Erasing Appearance Preservation in Optimization-based Smoothing

Lvmin Zhang[1,2], Chengze Li[3], Yi JI[2], Chunping Liu[2], and Tien-tsin Wong[3]

[1] Style2Paints Research
[2] Soochow University
[3] the Chinese University of Hong Kong
lvminzhang@acm.org, lvminzhang@siggraph.org, {jiyi,cpliu}@suda.edu.cn

Abstract. Optimization-based image smoothing is routinely formulated as the game between a smoothing energy and an appearance preservation energy. Achieving adequate smoothing is a fundamental goal of these image smoothing algorithms. We show that partially "erasing" the appearance preservation facilitate adequate image smoothing. In this paper, we call this manipulation as Erasing Appearance Preservation (EAP). We conduct an user study, allowing users to indicate the "erasing" positions by drawing scribbles interactively, to verify the correctness and effectiveness of EAP. We observe the characteristics of human-indicated "erasing" positions, and then formulate a simple and effective 0-1 knapsack to automatically synthesize the "erasing" positions. We test our synthesized erasing positions in a majority of image smoothing methods. Experimental results and large-scale perceptual human judgments show that the EAP solution tends to encourage the pattern separation or elimination capabilities of image smoothing algorithms. We further study the performance of the EAP solution in many image decomposition problems to decompose textures, shadows, and the challenging specular reflections. We also present examinations of diversiform image manipulation applications like texture removal, retexturing, intrinsic decomposition, layer extraction, recoloring, material manipulation, *etc*. Due to the widespread applicability of image smoothing, the EAP is also likely to be used in more image editing applications.

Keywords: image smoothing

1 Introduction

Image smoothing is one important foundation of image processing. Denoting the input image as $\boldsymbol{X} \in \mathbb{R}^{H\times W\times C}$ and the output image as $\boldsymbol{Y} \in \mathbb{R}^{H\times W\times C}$ (with $\{H, W, C\}$ being the height, width, and channel), one typical formulation for optimization-based image smoothing could be

$$f(\boldsymbol{X}) = \arg\min_{\boldsymbol{Y}}(\underbrace{\sum_p \rho(\boldsymbol{Y})_p}_{\text{smoothing energy}} + \underbrace{\sum_p L(\boldsymbol{X},\boldsymbol{Y})_p}_{\text{appearance preservation}}) \qquad (1)$$

where p is pixel position. Herein, the smoothing energy $\rho(\cdot)$ depends on different tasks, which could be total variance [28], L0 gradient counting (L0) [33], L1 piece-wise constraint (L1) [5], Relative Total Variance (RTV) [34], Weighted Least Squares (WLS) [32], *etc.* And, the appearance preservation energy $L(\cdot,\cdot)$ depends on different data likelihoods, which could be L2 distance ($||\boldsymbol{X}-\boldsymbol{Y}||_2^2$) [28, 33, 5, 34], Euclidean (or L1) distance ($||\boldsymbol{X}-\boldsymbol{Y}||$) [35], other special Laplacian or Poisson distances [7], and so on.

This paper starts with a key assumption: partially "erasing" the appearance preservation energy facilitate adequate image smoothing. The mathematical form is as follows. We denote all pixel positions as the set $\hbar$. Given an user-indicated set $\mathcal{E}$ of pixel positions to be erased, the remaining pixel positions can be written as the set $\hbar-\mathcal{E}$. We formulate the smoothing problem

$$F(\boldsymbol{X},\mathcal{E})=\arg\min_{\boldsymbol{Y}}(\sum_{p}\rho(\boldsymbol{Y})_p+\sum_{i\in\hbar-\mathcal{E}}L(\boldsymbol{X},\boldsymbol{Y})_i) \tag{2}$$

where our assumption is that users can tune the erasing set $\mathcal{E}$ to achieve adequate and satisfying image smoothing. We conduct an user study, allowing users to draw scribbles to indicate the erasing set $\mathcal{E}$. As shown in Fig. 1, we present the smoothed results from L0, RTV, and L1 algorithms with or without interactive user erasing. Based on this user study, we present discussions as below.

Firstly, such erasing can facilitate more satisfying and adequate smoothing in many practical cases, supported by several evidences: (1) Users can erase their undesired pixels to achieve satisfying smoothing, *e.g.*, in Fig. 1-(a), the user draws scribbles to erase the specular reflections on the car so as to achieve satisfactory car albedo. Fig. 1-(b,h,f,j) are similar cases. (2) Users can preserve and emphasize their desired pixels to facilitate adequate smoothing, *e.g.*, in Fig. 1-(d), the user only traces the leaf texture whereas the tree branches remain untouched and emphasized, so as to only preserve the desired tree branch structure and smooth the leave texture adequately. Fig. 1-(c,e,g,i,k) are similar cases. (3) Given the typicality of these evidences, we are likely to find more practical cases in other color, texture, object, and illumination manipulation scenarios.

Secondly, this erasing is applicable to a variety of applications, supported by several evidences: (1) This erasing can be used in intrinsic image and illumination editing applications, *e.g.*, in Fig. 1-(h), the user draws scribbles to erase the specularity on the sofa, which can ease further illumination decomposition or editing. Fig. 1-(b,f,j) are similar cases. (2) This erasing can be used in texture removal and texture editing applications, *e.g.*, in Fig. 1-(i), the user outlines and eliminate the spider web texture, which can aid in further structure extraction or retexturing. Fig. 1-(c,d,g) are similar cases. (3) Given the typicality of these applications, this erasing is likely to be applied to more applications.

Thirdly, the appearance preservation erasing also brings new challenges: (1) It is labor-intensive and time-consuming to indicate the erasing set $\mathcal{E}$ manually, *e.g.*, in Fig. 1-(g), the user have to draw a large number of scribbles to accurately erase the strawberry texture. Fig. 1-(c,k) are similar cases. (2) Many real-life image smoothing applications require fully automatic processing, *e.g.*, many

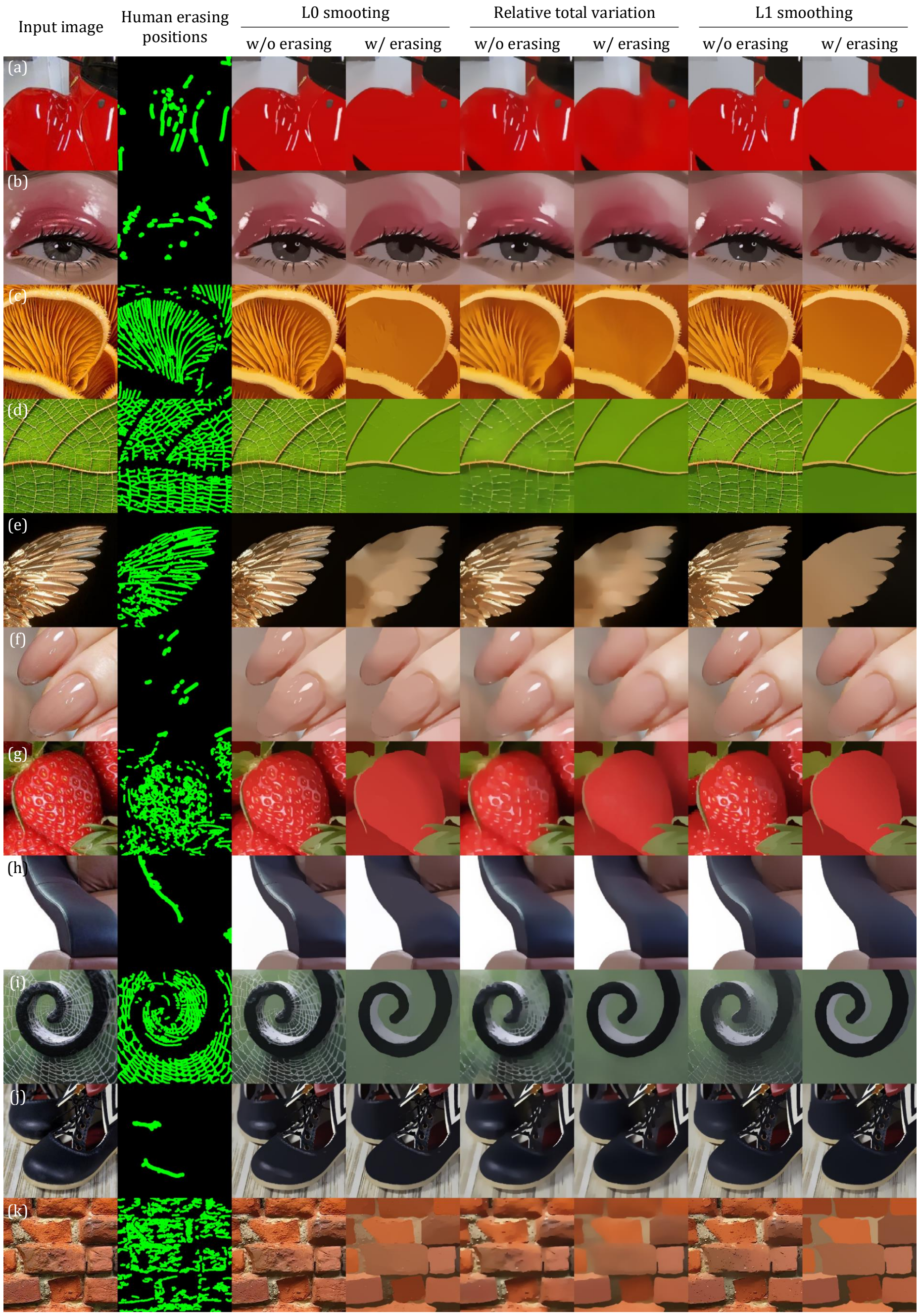


Fig. 1. Evidences for our motivation: partially "erasing" the appearance preservation energy facilitate adequate image smoothing. The user-given erasing pixel position (the set $\mathcal{E}$) are marked with green scribbles.

intrinsic image methods are applicable to video processing, and it is unreasonable to manually annotate videos frame-by-frame. Similar challenges also exist in applications like retexturing, materializing, recoloring, relighting, and so on.

Generally, these results verify that partially "erasing" the appearance preservation encourages adequate and satisfying image smoothing, and achieving such "erasing" can benefit a variety of applications. Nevertheless, the user input is labor-intensive to obtain and is unacceptable in many automatic applications. With the motivation of scalability and applicability, we present the Erasing Appearance Preservation (EAP) problem: solving the erasing set $\mathcal{E}$ in absence of human interaction.

We present a simple and effective solution to the EAP problem. We observe how human indicate their erasing set $\mathcal{E}$ and then use a knapsack model to discover pixels that are likely to be erased by human. Experiments show that our solution can be applied to a variety of downstream tasks, and many state-of-the-art image smoothing applications can benefit from our solution.

Our contributions are: (1) We motivate the Erasing Appearance Preservation (EAP) problem by verifying the assumption that partially "erasing" the appearance preservation energy facilitate adequate image smoothing. (2) We present a simple and effective solution to the EAP problem. We observe how human erase the appearance preservation and formulate a 0-1 knapsack to solve the erasing positions. (3) We show that our solution can be applied to many optimization-based image smoothing methods. Extensive qualitative results, quantitative analysis, and large-scale perceptual human judgments show that this solution facilitates their structure extraction and pattern decomposition capabilities. Furthermore, we study the effectiveness of EAP in various image smoothing applications, *e.g.*, texture decomposition, intrinsic decomposition, color manipulation, material manipulation, *etc.* Additionally, we show results from our EAP solution in several open problems like adequate texture elimination and the challenging specular reflection decomposition.

2 Related Works

Image smoothing. Image smoothing is as an essential component of many image manipulation techniques. Early approaches are filtering-based [31, 18, 36] and recently optimization methods achieve impressive visual effects, *e.g.*, L0-smoothing [33], L1-smoothing [5], and other energy-based methods [34, 11, 1, 30, 27], to name a few. A wide variety of visual effects can be achieve with image smoothing, *e.g.*, visual enhancement [26], intrinsic decomposition [5], texture replacement [6], relighting [30], recoloring [29], stylization [14], *etc.*

Interactive image smoothing and optimization. Many real-world image smoothing applications and image optimization techniques allow users to interactively control the optimization, *e.g.*, matting [10], stylization [9], coloring [22], relighting [30], and so on. The existence of these application also verifies the correctness of EAP. The difference is that these approaches are routinely focused on making use of user inputs (like scribbles), whereas we are focused on finding where to "erase" in absence of human interaction for image smoothing algorithms.

Image inpainting and point-based imaging. Image inpainting methods [20, 25, 12] also "erase" pixels to edit image contents. The difference is that we are aimed at solving the unknown erasing positions, whereas image inpainting is aimed at solving the erased content with known erasing positions. Furthermore, point-based imaging literatures [15, 19] compute pixel points to process images. The different is that these methods are aimed at representing images with detected key points (*e.g.*, sparse control points [15]), whereas we are aimed at finding pixels that need to be "erased".

3 A Solution to the EAP Problem

Our goal is to solve the "erasing" positions for image smoothing algorithms. Observing Fig. 1, we have two discoveries: (1) Human is erasing their "undesired" pixels. For example, if the user wants to remove textures, that user will draw scribbles on the undesired texture, *e.g.*, the tree leaf (Fig. 1-(d)) and the spider web (Fig. 1-(i)). (2) Human is preserving their "desired" pixels. For example, if the user wants to preserve salient object structures, that user will prevent drawing scribbles on the objects' structural constitutes, *e.g.*, the nail surface (Fig. 1-(f)) and the strawberry structural outline (Fig. 1-(g)). Motivated by these two discoveries, we can determine where to erase by estimating desired and undesired pixels.

Estimating how each pixel is "undesired". Image smoothing routinely penalizes task-specific patterns. These patterns can be textures, shadows, specular reflections, noises, and so on. In many specific applications, these penalized patterns can be viewed as the user undesired patterns. For example, in texture removal applications, the smoothing energy is penalizing textures, and simultaneously, those textures are also undesired by the users. Therefore, we estimate how each pixel is penalized, to reflect how each pixel is undesired. During image smoothing (Eq. (2)), the more a pixel is penalized, the more its color changes. We can compute the input-output color change to estimate the penalty:

$$\boldsymbol{V}_p = ||\boldsymbol{X}_p - \boldsymbol{Y}_p||_2^2 \tag{3}$$

where p is pixel position and $\boldsymbol{V}_p$ is the estimated penalty. To aid in the robustness of this estimation, we apply some routinely used image processing strategies to ameliorate the formulation: we compute the CIE RGB-to-Lab transform $\tau(\cdot)$, multiply the Gaussian term $w_{ij} = \exp(||\tau(\boldsymbol{X}_i) - \tau(\boldsymbol{Y}_j)||_2^2/2\sigma^2)$, and focus on the local window l_p at p. The final equation becomes

$$\boldsymbol{v}_p = \sum_{i \in l_p} \sum_{j \in l_p} w_{ij} ||\tau(\boldsymbol{X}_i) - \tau(\boldsymbol{Y}_j)||_2^2 \tag{4}$$

In this way, this equation represents how each pixel is penalized and undesired in our problem. We provide a detailed verification of this equation using Xu's tests [34] in the supplemental material.

Estimating how each pixel is "desired". Image smoothing routinely protects and preserves important image constitutes. For example, texture removal smoothing preserves salient structure, and intrinsic image smoothing preserves object reflectance or albedo. In many specific applications, these preserved patterns can be viewed as user desired patterns. In most cases, after the image smoothing process, those preserved patterns tend to show more salient contours than those penalized patterns. For example, after texture removal, the original textured location tend to be flat, and thus have less salient contours than the preserved structure, which may have many color transitions. Therefore, we compute the salient contours for each pixel in the smoothed image $\boldsymbol{Y}$ to estimate how each pixel is preserved, so as to reflect how each pixel is desired:

$$\boldsymbol{w}_p = \epsilon + \sum_{i \in l_p} \sum_{j \in l_p} ||\tau(\boldsymbol{Y}_i) - \tau(\boldsymbol{Y}_j)||_2^2 \tag{5}$$

where ϵ prevents zero output. When local window l_p is located at salient contours like edges, w_p becomes numerically large. On the contrary, when the colors in l_p are nearly uniform, w_p will decrease to ϵ as no important pattern can be found. In this way, this equation represents how each pixel is preserved and desired in our problem. We provide a detailed verification in the supplemental material.

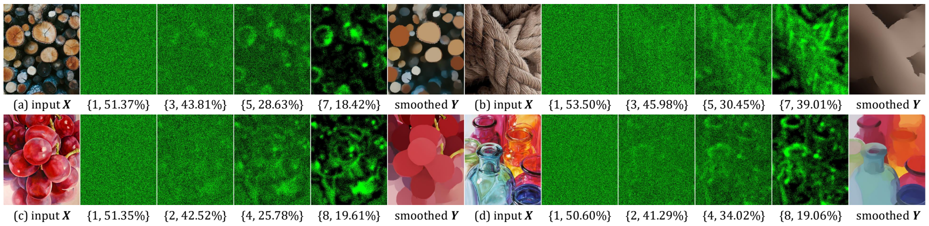
0-1 Knapsack. Now that we have two estimations for each pixel: the $\boldsymbol{v}_p$ estimates how each pixel is "undesired" whereas the $\boldsymbol{w}_p$ estimates how each pixel is "desired". Naturally, we formulate our problem as a 0-1 knapsack: each pixel is a knapsack item, and each item is then chosen whether to put in the knapsack to achieve the largest possible value, while preserving limited total weights. The item value is $\boldsymbol{v}_p$ and the item weight is $\boldsymbol{w}_p$. Then, the knapsack solves as much undesired pixels as possible, while preserving an amount of desired ones. In this way, we achieve a game between erasing undesired pixels and preserving desired pixels.

The 0-1 knapsack is significant and indispensable in our solution, and it is a must to use both estimations of $\boldsymbol{v}_p$ and $\boldsymbol{w}_p$, supported by three evidences: (1) The determination between erasing undesired pixels and preserving desired pixels can only be formulated as a trade-off. In real-world images, the patterns over pixels are complicated and there is no fixed threshold on $\boldsymbol{v}_p$ or $\boldsymbol{w}_p$ to determine what pixel must be erased or preserved. (2) It is hardly possible for users to accurately determine the pixel quantity that should be erased or preserved. Nevertheless, the game of 0-1 knapsack enables a "smart" erasing: given a coarsely indicated knapsack capability, it can adaptively solve how many pixels should be erased. (3) Our later ablative experiments show that, if either one of $\boldsymbol{v}_p$ or $\boldsymbol{w}_p$ is discarded, the performance of our solution will decrease significantly.

To be specific, our maximum knapsack capability is denoted by a manageable scaler $U \in \mathbb{R}$. Considering images vary in scale, for flexibility, we denote $U =$

Algorithm 1: Solver of EAP.

```
Input: Source Image $\boldsymbol{X} \in \mathbb{R}^{H\times W\times 3}$, $u$,$\rho(\cdot)$,$t$;
Output: Smoothed Image $\boldsymbol{Y} \in \mathbb{R}^{H\times W\times 3}$;
Randomly assign 50% pixel positions to $\mathcal{E}$;
for $(i = 0; i < t; i++)$ do
    $\boldsymbol{Y} \leftarrow F(\boldsymbol{X}, \mathcal{E})$;
    Solve $\mathcal{E}$ using 0-1 knapsack;
end
Output $\boldsymbol{Y}$;
```


(a) input $\boldsymbol{X}$ {1, 51.37%} {3, 43.81%} {5, 28.63%} {7, 18.42%} smoothed $\boldsymbol{Y}$ (b) input $\boldsymbol{X}$ {1, 53.50%} {3, 45.98%} {5, 30.45%} {7, 39.01%} smoothed $\boldsymbol{Y}$

(c) input $\boldsymbol{X}$ {1, 51.35%} {2, 42.52%} {4, 25.78%} {8, 19.61%} smoothed $\boldsymbol{Y}$ (d) input $\boldsymbol{X}$ {1, 50.60%} {2, 41.29%} {4, 34.02%} {8, 19.06%} smoothed $\boldsymbol{Y}$

Fig. 2. Visualization of erasing set $\mathcal{E}$ (marked in green) during optimization. Each $\{\cdot,\cdot\}$ indicates iteration step and percentage of $\mathcal{E}$ in all positions $\hbar$.

HWu where H and W are image height and width with $u \in \mathbb{R}$ being an user-given scaler. Finally, this 0-1 knapsack problem of $\{\boldsymbol{v}, \boldsymbol{w}, U\}$ is solved via knapsack dynamic programming. The overall procedures are provided in Algorithm 1. In the supplemental materials, we also include related technical backgrounds of knapsack algorithms and codes of our solver implementations to aid in reproducibility.

Analysis of the EAP solution. Our solution succeeds in solving "undesired" pattern positions. Fig. 2-(a,b) shows experiments for texture removal using relative total variance energy [34]. We can find that EAP is capable of discovering textural positions progressively, *i.e.*, pixels of the tree rings and rope twists are gradually recognized. Fig. 2-(c,d) are for reflectance extraction using L1 intrinsic energy [5], where specular reflections and shadows are efficiently detected. Furthermore, these experiments convert another important message that the EAP framework is adaptive. Given fixed $u = 0.25$ for all examples in Fig. 2, the final percentage of $\mathcal{E}$ varies significantly over different sources and tasks. This result reflects the fact that EAP can achieve adaptive soft constraints for diversified source images and various frontend smoothing energy designed for different tasks. Finally, we also compare our estimated erasing positions to human-indicated erasing points as shown in Fig. 3. We conduct this experiment using both L0 and L1 smoothing. We can see that our solved erasing positions are visually similar to human indication, and our smoothing results are comparable to human performance.

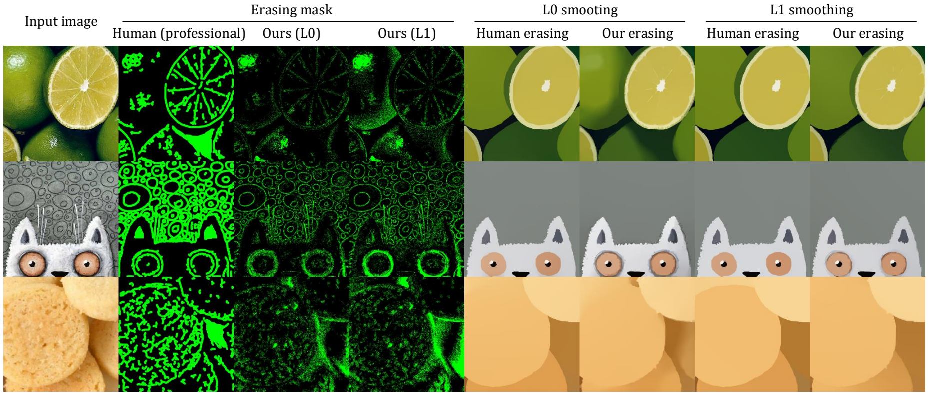


Fig. 3. Comparison of algorithmically solved and manually indicated erasing positions.

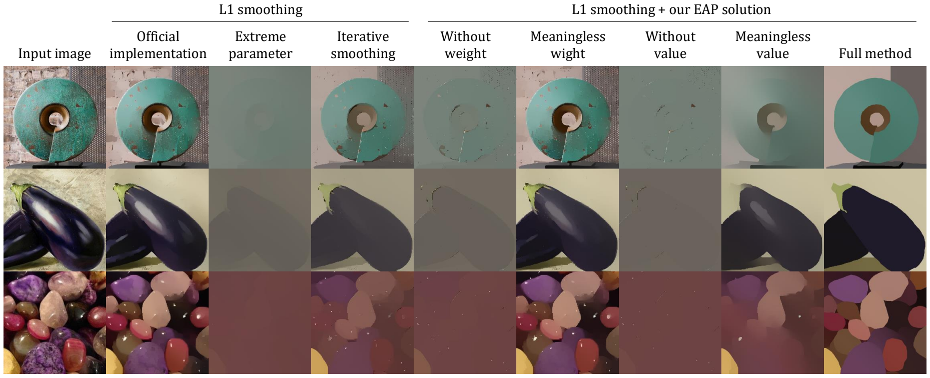


Fig. 4. Ablative study with different alternative configurations for L1 smoothing.

Ablative Study. We perform an ablative study as shown in Fig. 4. More ablative results and implementation details are attached in the supplemental material. We mainly focus on L1 smoothing. We first present the L1 smoothing results from the (1) *official implementation*, and then try some existing strategies to facilitate more throughout smoothing without using EAP: (2) *extreme parameter:* using extreme lambda (10.0) in original L1 smoothing, but without using EAP. We can see that this causes the failure cases where the image constitutes are destroyed. (3) *iterative smoothing:* repeating original L1 smoothing multiple times (10 times, same as the later EAP configuration), but without using EAP. We can see that this causes desaturated and low-contrast artifacts. Then, we try different configurations in our EAP solution: (4) *without weight:* not using knapsack weights $\boldsymbol{w}_p$. Instead, we set a fixed threshold (0.1) to the knapsack values, and all pixels above this threshold are viewed as erasing positions. We

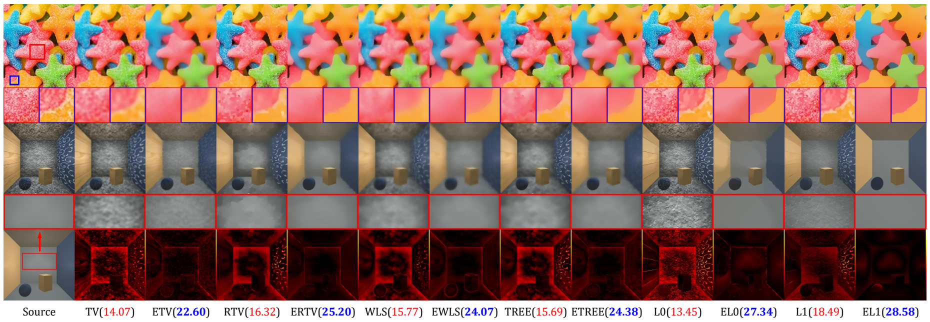


Fig. 5. Texture decomposition. Leftmost is the source (ground truth). We show extracted structures and error maps against the ground truth for each method and report PSNR for each instance. Better scores are marked in blue against red baselines.

can see that this causes the image collapsing to a few colors. (5) *meaningless weight:* replacing all knapsack weights $\boldsymbol{w}_p$ with a constant (1.0). We can see that this causes all undesired patterns being preserved in the final result. (6) *without value:* not using knapsack values $\boldsymbol{v}_p$. Instead, we set a fixed threshold (0.1) to the knapsack weights, and all pixels below this threshold are viewed as erasing positions. We can see that this causes all salient counters being eliminated. (7) *meaningless value:* replacing all knapsack value $\boldsymbol{v}_p$ with a constant (1.0). We can see that this can preserve salient constitutes, but the original structure is corrupted. (8) *full method:* our proposed solution, able to facilitate more adequate smoothing without causing other artifacts.

4 Image Smoothing Energy under EAP

Via changing the $\rho(\cdot)$ and $L(\cdot,\cdot)$ term, various smoothing energies can be flexibly implemented in EAP framework. We denote such strategy by prefix **E**, *i.e.*, total variation (TV) will be named as ETV when EAP is applied. Here, we address several widely-used smoothing energies: **WLS** (1977 [32]) weighted least square; **TV** (1992 [28]) total variation; **L0** (2011 [33]) L0 smoothing; **RTV** (2012 [34]) relative total variation; **TREE** (2014 [2]) optimization-based spanning tree; **L1** (2015 [5]) L1 smoothing; **DL1** (2015 [5]) DPGMM (Dirichlet Process Gaussian Mixture Model) L1 reflectance extraction; When applying EAP approaches to these methods, we obtain: **EWLS**; **ETV**; **EL0**; **ERTV**; **ETREE**; **EL1**; **EDL1**. Unless noticed, we set $u = 0.50$, 3×3 window l_p, $\epsilon = 1.00$, $\sigma = 0.10$, and $t = 10$.

5 Applications

Texture decomposition and retexturing. EAP significantly improves the texture separation capability of optimization-based smoothing methods. Qualitatively,

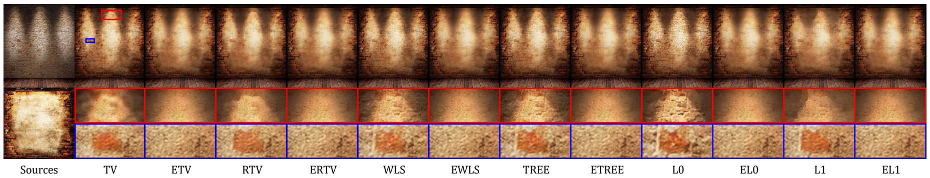


Fig. 6. Texture replacement. Leftmost is the source image and the target texture. Results are presented in following cols.

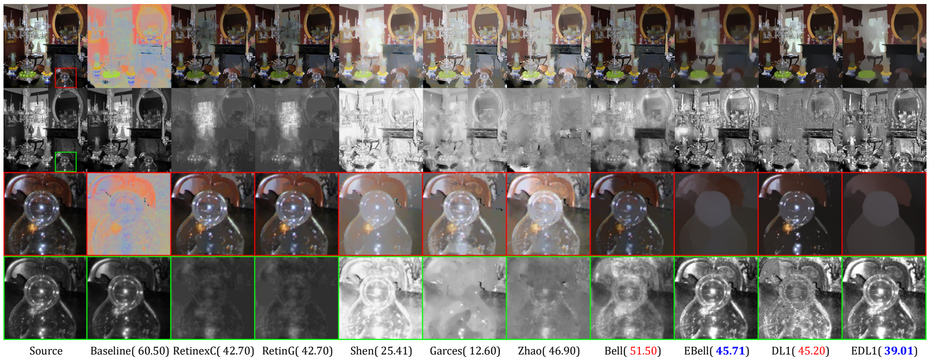


Fig. 7. Intrinsic images. IIW-WHDR(%) (lower is better) is reported. Blue EAP scores are compared against red baselines.

as in the first row of Fig. 5, we compare six appearance-preserving smoothing methods and the respective EAP versions. All EAP-based methods succeed in the complete removal of the challenging dense candy texture, whereas all remaining methods suffer from incomplete texture removal. Quantitatively, as in the second row of Fig. 5, we blend several challenging regular or irregular textures to the Cornell Box to obtain ground truth image pairs with/without textures and evaluate these methods using the PSNR metric. Among all methods, the EAP-based method reports significantly higher PSNR values than others, thus enabling more thorough texture decomposition. More examples are provided in the supplemental material. Fig. 6 shows texture replacement examples. The objective is to remove the brick texture on the wall completely (top-left of Fig. 6) and replace it with the new texture of the uncovered wall (bottom-left of Fig. 6) via removing and then swapping texture. All EAP-based methods managed to remove original bricks and swap texture. In contrast, the remaining methods more or less fail in eliminating the original brick texture and cause maroon-spotted artifacts due to incomplete removal of the original red bricks. This phenomenon further validates that EAP improves pattern separation capability of current image smoothing algorithms.

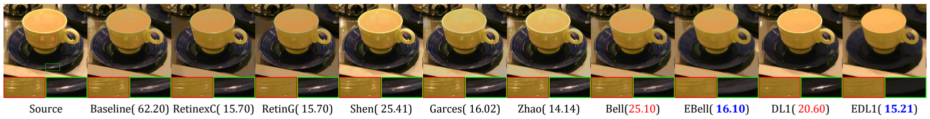


Fig. 8. Specular reflection removal. IIW-WHDR(%) are reported and blue EAP scores are compared to the red baselines.

Intrinsic images and illumination editing. Intrinsic decomposition [3] aims at extracting reflectance and illumination (shading) maps from single image. Interestingly, many optimization-based intrinsic methods are closely related to image smoothing, and their objectives can also be viewed as the appearance preservation terms and smoothing (or penalization) terms. To evaluate the performance of EAP in the intrinsic decomposition task, we use typical Bell [4] and DL1 methods as examples as well as their EAP versions EBell and EDL1 in the experiments. More details of involved algorithms and their EAP implementations are provided in the supplemental materials.

Qualitatively, the EAP framework significantly improves structure abstraction and color separation capability of the involved candidates. We provide a challenging example in Fig. 7 with transparent objects and intensive specular reflections. We can see that EBell and EDL1 succeed in separating specular reflection colors and object colors, whereas other methods fail in telling them apart.

Quantitatively, we test on several routine intrinsic metrics: Intrinsic Image in the Wild [4] (IIW) and Shadow Annotation in the Wild [21] (enhanced SAW in [23]). In IIW/SAW tests, humans are invited to annotate pixels based on whether reflectance/shadow colors are similar or not (having offsets), and the human judgments are then compared with the estimations of tested methods. We report scores of optimization-based methods [21, 37, 16, 17, 30, 4, 5] in Table. 1. We can see that the EAP framework can significantly improve the quantitative performance of Bell and DL1 with the controllable parameter u (the 0-1 Knapsack capability). Given the typicality of Bell and DL1, the EAP framework is likely to improve more optimization-based intrinsic decomposition methods.

On the other hand, although recent IIW/SAW benchmarks are dominated by deep learning methods [13, 24, 38, 23] (CGIntrinsics [23] 99.11% AP and GloSH [38] 15.2% WHDR), optimization methods are still widely used methods. IIW/SAW only evaluates the quantitative accuracy of the reflectance and shadows but ignores the real-life usability of decomposed layers in image editing tasks like relighting, retexturing, rematerializing, *etc.* In Fig. 7, it is obvious that Garces [16] is not suitable for image editing despite its best IIW-WHDR score. Another example of specular reflection removal in Fig. 8 shows that only EAP-based methods manage to eliminate specular reflections, but in the meanwhile, the unusable Zhao [37] reports the best IIW score.

To evaluate image editing performance of intrinsic methods, we introduce real humans to help with the evaluation. We apply state of the art deep learn-

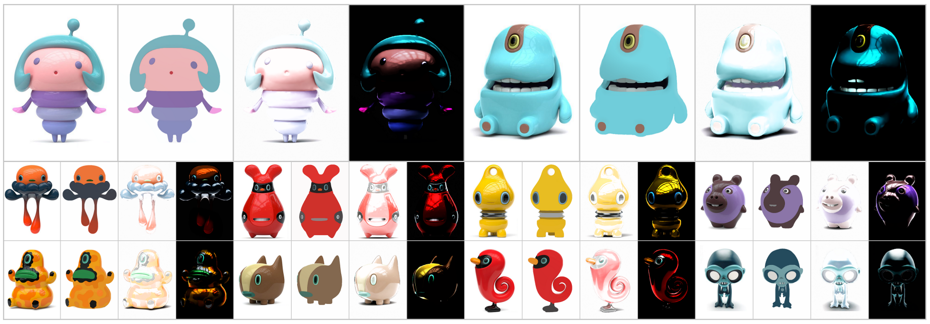


Fig. 9. Qualitative results on image decomposition. We visualize decomposed layers using our EAP-based method EDL1.

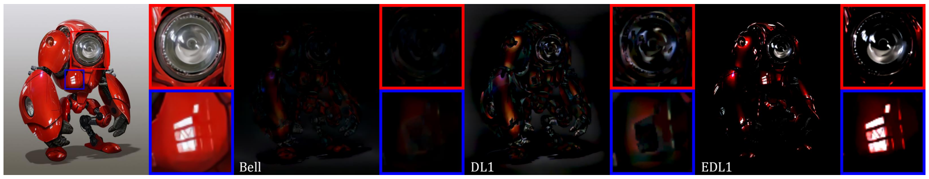


Fig. 10. Comparisons of specular reflections extracted for image decomposition task using different smoothing algorithms.

ing methods [38, 23], optimization methods [4, 5], and their EAP versions to decompose 100 scenes into layers, and then apply their layers to two typical illumination editing tasks: shadow enhancement and specular reflection removal. In this way, we obtain 600 results with enhanced shadows and 600 results with specular reflections eliminated. We employ Amazon Mechanical Turk (AMT) to rank the visual quality of these results and report the obtained ranking in Table. 2. Interestingly, traditional optimization algorithms outperform learning-based models in both tasks, and EAP-based optimization outperforms standard optimizations. We provide 1200 raw results and 1200 raw AMT ranks in the supplement. This strong evidence shows that EAP achieves beyond state of the art performance in intrinsic decomposition based image manipulation.

Advanced illumination decomposition. Both natural images and artistic illustrations may contain complex lighting conditions, and the lighting in digital paintings could be arbitrarily drawn for aesthetic purposes. Advanced decomposition method [8] supports extracting multiple illumination layers and assign specular reflections into separated layers when reflectance maps are given. Notably, this technique alone is not functional for image decomposition, and it requires users to input the smoothed image structure. In this task, we use image smoothing methods to provide such smoothed image structures for layer

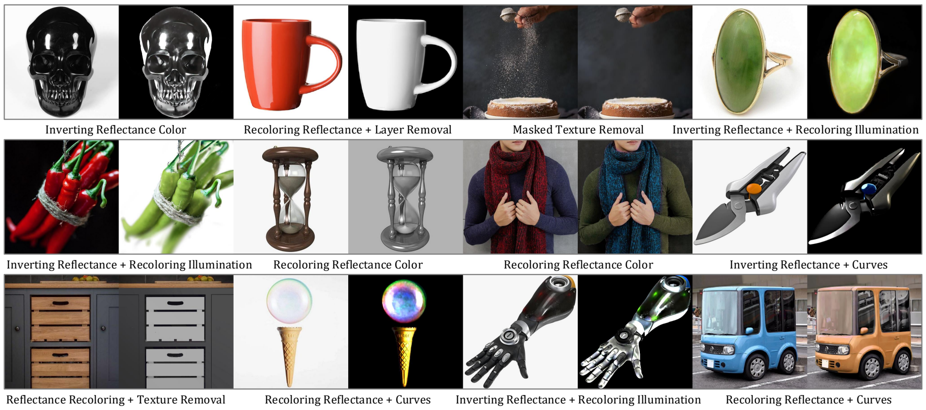

Inverting Reflectance Color
Recoloring Reflectance + Layer Removal
Masked Texture Removal
Inverting Reflectance + Recoloring Illumination
Inverting Reflectance + Recoloring Illumination
Recoloring Reflectance Color
Recoloring Reflectance Color
Inverting Reflectance + Curves
Reflectance Recoloring + Texture Removal
Recoloring Reflectance + Curves
Inverting Reflectance + Recoloring Illumination
Recoloring Reflectance + Curves

Fig. 11. EAP-based image manipulation. We show various image editing use cases with EDL1 layer decomposition method.

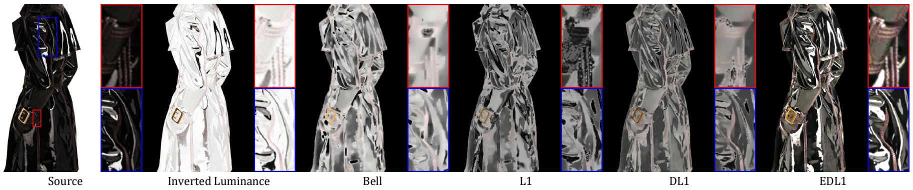


Fig. 12. Comparison of black-to-white rematerialization via inverting reflectance colors extracted with or without EAP.

decomposition. We compare DL1, Bell, and our revamped EDL1. We provide qualitative results in Fig. 9. These results show that our method is capable of decomposing challenging hard shadows and specular reflections simultaneously. On the contrary, as in Fig. 10, previous methods are almost not likely to achieve visually satisfactory decomposition because appearance preservation prevents them from completely separating desired patterns.

Decomposition-based image manipulation. A wide variety of intrinsic decomposition based image manipulation applications can be greatly benefited from the EAP framework. Fig. 11 shows qualitative results of use cases like reflectance/illumination recoloring, retexturing, curve tuning, *etc*. Related technical backgrounds are provided in the supplementary materials.

We discuss the significance of EAP using recoloring examples. One of the most difficult case in recoloring is to invert the scene reflectance while preserving the original illumination. Specular materials and complicated lighting conditions can make this case even more challenging. Fig. 12 shows an example to turn the black coat into white while preserving the specular latex material, by inverting

Table 1. IIW/SAW test for optimization-based intrinsic decomposition methods. WHDR scores are reported for IIW test evaluating reflectance quality whereas AP scores are reported for SAW test evaluating shadow quality. Arrows indicate that lower (↓) or higher (↑) is better. Top 1 (or 2) score is marked in blue (or red).

Method	C-R [21]	Zhao [37]	Garces [16]	RetinexC [17]	Shen [30]	Bell [4]	DL1 [5]
WHDR% ↓	36.6	23.8	24.8	26.9	31.8	20.6	17.7
AP% ↑	75.5	89.72	92.39	85.26	91.4	92.18	93.57

Method	EAP + Bell [4]					EAP + DL1 [5]				
u	0.01	0.05	0.10	0.25	0.50	0.01	0.05	0.10	0.25	0.50
WHDR% ↓	20.4	20.9	19.9	25.3	44.7	17.9	**16.2**	**16.9**	20.4	52.5
AP% ↑	92.97	92.17	**94.43**	86.25	77.52	93.69	94.17	**95.21**	88.41	65.47

Table 2. Amazon Mechanical Turk (AMT) average human ranking on image manipulation tasks of Shadow Enhancement (SE) and Specular Reflection Removal (SRR) over intrinsic decomposition methods. Arrows indicate that lower (↓) is better. Top 1 (or 2) score is marked in blue (or red).

Method	CGIntrinsics [23]	GLoSH [38]	Bell [4]	EAP + Bell [4]	DL1 [5]	EAP + DL1 [5]
SE ↓	5.70 ± 0.46	5.30 ± 0.46	3.55 ± 0.50	**1.71 ± 0.45**	3.45 ± 0.50	**1.29 ± 0.45**
SRR ↓	5.22 ± 0.41	5.77 ± 0.44	3.54 ± 0.52	**1.66 ± 0.47**	3.47 ± 0.50	**1.34 ± 0.47**

the reflectance colors extracted using Bell, L1, DL1, and EAP-based EDL1 method. We also include a naive luminance inverting method. Because the appearance preservation terms of previous methods tend to prevent thorough specular decomposition, their results suffer from visible hole-like artifacts and non-saturated artifacts. On the contrary, EDL1 succeeded in producing satisfactory results because the EAP technique greatly enhances the pattern separation capability, resulting in more reliable decomposition.

6 Conclusion

This paper presents the Erasing Appearance Preservation (EAP) problem that partially "erase" the appearance preservation energy to facilitate adequate image smoothing. We conduct an user study to verify the effectiveness and correctness of EAP. We also presents a method to synthesize the erasing positions automatically. Qualitative, quantitative, and perceptual evidences show that EAP can facilitate the pattern separation and structure extraction capabilities of a majority of optimization-based image smoothing algorithms. We extensively study the performance of the EAP framework in smoothing-based image decomposition problems for textures, shadows, and the challenging specular reflections. Diversiform image manipulation applications like texture decomposition, intrinsic decomposition, color/material manipulation, and many others are also studied with the EAP solution. Due to the widespread applicability of image smoothing in visual editing, the EAP method is likely to benefit more applications.

References

1. Bach, F., Jenatton, R., Mairal, J., Obozinski, G., et al: Structured sparsity through convex optimization. Statist (2012)

2. Bao, L., Song, Y., Yang, Q., Yuan, H., Wang, G.: Tree filtering efficient structurepreserving smoothing with a minimum spanning tree. IEEE Transactions on Image Processing (2014)
3. Barrow, H.G., Tenenbaum, J.M.: Recovering intrinsic scene characteristics from images. In: Hanson, A., Riseman, E. (eds.) Computer Vision Systems. pp. 3–26. Academic Press (1978)
4. Bell, S., Bala, K., Snavely, N.: Intrinsic images in the wild. ACM Transactions on Graphics **33**(4) (2014)
5. Bi, S., Han, X., Yu, Y.: An l1 image transform for edgepreserving smoothing and scenelevel intrinsic decomposition. ACM Transactions on Graphics (2015)
6. Bousseau, A., Paris, S., Durand, F.: User-assisted intrinsic images. ACM Transactions on Graphics (2009)
7. Buzug, M, T.: Computed tomography. Springer (2008)
8. Carroll, R., Ramamoorthi, R., Agrawala, M.: Illumination decomposition for material recoloring with consistent interreflections. ACM Transactions on Graphics (2011)
9. Champandard, A.J.: Semantic style transfer and turning two-bit doodles into fine artworks. CoRR **abs/1603.01768** (2016)
10. Chen, Q., Li, D., Tang, C.K.: Knn matting. Pattern Analysis and Machine Intelligence, IEEE Transactions on **35**(9), 2175–2188 (Sept 2013)
11. Cho, H., Lee, H., Kang, H., Lee, S.: Bilateral texture filtering. ACM Transactions on Graphics (2014)
12. Criminisi, A., Perez, P., Toyama, K.: Region filling and object removal by exemplar-based image inpainting. IEEE Transactions on Image Processing **13**(9), 1200–1212 (Sep 2004)
13. Fan, Q., Yang, J., Hua, G., Chen, B., Wipf, D.: Revisiting deep intrinsic image decompositions. CVPR (2018)
14. FAN, Q., YANG, J., WIPF, D., CHEN, B., TONG, X.: Image smoothing via unsupervised learning. ACM Transactions on Graphics (2018)
15. Galić, I., Weickert, J., Welk, M., Bruhn, A., Belyaev, A., Seidel, H.P.: Image compression with anisotropic diffusion. Journal of Mathematical Imaging and Vision **31**(2-3), 255–269 (apr 2008)
16. Garces, E., Munoz, A., Lopez-Moreno, J., Gutierrez, D.: Intrinsic images by clustering. Computer Graphics Forum (2012)
17. Grosse, R., Johnson, M.K., Adelson, E.H., Freeman, W.T.: Ground truth dataset and baseline evaluations for intrinsic image algorithms. ICCV (2019)
18. He, K., Sun, J., Tang, X.: Guied image filtering. TPAMI (2013)
19. Hoeltgen, L., Setzer, S., Weickert, J.: An optimal control approach to find sparse data for laplace interpolation. In: Lecture Notes in Computer Science, pp. 151–164. Springer Berlin Heidelberg (2013)
20. Iizuka, S., Simo-Serra, E., Ishikawa, H.: Globally and Locally Consistent Image Completion. ACM Transactions on Graphics (Proc. of SIGGRAPH 2017) **36**(4), 107:1–107:14 (2017)
21. Kovacs, B., Bell, S., Snavely, N., Bala, K.: Shading annotations in the wild. CVPR (2017)
22. Levin, A., Lischinski, D., Weiss, Y.: Colorization using optimization. In: ACM SIGGRAPH 2004 Papers. p. 689694. SIGGRAPH 04, Association for Computing Machinery, New York, NY, USA (2004)
23. Li, Z., Snavely, N.: Cgintrinsics: Better intrinsic image decomposition through physically-based rendering. ECCV (2018)

24. Li, Z., Snavely, N.: Learning intrinsic image decomposition from watching the world. CVPR (2018)
25. Liu, G., Reda, F.A., Shih, K.J., Wang, T.C., Tao, A., Catanzaro, B.: Image inpainting for irregular holes using partial convolutions. In: The European Conference on Computer Vision (ECCV) (2018)
26. Min, D., Choi, S., Lu, J., Ham, B., Sohn, K., Do, M.N.: Fast global image smoothing based on weighted least squares. IEEE Transactions on Image Processing (2014)
27. Prasath, V.S., Vorotnikov, D., Pelapur, R., Jose, S., Seetharaman, G., Palaniappan, K.: Multiscale tikhonovtotal variation image restoration using spatially varying edge coherence exponent. IEEE Transactions on Image Processing (2015)
28. RUDIN, L, OSHER, S, , FATEMI, E: Nonlinear total variation based noise removal algorithms. Physica D: Nonlinear Phenomena (1992)
29. Serra, M., Penacchio, O., Benavente, R., Vanrell, M.: Names and shades of color for intrinsic image estimation. CVPR (2012)
30. Shen, J., Yang, X., Jia, Y., Li, X.: Intrinsic images using optimization. CVPR (2011)
31. Tomasi, C.: Bilateral filtering for gray and color images. ICCV (1998)
32. Welsch, P.W.H.R.E.: Robust regression using iteratively reweighted leastsquares. Communications in Statisticstheory and Methods (1977)
33. Xu, L., Lu, C., Xu, Y., Jia, J.: Image smoothing via l0 gradient minimization. ACM Transactions on Graphics (2011)
34. Xu, L., Yan, Q., Xia, Y., Jia, J.: Structure extraction from texture via relative total variation. ACM Transactions on Graphics (2012)
35. Yang, J., Zhang, Y., Yin, W.: An efficient TVL1 algorithm for deblurring multichannel images corrupted by impulsive noise. SIAM Journal on Scientific Computing **31**(4), 2842–2865 (jan 2009)
36. Yin, H., Gong, Y., Qiu, G.: Side window filtering. CVPR (2019)
37. Zhao, Q., Tan, P., Dai, Q., Shen, L., Wu, E., Lin, S.: A closed-form solution to retinex with nonlocal texture constraints. TPAMI (2012)
38. Zhou, H., Yu, X., Jacobs, D.W.: Glosh: Global-local spherical harmonics for intrinsic image decomposition. ICCV (2019)