Noname manuscript No.
(will be inserted by the editor)

Parametric Meta-filter Modeling from a Single Example Pair

Shi-Sheng Huang · Guo-Xin Zhang · Yu-Kun Lai · Johannes Kopf · Daniel Cohen-Or · Shi-Min Hu

Received: date / Accepted: date

Abstract We present a method for learning a *meta-filter* from an example pair comprising an original image A and its filtered version A' using an unknown image filter. A meta-filter is a parametric model, consisting of a spatially varying linear combination of simple basis filters. We introduce a technique for learning the parameters of the meta-filter f such that it approximates the effects of the unknown filter, i.e., $f(A)$ approximates A'. The meta-filter can be transferred to novel input images, and its parametric representation enables intuitive tuning of its parameters to achieve controlled variations. We show that our technique successfully learns and models meta-filters that approximate a large variety of common image filters with high accuracy both visually and quantitatively.

Keywords image filters, filter space, sparsity, learning, transfer

1 Introduction

Image filtering is one of the most fundamental operations in computer graphics. It is the key building block in many graphics algorithms as well as an important tool in many image editing and image enhancement applications. In this paper we examine the problem of learning an image filter from a pair of example images, transferring it to new inputs, and intuitively tuning its parameters. Learning filters from examples is an important task, because the exact functioning principles behind many image filters in commercial software are undisclosed. Even if the algorithmic details are known, source code is often not available and the filter might be difficult to re-implement from scratch. Moreover, applying image filters often involves manual tuning of (spatially varying) parameters, which might require expert knowledge and can be time consuming.

The task of learning an image filter from an example pair can be challenging since in its widest sense image filtering is a very general concept. Filters are implemented using a variety of techniques, including iterative, recursive, and data-driven approaches. Often several filters are applied in sequence to achieve a desired compound effect. Even some manual operations, such as retouching skin blemishes in portraits can be considered as a kind of image filter.

To alleviate this task we introduce the *parametric meta-filter*. The meta-filter is a linear combination of elementary basis filters from small filter bank. Given an example pair comprising an original image A and its filtered version A' (Figure 1a), our method learns the spatially varying combination weights of the meta-filter f, so that $f(A) \approx A'$ (Figure 1b). The learnt meta-filter can then be applied to novel input images, $B \rightarrow f(B)$ (Figure 1c). Since our basis filters are parametric we can intuitively tune their parameters to achieve controlled variations (Figure 1d).

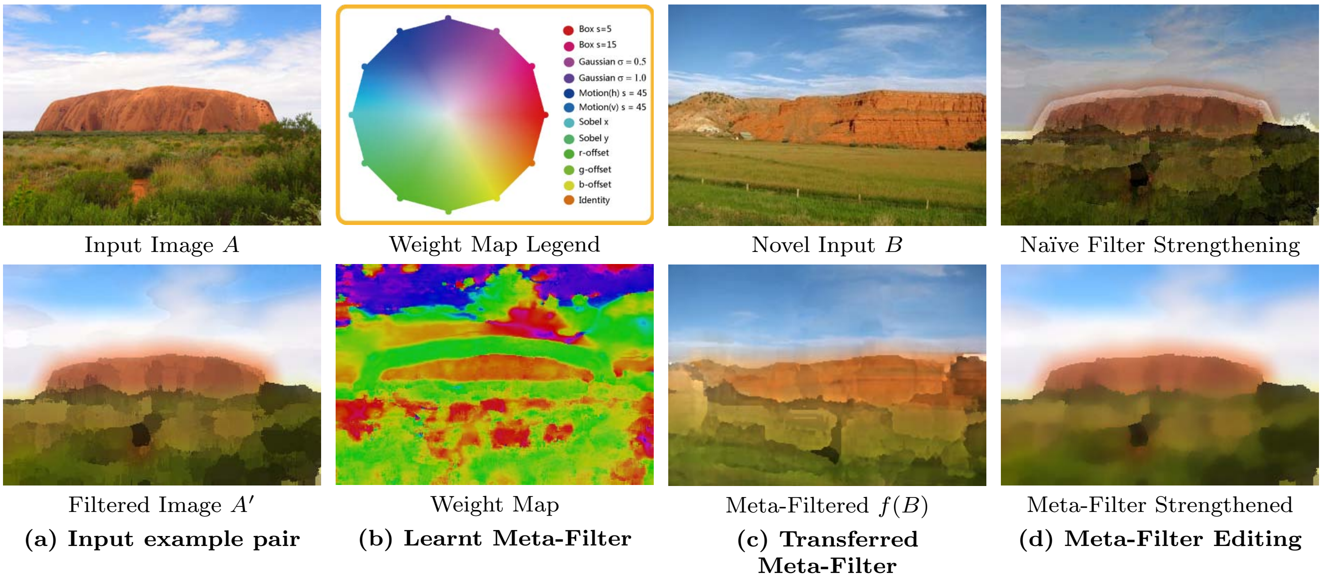
The Image Analogies algorithm [1] attempts a similar problem using a non-parametric texture synthesis algorithm. As such, it works well for "texture-like" ef-

Shi-Sheng Huang, Guo-Xin Zhang and Shi-Min Hu
Tsinghua University, Beijing
E-mail: shimin@tsinghua.edu.cn

Yu-Kun Lai
Cardiff University
E-mail: YuKun.Lai@cs.cardiff.ac.uk

Johannes Kopf
Microsoft Research, Redmond
E-mail: kopf@microsoft.com

Daniel Cohen-Or
Tel Aviv University
E-mail: dcor@tau.ac.il



Input Image A Weight Map Legend Novel Input B Naïve Filter Strengthening

Filtered Image A' Weight Map Meta-Filtered $f(B)$ Meta-Filter Strengthened

(a) Input example pair **(b) Learnt Meta-Filter** **(c) Transferred Meta-Filter** **(d) Meta-Filter Editing**

Fig. 1 Given an example pair comprising an input image and a filtered version (a), our method learns the parameters of a *meta-filter* that approximates the latent filter (b). The meta-filter can be transferred to novel input images (c). Its parametric representation enables intuitive parameter tuning to achieve controlled filtering variations (d).

fects (e.g., painterly filters), however, we show that it does not perform as successfully on many other typical image filter categories. In addition, the non-parametric nature of the algorithm makes it difficult to tune filter parameters to achieve variations. Our parametric method, in contrast, is applicable to a wider range of image filters, including artistic filters (e.g., from the Photoshop Filter Gallery), tone adjustment, color transfer, curves, and some manual image enhancement tasks such as skin smoothing.

We tested our method on more than 50 examples from before mentioned categories. We show that our learnt meta-filters approximate the latent filter on the given exemplar pairs near perfectly, and also transfer well to novel input images. We evaluate our results numerically using common image similarity metrics, as well as perceptually through a user study. In addition to the results shown in the paper, we include further results and more extensive comparisons and evaluations in the supplementary material.

2 Related Work

Filter Estimation. An ongoing area of research in the field of image restoration is filter estimation, where an original image is sought to be recovered from a given "filtered" image. The most important instance of this problem is removing blur from images. Here, the filters are typically modelled as convolutions with blur kernels, and their inversion is referred to as deconvolution [11]. When the filter is unknown, the result is a blind deconvolution problem. These techniques use some priors and regularization to constrain the solution and restrict the search space [5, 9, 10, 15–19, 30]. Most filter estimation methods assume that a homogenous filter is applied to the whole image (or a sufficiently large region). The recent work of Joshi et al. [17] estimates the point-spread functions in local windows and, thus, allows recovering spatially varying blur kernels. Li et al [14] apply a nonlinear filter bank to the neighborhood of each pixel. Outputs of these spatially-varying filters are merged using global optimization, which benefit a set of applications. The problem we address in this paper is different from image restoration in two important ways: First, we have no knowledge of the nature of the unknown filter; we are dealing with general and spatially varying filters. Second, we do have the original image available as part of the input.

Learning from Pairs. Our work is strongly related to various transfer techniques. These techniques often work by taking one or more example pairs, where each consists of an image A and a modified version A'. Then for a given input image B, the aim is to produce B' that somehow mimics the transform from A to A'. Image analogies [1, 26] is a well-known technique that uses non-parametric texture synthesis. By using appropriate example pairs, a large variety of effects can be achieved, from simple smoothing to sophisticated artistic effects. Our approach explicitly learns and models the filter from example pairs, and avoids various artifacts associated with a direct patch work in image space. As mentioned earlier, having a parametric model offers control and efficiency.

There are more techniques that learn from pairs or examples. For example, the work by Kang et al. [20] and Bychkovsky et al. [23] consider learning global tone mapping from a training set using machine learning techniques, the work of Wang et al. [2] considers example-based learning of color and tone mapping, Ling et al. [27] introduce an adaptive tone-preserved method for image detail enhancement, and Huang et al. [8,28] consider example-based contrast enhancement by gradient mapping. By analyzing the relation between the color theme and affective word, Wang et al. [24,25] introduce an example based affective adjustment method with a single word. Unlike these techniques, our method is generic and learns a more general filter structure.

Our work is also related to the work of Berthouzoz et al. [7], who introduce a framework for transferring photo manipulation macros to new images. Multiple training demonstrations are used to learn the relationship between the image features, and macro parameters of selections, brush strokes and image processing operations, using image labeling and machine learning. While having similar goals to our work, their method requires Photoshop macros to be recorded. Our method fully automatically learns the filter from a single pair of input images.

Linear Combination of Filters. In this work we model a compound filter by a linear combination of basis filters. Sahba and Tizhoosh [6] also use a linear combination of four filters to produce an improved denoising filter for a given input image using a reinforced learning algorithm. Their algorithm is only suitable for a specific type of filter, which cannot be spatially varying. Given an additional guide image, which can be identical to the input image, He et al. [12] construct a linear combination of local mappings within windows of the guided image. Simple linear mappings are derived within each overlapping window such that when applied to the guided image, the results approximate the input image. In our work, we consider locally linear combinations of general filters that approximate a large variety of many different composite filters.

3 Overview

We define the parametric meta-filter as a linear combination of elementary basis filters f_k:

$$f(p) = \sum_k w_k(p)\, f_k(p), \tag{1}$$

where p is a pixel coordinate. To facilitate the operation we precompute the basis filters, i.e., f_k is an image that contains the result of applying the basis filter to the input image A. The spatially varying weights $w_k(p)$ comprise the parameters of the meta-filter. Note that we do not restrict the weights at a pixel to be a partition of unity, i.e., $\sum w_k$ is not required to be 1. This flexibility is essential since the original and filtered images may differ in contrast, brightness, or tone.

Our basis filter bank contains instances from a few families of filters, in particular, Gaussian, Box, Motion Blur (i.e., directional Gaussians), Sobel edge, Color Offset, and Identity filters. The Motion Blur and Sobel edge filters include horizontal and vertical variants. Since most basis filters are parameterized we include for each family a number of variations in our filter bank:

Filter	**Para.**	**Count**	**Instances**
Gaussian	Stdev. σ	20	$\sigma = \{0.5, 1, ..., 10\}$
Box	Size s	10	$s = \{5,\ 10,\ ...,\ 50\}$
Motion Blur	Size s, Angle α	20	$s = \{5,\ 10,\ ...,\ 50\}$ $\alpha = \{0^0, 90^0\}$
Sobel	n/a	2	horizontal, vertical
Color Offset	n/a	3	red, green, blue
Identity	n/a	1	
		$\sum$ **56**	

A linear combination of these basis filters enables approximating more complex filters; for example, a Laplacian filter can be approximated using a difference of Gaussians. Even many non-linear filters can be well approximated by the meta-filter due to its spatially varying nature. Figures 1a–b show a visualization of the optimized meta-filter weights for a highly non-linear example filter pair.

In Section 4 we describe how we learn meta-filters from example pairs using constrained optimization in the filter space. Optimizing the meta-filter over all basis filters, however, is prohibitively expensive. Therefore, we first select a smaller subset that is able to represent the latent filter $A \to A'$ well (Section 4.1), and carry out the optimization over this smaller set using an energy minimization formulation (Section 4.2) that can be efficiently optimized (Section 4.3). In Section 5 we discuss transferring the learnt filters to novel input images as well as editing the meta-filter parameters. In Section 6 we present our results, discuss optimization objective alternatives, and present extensive numerical and perceptual evaluations of our method.

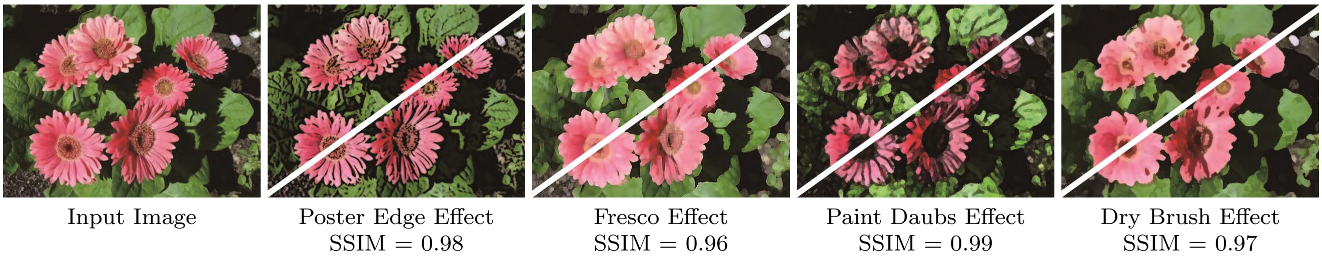


Fig. 2 Our learnt meta-filters approximate a wide variety of non-linear filters with high accuracy. The top left of the split figures shows the ground truth result A', while the bottom right shows our meta-filter approximation $f(A)$.

4 Learning Meta-filter Parameters

Given an image A and its filtered version A' produced by some latent filter or potentially a sequence of filters, our goal is to compute the parameters (i.e., weight maps) for the meta-filter f such that $f(A) \approx A'$.

4.1 Filter Selection

Our first task is to select a subset $S = \{f_i\}$ from the full filter bank that is still sufficient to represent the example $A \to A'$ well. This selection process makes the following optimization computationally tractable while still achieving high accuracy.

The following filters are always included in the subset, as our experiments showed they are almost always needed:

1. The *identity* filter, $f_{ID}(p) = A(p)$, which passes through the input color unchanged. It is useful when certain parts of the image are either unchanged or only changed by a linear mapping (e.g., contrast adjustments).
2. Three *color offset* filters, which provide a constant color offset for a specific channel:

   $$f_R(p)=(c,0,0)^\top,\ f_G(p)=(0,c,0)^\top,\ f_B(p)=(0,0,c)^\top,$$

   where $c = 0.01$ is a small empirically determined constant. The amount of actual offset is controlled by the weight map. The offset filters are particularly useful when the intensity or color of a region is shifted by a certain amount (e.g., brightness or tonal adjustments).

The initial filter subset $S^{(0)} = \{f_{ID}, f_R, f_G, f_B\}$ is now augmented by additional candidate filters $f_c \notin S^{(0)}$ that are found to be effective.

Each candidate filter is evaluated independently by finding the optimal weight map for the reduced meta-filter $\hat{f}_c$ that contains only the initial filter subset and the candidate itself,

$$\hat{f}_c = w_c f_c + \sum_{i \in S^{(0)}} w_i f_i. \tag{2}$$

such that $\hat{f}_c(A) \approx A'$. The details of this optimization are provided in the next subsections. The contribution of f_c is measured as the approximation error when it is used in isolation, i.e., $\sum_p \left(w_c f_c(p) - A'(p)\right)^2$. We include the two filters from each family that exhibit the lowest approximation errors into S.

Overall, S contains 12 filters: two from each of family of Gaussian, Box, Motion Blur, and Sobel, as well as the three color offset filters, and the identity filter. Our results demonstrate that this empirically determined filter selection heuristic works well in practice.

4.2 Energy Formulation

We formulate the task of determining the optimal weight maps for a given meta filter and filter example pair as an energy minimization problem. Our objective function comprises three terms.

The *data fitting* term, E_{data}, aims at approximating the filtering effect:

$$E_{data} = \sum_p \Big(\big(\sum_{i \in S} w_i(p) f_i(p)\big) - A'(p) \Big)^2. \tag{3}$$

The *smoothing* term, E_{smooth}, aims at reducing spatial variation in the weight maps:

$$E_{smooth} = \sum_p \sum_{i \in S} \left\| \nabla w_i(p) \right\|_1^1. \tag{4}$$

The term forces spatially close pixels to have similar weights and concentrates necessary changes into few pixels, yielding less fragmented and more homogeneous weight maps. Note, that we minimize the term in the L1 norm,

$$\begin{aligned} \left\|\nabla w_i(x,y)\right\|_1^1 \;=\; & \left|w_i(x{+}1,y) - w_i(x,y)\right| + \\ & \left|w_i(x,y{+}1) - w_i(x,y)\right|. \end{aligned} \tag{5}$$

In Section 6.4 we compare our L1 minimization against L2 minimization and show that ours leads to significantly improved results. Our formulation is related to total variation [13], however, here we seek sparsity of filter weights rather than of pixel intensities.

The third term, E_{sparse}, is essential to ensure the uniqueness of the solution:

$$E_{sparse} = \sum_{p} \sum_{i \in S} \left| w_i(p) \right|. \tag{6}$$

Without this term the system would become singular and numerically unstable. It also improves the concentration of weights at each pixel to fewer basis filters.

The overall energy is given as

$$E = \lambda E_{data} + E_{smooth} + \alpha E_{sparse}. \tag{7}$$

The balancing coefficients are empirically determined: $\lambda = 50$ prefers accuracy over smoothness, and $\alpha = 10^{-4}$ takes a small value just to ensure the stability of the solution.

Figure 2 demonstrates the ability of our meta-filters to approximate several non-linear filters from the Photoshop Filter Gallery. We measure the approximation quality using the Structure Similarity Image Metric (S-SIM) [29], which is widely used and known to be more consistent with perception than root mean square (RM-S) errors. In the supplementary material we provide extensive results to show that we can successfully approximate a wide range of filters.

4.3 Implementation

Let n denote the number of basis filters and m the number of pixels in A/A'. In matrix notation, we can rewrite Equation 7 as

$$\begin{aligned} E &= \lambda \underbrace{\|F\mathbf{W} - \mathbf{V}\|_2^2}_{E_{data}} + \underbrace{\|G\mathbf{W}\|_1^1}_{E_{smooth}} + \alpha \underbrace{\|\mathbf{W}\|_1^1}_{E_{sparse}} \\ &= \lambda \|F\mathbf{W} - \mathbf{V}\|_2^2 + \| \left(G \quad \alpha I\right)^{\top} \mathbf{W}\|_1^1. \end{aligned} \tag{8}$$

where $F_{m \times mn}$ is the matrix of precomputed basis filter results $f_i(p)$, $\mathbf{W}_{mn \times 1}$ is the vector of unknown basis weights $w_i(p)$, $\mathbf{V}_{mn \times 1}$ is the vector of pixel values from A', G is the matrix of the gradient operator in Equation 6, and $I_{mn \times mn}$ is the identity matrix.

This is an L1 regularized convex problem. The global minimum can be efficiently obtained using the Split Bregman method [21]. Let $\Phi = (G \quad \alpha I)^{\top}$. Using two additional vectors $\mathbf{b}$ and $\mathbf{d}$ and the unknown vector $\mathbf{W}$ (all initialized as zero vectors of length mn), we apply the following three steps iteratively until convergence:

S1: $\mathbf{W}^{k+1} = \min\limits_{\mathbf{W}} \frac{\lambda}{2}\|F\mathbf{W} - \mathbf{V}\|_2^2 + \frac{\gamma}{2}\|\mathbf{d}^k - \Phi\mathbf{W} - \mathbf{b}^k\|_2^2$

S2: $\mathbf{d}^{k+1} = \min\limits_{\mathbf{d}} \|\mathbf{d}\|_1^1 + \frac{\gamma}{2}\|\mathbf{d} - \Phi\mathbf{W}^{k+1} - \mathbf{b}^k\|_2^2.$

S3: $\mathbf{b}^{k+1} = \mathbf{b}^k + \Phi\mathbf{W}^{k+1} - \mathbf{d}^{k+1}.$

Here, k is the iteration number, and $\gamma = 10$ is a relaxation constant which affects the convergence rate but not the final result. Step 1 involves a quadratic function of $\mathbf{W}$. Denote $N(\mathbf{W}) = \frac{\lambda}{2}\|F\mathbf{W} - \mathbf{V}'\|_2^2 + \frac{\gamma}{2}\|\mathbf{d}^k - \Phi\mathbf{W} - \mathbf{b}^k\|_2^2$. The minimizer is computed using $\frac{\partial N(\mathbf{W})}{\partial \mathbf{W}} = \lambda F^{\top}(F\mathbf{W} - \mathbf{V}') + \gamma\Phi^{\top}(\Phi\mathbf{W} + \mathbf{b}^k - \mathbf{d}^k) = 0$. This is equivalent to solving the linear system $(\lambda F^{\top}F + \gamma\Phi^{\top}\Phi)\mathbf{W} = \lambda F^{\top}\mathbf{V}' - \gamma\Phi^{\top}(\mathbf{b}^k - \mathbf{d}^k)$. The matrix $(\lambda F^{\top}F + \gamma\Phi^{\top}\Phi)$ is symmetric positive definite and does not change over the course of the optimization. We use sparse Cholesky factorization [22] to efficiently decompose this matrix into $LDL^{\top}$ where L is a lower triangular matrix and D is a diagonal matrix. This only needs to be factorized once; during iteration the linear systems have triangular matrices and can be solved efficiently using substitution. Step 2 can be solved in linear time using the *shrink* operator (see [21]), and Step 3 is direct.

5 Applications

5.1 Filter Transfer

Once a meta-filter is learnt from an example pair $A \rightarrow A'$, it can be applied to novel input images B to obtain a filtered result $f(B)$ that approximates the (unknown) ground truth B'. To transfer the filter we establish pixel correspondence between A and B, and copy the weights of the elementary filters using the correspondence warp map.

Computing reliable correspondence between general images is a challenging problem. However, since we are only transferring basis filter weights between the images, obtaining exact correspondence is less critical. We use the state-of-the-art SIFT flow algorithm [4] to find an initial correspondence map that globally aligns the two images while well preserving spatial coherence. We found that SIFT flow sometimes does not work reliably around strong image edges. For that reason we refine (replace) the initial correspondence around strong edges with one that is computed using the PatchMatch algorithm [3] on Canny edge images extracted from A and B.

Figure 3 shows examples from before mentioned categories. The first row shows curve adjustment (see the inset figure in the filtered image). The second row shows an example of tone transfer. A similar result could be achieved by Wang et al.'s method [2]. However, while their method learns the tone adjustment filter from a dataset containing several examples, our method requires only a single example pair, as shown here. Rows three to six show various artistic stylization filters. These kinds of filter are more challenging to transfer. Finally, in the last row we learn and transfer a manual face polishing job (includes removing blemishes and wrinkles, and improving skin tone). Many more results are provided in the supplementary material.

The correspondence for all of our results is computed fully automatically, with the only exception being the face polishing results (last row in Figure 3). Here, we found it necessary to interactively select the skin regions. These are set as hard constraints and the remaining correspondence is computed as described above.

5.2 Filter Editing

The parameters of the meta-filter comprise the per-pixel weights of the basis filters $w_i(p)$, and their global parameters (i.e., the size of the Box, Gaussian, Motion Blur filters, as well as the Motion Blur angle). By manipulating these parameters, we can edit the learnt meta-filter in a semantic manner and obtain interesting controlled variations. For instance, we can increase or reduce all or some of the weights to yield a strengthened or weakened filter.

In Figure 4 we show some filter variations that were obtained through simple manipulations of the meta-filter parameters. The first row shows a manipulation of the Motion Blur basis filters: the blur size s is reduced to $0.5s$ to obtain a reduced "Motion Blur" effect (third column) and enlarged to $4.5s$ to obtain a strengthened motion blur (the forth column), while keeping the per-pixel weights unchanged. The second row shows a manipulation of the Sobel basis filters: the filter per-pixel weights $w_i(p)$ are uniformly reduced to 0.5× and increased by 8× to obtain reduced and strengthened "Poster Edge" effects. The third row shows results of a manipulation for the Box basis filters: the blur size s is reduced/increased to $0.5s/4s$ to obtain a reduced/ strengthened "Color Cut" effect. Many other filter editing results are provided in the supplementary material.

In Figure 1d we compare a simple meta-filter manipulation of the Box blur size against the result achieved by naïve filter strengthening.

6 Results and Evaluation

We tested our algorithm with a wide range of common image filters, including artistic filters, tone adjustment, color transfer, curves, and manual image edits. For effects generated by automatic algorithms (such as Photoshop filters), the same algorithms are used to obtain the ground truth images. More complicated effects involve manually applying various filters to selected regions. For example, the "Gouache" effect in Figure 3 was created by an artist using a combination of smart blur, overlay, paint daubs, hue/saturation adjustment, curve adjustment etc. to selected regions using manual layering. The ground truth results of such effects were also created by artists. It typically takes 15-20 minutes for an artist to create such effects for a given image. Apart from face polishing, which required minimal user interaction, all results were achieved fully automatically using the same algorithm settings (as described in the paper).

6.1 Comparison to Image Analogies

In Figure 5 we compare our method against Image Analogies [1]. In contrast to their method, ours does not synthesize a new image by stitching small patches, but rather transfers a set of basis filters. For this reason our method is less sensitive to exact correspondence and avoids several artifacts present in the Image Analogies results.

In the supplementary material we include a more extensive ground truth comparison with their method on a larger number of image filters and target images. Our numerical analysis shows that our method increases the average SSIM score from 0.34 (Image Analogies) to 0.61 (Our results).

6.2 User Study

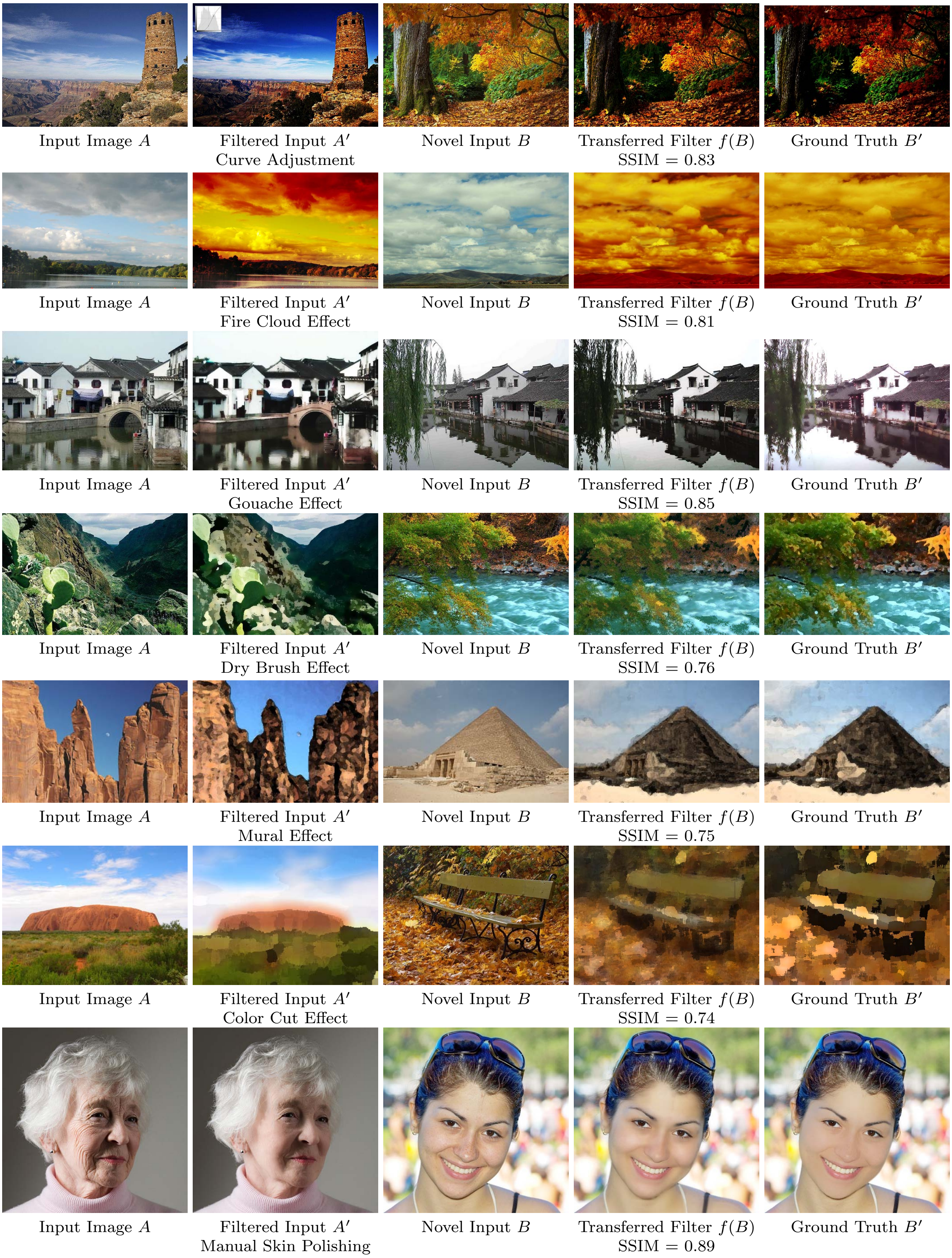
We validated our algorithm further by conducting a formal user study with 20 participants (25% female, ages ranging from 18 to 29). For this study we generated 72 filter transfer examples with our method and Image Analogies [1] using the software provided on their project page. The images we used for our study are included in the supplementary material.

In each test we showed the participant the input images A, A', B and two choices for B', one produced by our algorithm, and the other either produced by Image Analogies, or the actual ground truth result. Participants were asked which result was closer to the transfer result they would imagine (Two-Alternative Forced Choices, or 2AFC).

The results of our study are summarized in Figure 6. When comparing against Image Analogies participants chose our method in 73.7% of all cases. When comparing against ground truth participants still chose our method in 45.8% of all cases.

6.3 Filter Bank

We validate that our filter bank contains enough variation in filter families and instances to support our target applications, and is minimal in a sense that it



Input Image A

Filtered Input A'
Curve Adjustment

Novel Input B

Transferred Filter $f(B)$
SSIM = 0.83

Ground Truth B'

Input Image A

Filtered Input A'
Fire Cloud Effect

Novel Input B

Transferred Filter $f(B)$
SSIM = 0.81

Ground Truth B'

Input Image A

Filtered Input A'
Gouache Effect

Novel Input B

Transferred Filter $f(B)$
SSIM = 0.85

Ground Truth B'

Input Image A

Filtered Input A'
Dry Brush Effect

Novel Input B

Transferred Filter $f(B)$
SSIM = 0.76

Ground Truth B'

Input Image A

Filtered Input A'
Mural Effect

Novel Input B

Transferred Filter $f(B)$
SSIM = 0.75

Ground Truth B'

Input Image A

Filtered Input A'
Color Cut Effect

Novel Input B

Transferred Filter $f(B)$
SSIM = 0.74

Ground Truth B'

Input Image A

Filtered Input A'
Manual Skin Polishing

Novel Input B

Transferred Filter $f(B)$
SSIM = 0.89

Ground Truth B'

Fig. 3 Transferring learnt meta-filters to novel input images. A more extensive set of results can be found in the supplementary material.

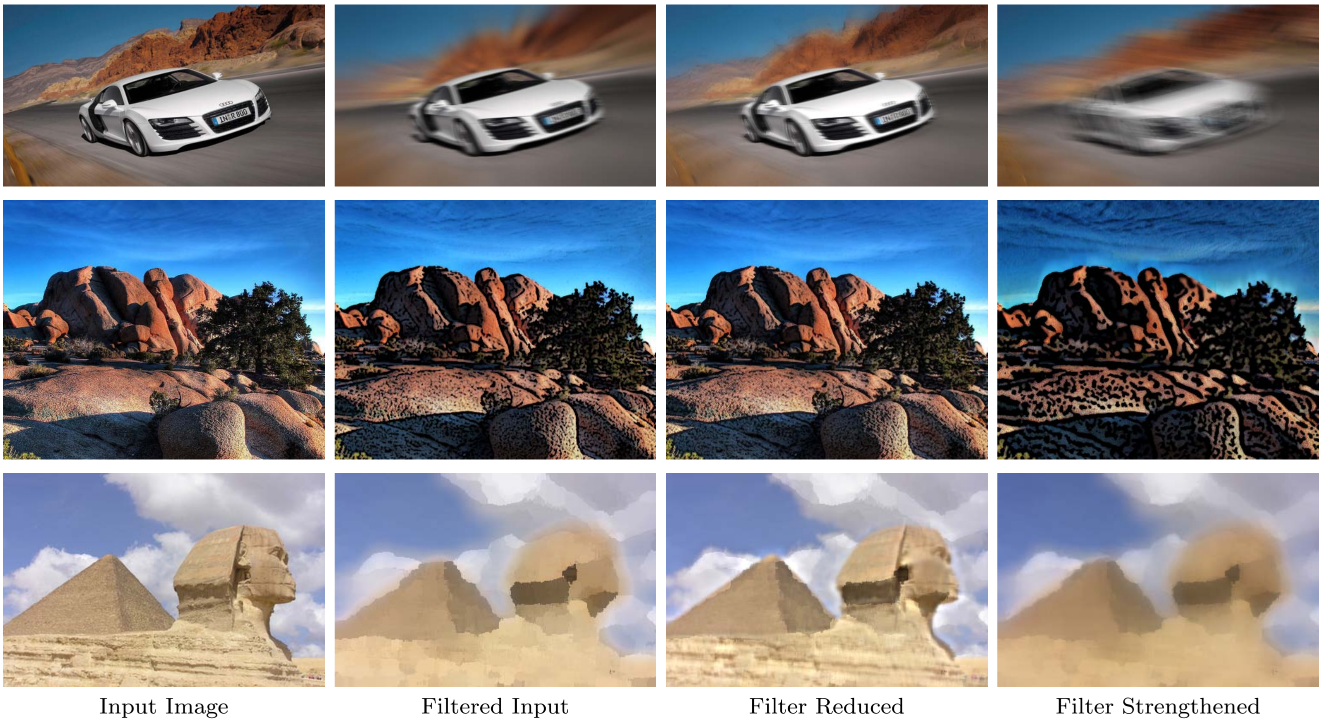


Fig. 4 Filter editing results. Given the original (first column) and filtered (second column) input images, the effect can be easily manipulated to obtain reduced (third column) and strengthened (fourth column) results.

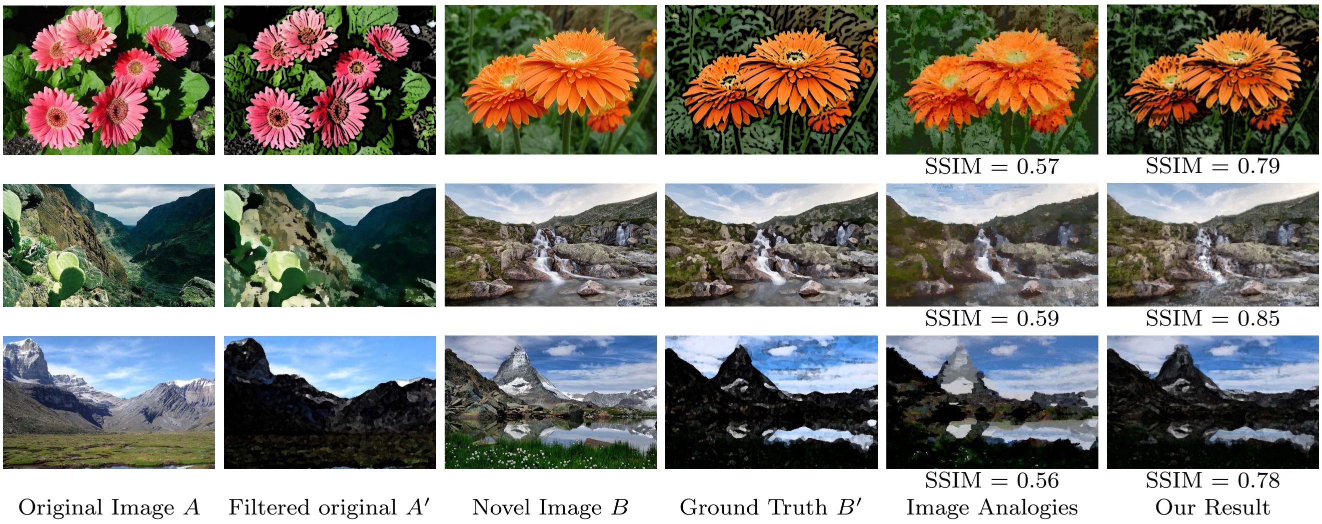


Fig. 5 Comparing our results with Image Analogies [1].

does not contain more filters than necessary. Our results throughout the paper and supplementary material demonstrate that the filter bank is able to represent a wide range of common image filters well. To show that it is minimal we perform a series of "leave-one-out" tests, in which we show that each subset of the filter bank where one whole family is removed yields poor results at least for some input pairs.

We evaluate the approximative power of the meta-filter as well as its ability to transfer filters to novel input images. For this task we prepared images A, A', B, B' using filters from the Photoshop Filter Gallery, and then compare the approximation results $f_{full}(A)/f_{subset}(A)$ and transfer results $f_{full}(B)/f_{subset}(B)$ against their respective ground truths A' and B'. Here, f_{full} is the meta-filter learnt using the full filter bank, and f_{subset} is a meta-filter learnt using a filter bank in which one of the filter families is removed. We compare the images both numerically using SSIM score, as well as through visual inspection.

Our experiments showed that the approximation quality does *not* suffer much from removing single filter families. However, we found that it can have significant impact on the ability to *transfer* filters to novel input images, which is our main application. In the supplementary material we show results from our experiments that demonstrate how leaving each of the basic filter families out significantly affects the quality of transferred meta-filters on at least one important class of image filters. These experiments support our claim that all families in our filter bank are necessary for our target application.

6.4 L1 minimization

Our meta-filter learning algorithm uses $L1$ minimization objectives. In order to validate this design choice we tested two alternatives: (1) leaving out the sparsity term E_{sparse}, and (2) replacing the smoothness term E_{smooth} with an $L2$ objective.

Removed Sparsity Term $\boldsymbol{E}_{sparse}$*:* As mentioned in Section 4.2, the sparsity term E_{sparse} is necessary to ensure the numerical stability of the solution. When removing this term from the optimization objective, the **S1** term of the Split-Bregman method reduces to

$$\mathbf{W}^{k+1} = \min_{\mathbf{W}} \frac{\lambda}{2}\|F\mathbf{W} - \mathbf{V}'\|_2^2 + \frac{\gamma}{2}\|\mathbf{d}^k - G\mathbf{W} - \mathbf{b}^k\|_2^2,$$

which amounts to solving the least square problem

$$\mathbf{W}^{k+1} = \min_{\mathbf{W}} \|(\frac{\lambda}{2}F \quad G)^T\mathbf{W} - (\mathbf{V}' \quad \mathbf{d}^k - \mathbf{b}^k)^\top\|_2^2.$$

The problem lies with the least square matrix $A = (\frac{\lambda}{2}F \quad G)^\top$, which is highly singular. Solving for it is numerically unstable and very time consuming. Adding the sparsity term yields $A = (\frac{\lambda}{2}F \quad G \quad \alpha I)^T$, which is non-singular and can be robustly solved.

The Smoothness Term $\mathbf{E}_{smooth}$*:* An interesting design alternative is to replace the smoothness term with a $L2$ version:

$$E^{L2}_{smooth} = \sum_{p}\sum_{q\in N(p)}\sum_{i\in S}\big(w_i(p) - w_i(q)\big)^2 \tag{9}$$

This leads to a simpler optimization that can be solved much more quickly than solving the $L1$ energy (about 3× faster in our experiments). However, the approximation and transfer quality suffers dramatically for some filters, especially around edges in the images. We show some exemplary comparisons between results achieved with $L1$ and $L2$ optimization in the supplementary material.

6.5 Performance

We tested our MATLAB implementation on a dual Intel Core2Quad CPU at 2.4GHz. Our implementation is not optimized. Given an image of size 500 × 375 our filter learning algorithm implemented requires 1–3 minutes for filter selection and 1–2 minutes for meta-filter learning. Once the filter is learned, transferring it to novel images takes only about 2 seconds.

6.6 Limitations

Our current filter transfer algorithm performs less successfully for filters that create texture-like structures, as shown in Figure 7. This is partially due to our method for establishing correspondence which does not transfer structures in the filtering effect well. Alternative methods may be adopted to alleviate this.

Filters that depend not on image content, but only on the spatial position within the image (e.g., tilt-shift effect) can be well approximated by our meta-filter, but they do not transfer well to novel image, because the correspondence algorithm takes only the image content into account but not the position within the image.

Our current algorithm assumes that the example image pairs are well aligned. Effects that involve warping, projective transform, or any transform that involves moving pixels around cannot be approximated by the meta-filter. We are considering extending our method and integrating image registration methods to establish correspondences between pairs of images. However, these are not simple problems and are left for future research.

7 Conclusions

We have introduced a meta-filter that linearly combines spatially varying filters. We have presented a minimization technique with an L_1 regularization term that optimizes the weights of the meta-filter to approximate a general filter whose operation is determined from a before and after pair of examples.

Our meta-filter is a simplified model that, nevertheless, spans a surprisingly large space of filters that can well approximate various effects that were generated by applying a sequence of a number of unknown filters. We speculate that part of the power of our meta-filter stems from the fact that it is spatially varying, enriching the possible effects considerably.

In the future we want to explore the possibility of learning the generation of intermediate level filters. Such

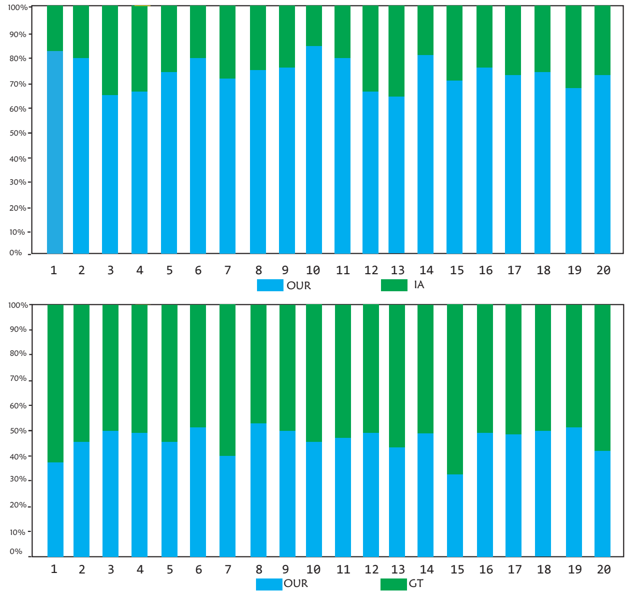
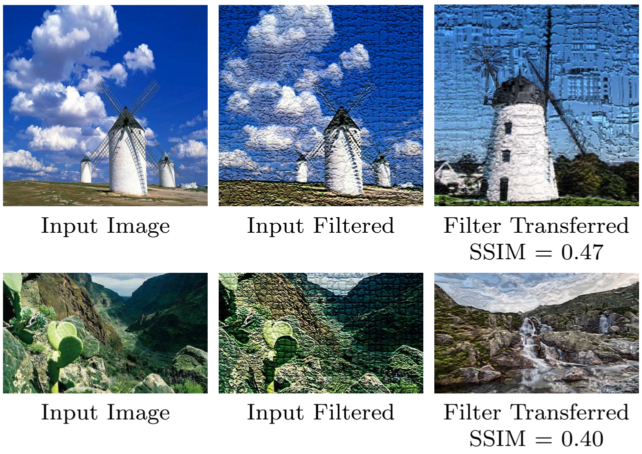


Fig. 6 Results of the user study. Top: the percentage in which participants chose our result (OUR) over Image Analogies (IA), broken down per participant. Bottom: results for our method compared against ground truth (GT).



Input Image Input Filtered Filter Transferred SSIM = 0.47

Input Image Input Filtered Filter Transferred SSIM = 0.40

Fig. 7 Limitation of our method: our method performs sometimes less successfully for transferring texture effects.

filters can be learnt from a large set of common and useful filters, and encapsulate the functionality of a series of low level filtering operations. We believe that such intermediate level filters can further strengthen the quality of the meta-filter, as well as improving its speed and expanding its capabilities.

Acknowledgements This work was supported by the National Basic Research Project of China (Project Number 2011 CB302205), the Natural Science Foundation of China (Project Number 61120106007, 61133008), the National High Technology Research and Development Program of China (Project Number 2012AA011802) and Tsinghua University Initiative Scientific Research Program.

References

1. Aaron Hertzmann, Charles E. Jacobs, Nuria Oliverm Brian Curless and David H. Salesin, Image analogies, Proc. ACM SIGGRAPH, 327–340(2001)
2. Baoyuan Wang, Yizhou Yu and Ying-Qing Xu, Example-based image color and tone style enhancement, ACM Trans. Graph., 30, 4, 64:1–64:12(2011)
3. Connelly Barnes, Eli Shechtman, Adam Finkelstein and Dan B Goldman, PatchMatch: a randomized correspondence algorithm for structural image editing, ACM Trans. Graph., 28, 3, 24:1–24:11(2009)
4. Ce Liu, Jenny Yuen and Antonio Torralba, Sift flow: Dense correspondence across scenes and its applications, IEEE Trans. Pattern Anal. Mach. Intell., 33, 5, 978–994(2011)
5. Ding Ziang, Zhang Xin, Chen Wei, Tricoche Xavier, Peng Dichao, Peng, Qunsheng, Coherent streamline generation for 2-D vector fields, Tsinghua Science and Technology, Volume:17 , Issue: 4 , 463 - 470(2012)
6. Farhang Sahba and Hamid R. Tizhoosh, Filter Fusion for Image Enhancement using Reinforcement Learning, Proc. IEEE Canadian Conference on Electrical and Computer Engineering, 847–850(2003)
7. Floraine Berthouzoz, Wilmot Li, Mira Dontcheva and Maneesh Agrawala, A Framework for content-adaptive photo manipulation macros: application to face, landscape, and global manipulations, ACM Trans. Graph., 30, 5, 120:1–120:14(2011)
8. Hua Huang and Xuezhong Xiao, Example-based contrast enhancement by gradient mapping, The Visual Computer, Vol. 26, 6-8, 731-738(2010)
9. H. Ji and K. Wang, Robust image deblurring with inaccurate blur kernels, IEEE Trans. Image Processing, 21, 4, 1624–1634(2012)
10. J. Mairal, F. Bach, J. Ponce, G. Sapiro and A. Zisserman, Non-local sparse models for image restoration, Proc. IEEE International Conference on Computer Vision (ICCV), 2272–2279(2009)
11. John C. Russ, The Image Processing Handbook (Fifth Edition), CRC Press, 2006
12. Kaiming He, Jian Sun and Xiaoou Tang, Guided image filtering, Proc. European Conference on Computer Vision: Part I, 1–14(2010)
13. Leonid I. Rudin, Stanley Osher and Emad Fatemi, Nonlinear total variation based noise removal algorithms, Physica D, 60, 259–268(1992)
14. Xian-Ying Li, Yan Gu, Shi-Min Hu, and Ralph R. Martin, Mixed-Domain Edge-Aware Image Manipulation IEEE Transactions on Image Processing, Vol. 22, No. 5, 1915 - 1925(2013)
15. Lu Yuan, Jian Sun, Long Quan and Heung-Yeung Shum, Image deblurring with blurred/noisy image pairs, ACM Trans. Graph., 26, 3, 1:1–1:10(2007)
16. Menon D. and Calvagno G., Regularization Approaches to Demosaicking, IEEE Trans. Image Processing, 18, 10, 2209–2220(2009)
17. N. Joshi, R. Szeliski and D.J. Kriegman, PSF estimation using sharp edge prediction, Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 1–8(2008)
18. Pierre-Yves Laffont, Adrien Bousseau, George Drettakis and Rich Intrinsic Image Decomposition of Outdoor Scenes from Multiple Views, IEEE Transaction on Visualizations and Computer Graphics, Vol. 19, No. 2, 210-224(2013)
19. Shi-Min Hu, Tao Chen, Kun Xu, Ming-Ming Cheng, Ralph R. Martin, Internet visual media processing: a survey

with graphics and vision applications, The Visual Computer, 29, 5, 393–405(2013)

20. Sing Bing Kang, Ashish Kapoor and Dani Lischinski, Personalization of image enhancement, CVPR, 1799–1806(2010)

21. T. Goldstein and S. Osher, The Split Bregman Method for L_1 Regularized Problems, SIAM Journal on Imaging Sciences, 2, 2, 323–343(2009)

22. Timothy A. Davis, CHOLMOD: a sparse Cholesky factorization and modification package, Univ. of Florida, 2011

23. Vladimir Bychkovsky, Sylvain Paris, Eric Chan and Frédo Durand, Learning photographic global tonal adjustment with a database of input/output image pairs, CVPR, 97–104(2011)

24. Xiaohui Wang, Jia Jia and Lianhong Cai, Affective image adjustment with a single word, The Visual Computer, Vol. 29, No. 11, 1121-1133(2013)

25. Wang, Xiao-Hui; Jia, Jia; Liao, Han-Yu; Cai, Lian-Hong, Affective Image Colorization, Journal of Computer Science and Technology, Vol. 27, No. 6, 1119-1128(2012)

26. Ying Tang, Xiaoying Shi, Tingzhe Xiao, Jing Fan, An improved image analogy method based on adaptive CUDA-accelerated neighborhood matching framework, The Visual Computer, Vol. 28, No. 6-8, 743-753(2012)

27. Yun Ling, Caiping Yan, Chunxiao Liu, Xun Wang, Hong Li, Adaptive tone-preserved image detail enhancement, The Visual Computer,Vol. 28, No.6-8, 733-742(2012)

28. Zang, Yu; Huang, Hua; Li, Chen-Feng, Stroke Style Analysis for Painterly Rendering, Journal of Computer Science and Technology, No. 28, No. 5, 762-775(2013)

29. Zhou Wang, A. C. Bovik, H. R. Sheikh and E. P. Simoncelli, Image quality assessment: from error visibility to structural similarity, IEEE Trans. Img. Proc., 13, 4, 600–612(2004)

30. Zhuo Su, Xiaonan Luo and Alessandro Artusi, A novel image decomposition approach and its applications, The Visual Computer, No. 29, No. 10, 1011-1023(2013)

Shi-Sheng Huang is a Ph.D. candidate at Tsinghua University in Beijing, his research interests include: Shape Analysis, Point Cloud Processing and Image Processing.

Guo-Xin Zhang received the Ph.D. degree in the Department of Computer Science and Technology, Tsinghua University in 2012. His research interests include computer graphics, geometric modeling, and image processing.

Yu-Kun Lai received the bachelors and PhD degrees in computer science from Tsinghua University, China in 2003 and 2008, respectively. He is currently a lecturer of visual computing in the School of Computer Science and Informatics, Cardiff University, Wales, UK. His research interests include computer graphics, geometry processing, image processing, and computer vision. He is on the editorial board of The Visual Computer.

Johannes Kopf received his bachelors degree and PhD degree in computer science from University of Hamburg (2003) and University of Konstanz (2008), respectively. He is currently a researcher at Microsoft Research in Redmond. His research is mainly in the area of Computer Graphics and Computer Vision.

Daniel Cohen-Or is a Professor at the Department of Computer Science, Tel Aviv University. His research interests are in Computer Graphics, Visual Computing and Geometric Modeling. He was on the editorial board of several international journals including CGF, IEEE TVCG, The Visual Computer and ACM TOG, and regularly serve as a member of the program committees of international conferences.

Shi-Min Hu is currently a professor in the Department of Computer Science and Technology at Tsinghua University, Beijing. He received the PhD degree from Zhejiang University in 1996. His research interests include digital geometry processing, video processing, rendering, computer animation, and computer aided geometric design. He is associate Editor-in-Chief of The Visual Computer, and on the editorial boards of IEEE TVCG, Computer-Aided Design and Computer and Graphics.