first row), alpha mattes of deblocked input images and deblocked alpha mattes as post-processing by DeJPEG (the second and third rows), and by BM3D (the fourth and fifth rows), and our reconstructed alpha mattes (the final row). The RMSE (The unit is $\times 10^{-3}$) are shown in the small boxes. Results with minimum RMSE are highlighted.

6. Acknowledgement

7

Image captured by Galaxy S3 (back camera)　　　　Image captured by iPhone4S (back camera)
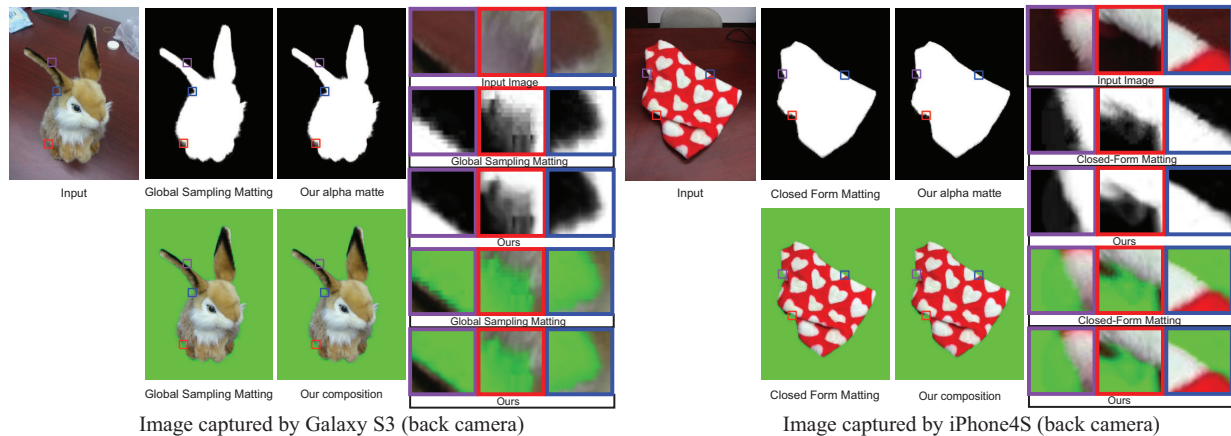
Figure 8. Real world examples from common cell phone cameras. Zoomed-in regions show the input images, estimated alpha mattes, our reconstructed alpha mattes and the corresponding composition respectively.

References

[1] F. Alter, S. Durand, and J. Froment. Deblocking dct-based compressed images with weighted total variation. In *ICASSP'04*, 2004.

[2] A. Averbuch, A. Schclar, and D. Donoho. Deblocking of block-transform compressed images using weighted sums of symmetrically aligned pixels. *IEEE TIP*, 14(2):200 –212, 2005.

[3] Q. Chen, D. Li, and C.-K. Tang. Knn matting. In *CVPR*, 2012.

[4] Y.-Y. Chuang, B. Curless, D. H. Salesin, and R. Szeliski. A bayesian approach to digital matting. In *CVPR*, 2001.

[5] K. Dabov, A. Foi, V. Katkovnik, and K. Egiazarian. Image denoising by sparse 3d transform-domain collaborative filtering. *IEEE TIP*, 16(8):2080–2095, 2007.

[6] A. Foi, V. Katkovnik, and K. Egiazarian. Pointwise shape-adaptive dct for high-quality denoising and deblocking of grayscale and color images. *IEEE TIP*, 16(5):1395–1411, 2007.

[7] E. S. L. Gastal and M. M. Oliveira. Shared sampling for real-time alpha matting. In *Eurographics*, 2010.

[8] K. He, C. Rhemann, C. Rother, X. Tang, and J. Sun. A global sampling method for alpha matting. In *CVPR*, 2011.

[9] Y. Kwon, K. I. Kim, J. Kim, and C. Theobalt. Efficient learning-based image enhancement: Application to super-resolution and compression artifact removal. In *BMVC*, 2012.

[10] H. Lee, A. Battle, R. Raina, and A. Y. Ng. Efficient sparse coding algorithms. In *NIPS*, 2007.

[11] P. Lee and Y. Wu. Nonlocal matting. In *CVPR*, 2011.

[12] A. Levin, D. Lischinski, and Y. Weiss. A closed-form solution to natural image matting. *IEEE TPAMI*, 30:0162–8828, 2008.

[13] H. Lin, Y.-W. Tai, and M. S. Brown. Motion regularization for matting motion blurred objects. *IEEE Trans. on PAMI*, 33(11):2329–2336, 2011.

[14] J. F. Murray and K. Kreutz-Delgado. Learning sparse over-complete codes for images. *J. VLSI Signal Process. Syst.*, 45(1-2):97–110, 2006.

[15] B. A. Olshausen and D. J. Field. Sparse coding with an over-complete basis set: a strategy employed by v1. *Vision research*, 23:3311–3325, 1997.

[16] T. Porter and T. Duff. Compositing digital images. *ACM SIGGRAPH*, 1984.

[17] S. Prabhu and A. Rajagopalan. Natural matting for degraded pictures. *IEEE TIP*, 20(12):3647–3653, 2011.

[18] C. Rhemann, C. Rother, J. Wang, M. Gelautz, P. Kohli, and P. Rott. A perceptually motivated online benchmark for image matting. In *CVPR*, 2009.

[19] J. Sun, J. Jia, C.-K. Tang, and H.-Y. Shum. Poisson matting. *ACM TOG*, 23(3):315–321, 2004.

[20] J. Sun, J. Zhu, and M. F. Tappen. Context-constrained hallucination for image super-resolution. In *CVPR*, 2010.

[21] R. Szeliski, R. Zabih, D. Scharstein, O. Veksler, V. Kolmogorov, A. Agarwala, M. Tappen, and C. Rother. A comparative study of energy minimization methods for markov random fields with smoothness-based priors. *IEEE TPAMI*, 30, 2008.

[22] M. F. Tappen and W. T. Freeman. Comparison of graph cuts with belief propagation for stereo, using identical mrf parameters. In *ICCV*, 2003.

[23] R. Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society, Series B*, 58:267–288, 1994.

[24] L. Topaz Labs. Dejpeg version 4.0.2. http://www.topazlabs.com/.

[25] G. K. Wallace. The jpeg still picture compression standard. *Commun. ACM*, 34(4):30–44, 1991.

[26] J. Wang and M. F. Cohen. Image and video matting: a survey. *Found. Trends. Comput. Graph. Vis.*, 3(2), 2007.

[27] J. Wang and M. F. Cohen. Optimized color sampling for robust matting. In *CVPR*, 2007.

[28] Z. Xiong, M. Orchard, and Y.-Q. Zhang. A deblocking algorithm for jpeg compressed images using overcomplete wavelet representations. *IEEE TCSVT*, 7(2):433 –437, 1997.

[29] J. Yang, J. Wright, T. S. Huang, and Y. Ma. Image super-resolution via sparse representation. *IEEE TIP*, 19(11), 2010.

[30] Y. Zheng and C. Kambhamettu. Learning based digital matting. In *ICCV*, 2009.