EUROGRAPHICS 2008 / G. Drettakis and R. Scopigno
(Guest Editors)

Volume 27 (2008), Number 2

Image-based Material Weathering

Su Xue[†1], Jiaping Wang[2], Xin Tong[2], Qionghai Dai[1], Baining Guo[2]

[1]Tsinghua University, [2]Microsoft Research Asia

Abstract

*The appearance manifold [WTL*06] is an efficient approach for modeling and editing time-variant appearance of materials from the BRDF data captured at single time instance. However, this method is difficult to apply in images in which weathering and shading variations are combined. In this paper, we present a technique for modeling and editing the weathering effects of an object in a single image with appearance manifolds. In our approach, we formulate the input image as the product of reflectance and illuminance. An iterative method is then developed to construct the appearance manifold in color space (i.e., Lab space) for modeling the reflectance variations caused by weathering. Based on the appearance manifold, we propose a statistical method to robustly decompose reflectance and illuminance for each pixel. For editing, we introduce a "pixel-walking" scheme to modify the pixel reflectance according to its position on the manifold, by which the detailed reflectance variations are well preserved. We illustrate our technique in various applications, including weathering transfer between two images that is first enabled by our technique. Results show that our technique can produce much better results than existing methods, especially for objects with complex geometry and shading effects.*

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation; I.4.10 [Computing Methodologies]: Image processing and computer vision-Image representation

1. Introduction

Most materials in the real world change their appearance over time due to the interactions between the materials and their surrounding environment. Modeling and editing these appearance variations caused by weathering (i.e., weathering effects) are not only crucial for enhancing realism in rendering, but also important for image manipulation due to the ubiquitous presence of weathering effects in real images. In this paper, we study the problem of modeling and editing the weathering effects of an object shown in a single image.

Lots of techniques have been developed for modeling and rendering various weathering effects of 3D scenes [HW95, GTR*06, DH96, DPH96, DEJ*99, CXW*05, LGG*07]. Unfortunately, none of these techniques can be directly used for images or photos, where the scene geometry, illumination and material properties are unknown. Recent advances [FH04, KRFB06] in image manipulation allow users to replace the object appearance shown in an image with new textures [FH04] or the appearance of a new material [KRFB06]. However, without any prior knowledge on weathering effects exhibited on the object surface, it is laborious to use these methods to generate a sequence of images with consistent weathering appearance from the weathered object shown in a single image.

Most recently, Wang et al. [WTL*06] introduced appearance manifolds for modeling the time-variant appearance of materials. Their method takes advantage of the key observation that the variations of reflectance captured over the weathered surface at single time instance represent the different degrees of weathering. In a simple attempt to model weathering effects from a single image with the appearance manifold, they construct the neighborhood graph of pixels with only pixel chroma values, and form an appearance manifold. The principal monotonic variation along the manifold reveals the relative degrees of weathering. After the weathering degree of each pixel is determined by its position in the manifold, the luminance of the pixels of the same weathering degree are averaged to be the reflectance luminance

† This work was done when Su Xue was a visiting student in Microsoft Research Asia.

Figure 1: *Image-based material weathering. From left to right: input photograph; deweathered result; weathered result; weathering material transfer between different images (from a mossy stone statue to the rusty iron tap).*

Figure 2: *Left: A failure result generated by the method described in [WTL*06]. Note that the artifacts in the shadow area are caused by the incorrect decomposition generated from noisy chroma values, while the intensive luminance changes of some pixels is caused by ignoring the luminance variation in manifold construction. Right: Our result generated with the same weathering degree map, which shows good decomposition and smooth weathering variation.*

for these pixels. The shading effect is then separated from weathering by dividing the pixel luminance value by their reflectance luminance. Given the modified weathering degrees for all pixels, the reflectance of the pixels is modified by interpolating along the shortest path from its point in the appearance manifold to the most/least weathered point sets.

Although this simple scheme works well for some images, it has several problems. First, the appearance manifold and the weathering degrees determined by chroma values cannot well model the nonlinear luminance variations caused by weathering. As a result, a small variation of weathering degree may lead to large luminance changes in editing results. Second, noise in chroma values significantly affects the weathering degrees of pixels and would result in an incorrect decomposition between shading and weathering, especially for pixels in shadowed areas. Finally, for a curved appearance manifold, the simple interpolation scheme used in weathering editing cannot preserve the secondary reflectance variations and loses details in the result image. Fig. 2 illustrates a result image generated by the method in [WTL*06].

In this paper, we propose a robust technique for modeling and editing the weathering effects of an object shown in a single image with its appearance manifold. Similar to [WTL*06], we focus on weathering effects of homogeneous materials that exhibit progressive reflectance variation. In our approach, we model the image as the product of reflectance and illuminance and assume that the chroma values of the illuminance are constant in the input image. Instead of constructing the appearance manifold with chroma values of pixels only, we develop an iterative method to reconstruct the appearance manifold for modeling the reflectance (both chroma and luminance) variations caused by weathering. A statistical approach is then applied to robustly estimate the weathering degree of each pixel. With this new approach, the shading and weathering effects in the input image are well decomposed. For editing, we parameterize secondary variations in the appearance manifold and introduce a "pixel walking" scheme to modify the pixel's reflectance. Our new editing scheme can well preserve the reflectance variations with the same weathering degree in the result image. Moreover, the parameterized manifold also enables us to transfer the weathering effects from one image to another.

Since our method is based on appearance manifolds, it has the limitations of the appearance manifold approach described in [WTL*06]. Specifically, our approach cannot be used for weathering effects with geometry variations and abrupt appearance variations, such as peeling and cracks. Moreover, since our method relies on chroma variations to decompose the shading from reflectance, it cannot well handle weathering effects with few chroma variations, such as drying [LGD*05]. Finally, as a pixel-based approach, our method ignores the time-variant texture variations over the object surface. Despite these limitations, our method generates convincing results for a wide variety of weathering effects captured with different illuminations and object geometry. Fig. 1 and Fig. 2 show results generated by our approach.

The remainder of this paper is organized as follows. After reviewing the related work in Section 2, we describe the details of our technique in Section 3 and then discuss the experimental results in Section 4. We finally conclude the paper in Section 5.

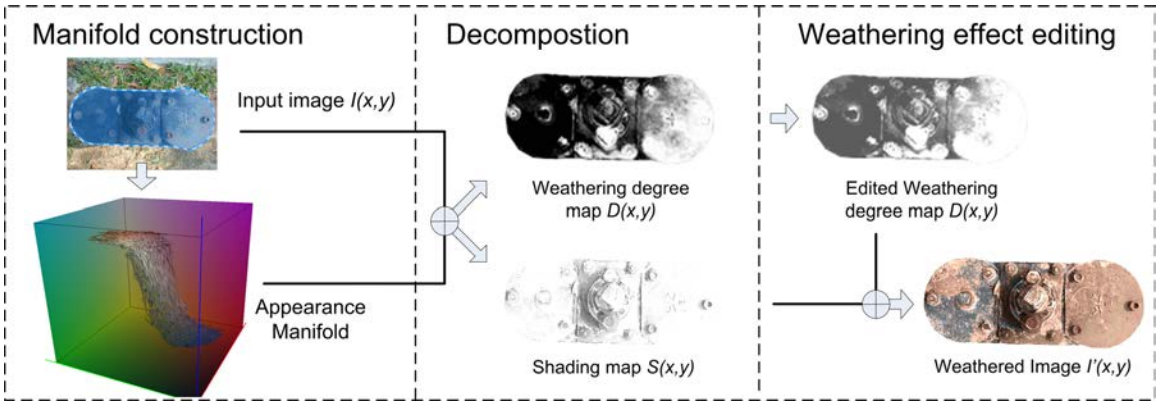

Figure 3: *The overview of our image-based material weathering system.*

2. Related Work

Physically-based simulation has been widely used for modeling specific weathering effects, such as metallic patinas [DH96], stone erosion [DEJ*99], paint cracks and peeling [PPD02], water flow [DPH96], corrosion [MDG01], stretches [BPMG04], and lichen growth [DGA04]. Although these methods can achieve realistic results, they need specialized physical parameters and processes for different weathering effects, which are difficult to derived from a single image. A set of techniques [Mil94, HW95, CXW*05] have been proposed to generate plausible weathering effects by simulating the interactions between object geometry and weathering factors in the environment. Combined with texture maps [CXW*05,LGG*07] or captured time-variant material reflectance [LGD*05,GLX*05,GTR*06], these methods can successfully generate convincing rendering results for different weathering effects. However, these techniques can not be used in our scenario where the geometry and environment are unknown. Our approach follows the method in [WTL*06] and can be regarded as a visual simulation technique. In our approach, we reconstruct the weathering degree map of the weathered object surface and the time-variant material reflectance with the appearance manifold. Without prior knowledge on scene geometry and environment, we allow the user to edit the weathering degree map and automatically generate the plausible weathering effects.

Additional related work deals with reflectance/illumination separation, which is a long standing research topic in computer vision and graphics. Nayar et al. [NB93] introduced an approach for recovering reflectance for a non-textured surface. In [TFA05, TAF06], the reflectance is decomposed from shading by utilizing classifiers or local estimates learned from training image sets. Finlayson et al. [FHD02, FDL04] relied on the assumptions that the light source is Planckian and pixel values are in accord with the light source property. While in our approach, we only use information in the input image for separation and we assume that the chroma of lighting is constant. Oh et al. [OCDD01] explore the reconstruction of the geometry, illuminance and surface texture of the scene shown in an image. By assuming the texture on the surface is a regular, high frequency pattern and the shading is low frequency on surface, the shading is decomposed from a surface texture via a bilateral filter. This method cannot be used for weathered objects with progressive appearance variations that are handled in our application. Khan et al. [KRFB06] edit the object material with a single HDR image as input. They employed a depth recovery scheme and computed surface normals from shading. By reconstructing illuminance from the surrounding environment, BRDF replacement and re-texturing can be performed over the object surface. In contrast to their approach, we directly separate illuminance from reflectance. Since our approach does not need any knowledge on object geometry, we can well handle objects with complex geometry or shading effects.

3. Image based material weathering

An overview of our system is illustrated in Fig. 3. Given an input image, user interaction is initially required to segment out the region of weathered object. The pixel $I(x,y)$ in the selected region is modeled as the product of reflectance $R(x,y)$ and illuminance $S(x,y)$ in Lab color space,

$$I_c(x,y) = R_c(x,y) * S_c(x,y) \quad (1)$$

$$I_l(x,y) = R_l(x,y) * S_l(x,y), \quad (2)$$

where the terms with subscript c denote the values in the chroma(ab) channels, and those with l denote values in the luminance (L) channel. Since the principal reflectance variations in the image are attributed to material weathering, we also denote reflectance $R(x,y)$ as the weathering component of pixel $W(x,y)$, and the illuminance $S(x,y)$ is denoted as the shading component of pixel. By assuming that

the chroma of illuminance S_c is constant, we further incorporate S_c into $W_c(x,y)$ and denote $S_l(x,y)$ as the shading component $S(x,y)$. Thus we have

$$I_c(x,y) = R_c(x,y) * S_c = W_c(x,y) \quad (3)$$
$$I_l(x,y) = W_l(x,y) * S(x,y) \quad (4)$$

To edit the weathering effects in $I(x,y)$, we should follow the underlying weathering process to modify the weathering component W(x,y) and then obtain the new image under the same shading $S(x,y)$. As shown in Fig. 3, this can be done in three steps. In the first step, we iteratively construct a neighborhood graph among pixels in Lab space to create the appearance manifold of an input image. During construction, we compute the weathering component $W(v)$ and its weathering degree $D(v)$ for each point v in the appearance manifold. In the resulting appearance manifold, the weathering process is modeled as a monotonic weathering degree evolution that corresponds to the principal mode of reflectance variation. In the second step, we then decompose the shading and weathering component for each pixel. To this end, we project each pixel in the appearance manifold and infer its weathering component $W(x,y)$ and weathering degree $D(x,y)$. The shading component $S(x,y)$ is simply decomposed from $I_l(x,y)$ as $S(x,y) = I_l(x,y)/W_l(x,y)$. We also obtain a weathering degree map $D(x,y)$ that characterizes distributions of weathering effects in the input image, which allows the user to easily edit the weathering effect. In the last step, we compute the weathering component $W'(x,y)$ for the new weathering degree map $D'(x,y)$ with the pixel walking scheme and render the final image as:

$$I'_c(x,y) = W'_c(x,y) \quad (5)$$
$$I'_l(x,y) = W'_l(x,y) * S(x,y) \quad (6)$$

For discrimination in later narration, the variables and functions of pixels use (x,y) as input, while those of the manifold points take v as input.

3.1. Appearance manifold construction

Given pixels of interest in the input image, we form an appearance manifold by constructing the neighborhood graph for the weathering components of pixels. Since the luminance of the weathering component in each pixel is unknown, we first construct the neighborhood graph in the chroma (ab) plane and then iteratively update the appearance manifold in Lab space. Before construction, we manually exclude heavily shadowed pixels that suffer severe chroma noise [LFSK06] to reduce the error in appearance manifold construction. As shown in Fig. 4, the region with heavily shadowed pixels is marked in white and removed with simple image operations. All other pixels are used in the appearance manifold construction. We will discuss how to handle these pixels later in the shading decomposition step.

Firstly we use only chroma values of pixels $I_c(x,y)$ to build the initial neighborhood graph between points by following the method described in [WTL*06]. Specifically, we utilize the k-rule to connect each point to its k nearest neighbors ($k = 8$ for all images shown in the paper) and then apply the ε-rule to prune the connections between points whose distance is larger than a threshold ε. We also set $\varepsilon = 1.5d_p$ where d_p is the average distance between the connected points after applying the k-rule. In this step, the distance between the two points is computed by the L^2 distance in Lab space.

Figure 4: *User input in manifold construction. The heavily shadowed region is marked in white and excluded before construction. While the red and green regions indicate the most weathered and least weathered pixels.*

Since we assume that the illuminance chroma is constant, the chroma variations in the input image are mainly caused by weathering. To infer the weathering degree in the initial neighborhood graph, we let the user identify the least weathered and most weathered points in image (shown as the green and red points in Fig. 4). We then compute the convex sets of the least and most weathered points V_0, V_1 in the appearance manifold using the method described in [WTL*06]. The weathering degree of a point v in the initial appearance manifold is defined as

$$D(v) = \phi(v,v_0)/(\phi(v,v_0) + \phi(v,v_1)), \quad (7)$$

where v_0 and v_1 are respectively the closest points in V_0 and V_1. $\phi(v,v_i)$, $i = 0,1$ is the geodesic distance between two points in the appearance manifold (e.g., the length of shortest path along the connected nodes between v and v_i). With this definition, the weathering degree D is defined within $[0.0, 1.0]$, with 0.0 indicating the least weathered points and 1.0 referring to the most weathered points (Fig. 5(b)).

Although we can get the weathering degree from the initial appearance manifold, it is biased because the reflectance luminance variations caused by the weathering are not taken in consideration. To solve this problem, we develop a statistical method to estimate the luminance variations caused by weathering and update the appearance manifold accordingly. Specifically, for all pixels that correspond to points in the appearance manifold with specific weathering degree d, the expectation of their luminance under d is represented as

$$E(I_l(x,y)|d) = E(S(x,y) * W_l(x,y)|d) \quad (8)$$

Since the distribution of the shading component $S(x,y)$ (i.e. illuminance) and the distribution of the weathering compo-

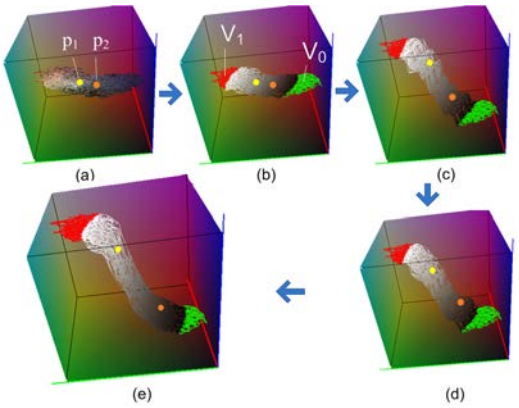

Figure 5: *Iterative construction of the appearance manifold. (a)Initial neighborhood graph using only chroma values. (b)Initial weathering degrees in the appearance manifold. V_0 and V_1 are convex sets of the least and the most weathered points. (c)Updated appearance manifold after luminance estimation. (d)Updated weathering degrees in the appearance manifold. (e)Final appearance manifold and weathering degrees after iterations. Note the difference of the weathering degrees of p_1 and p_2 in the initial appearance manifold and in the final appearance manifold after an iterations.*

nent $W_l(x,y)$ (i.e. reflectance) in the input image are independent, the expectation of the illuminance (i.e., shading component) under different weathering degrees should be invariant and is the same as the expectation of the shading component over the whole image. So we have

$$E(I_l(x,y)|d) = S * E(W_l(x,y)|d), \tag{9}$$

where S is the expectation of shading in the input image. Since S is a image-specific constant and is unchanged after weathering editing, we directly assign $E(I_l(x,y)|d)$ as the luminance of weathering components $W_l(v)$ for points v with the weathering degree d in the appearance manifold. After that, we construct the neighborhood graph of all points in the original appearance manifold with their updated weathering components in Lab space and update their weathering degrees accordingly. In the new appearance manifold, we estimate the luminance for each weathering degree and repeat the above process until the appearance manifold converges to the final shape.

In our current implementation, we uniformly partition weathering degrees defined within $(0,1)$ into $n = 100$ discrete intervals and put each point in the appearance manifold in the corresponding interval according to its weathering degree. For each interval, we compute $E(I_l(x,y)|d)$ as the mean of the luminance values of pixels that correspond to points in the interval and update the luminance of the weathering components of points by $W_l(v) = E(I_l(x,y)|d)$. As illustrated in Fig. 5, after several iterations, an appearance manifold is ultimately well constructed. By taking both chroma and luminance variations into consideration, the new scheme can well model the nonlinear reflectance variations caused by weathering in the resulting appearance manifold.

3.2. Shading decomposition

With the appearance manifold constructed from an input image, we compute the weathering degree map $D(x,y)$ and decompose the shading map $S(x,y)$ from $I(x,y)$ in this step. A straightforward solution is that for each pixel of interest, we find an point v in the appearance manifold whose chroma value is closest to $I_c(x,y)$. We then assign the weathering degree $D(v)$ and weathering component $W(v)$ to this pixel and compute the shading as $S(x,y) = I_l(x,y)/W_l(v)$.

Although this simple scheme theoretically sounds good, it always fails for real images with noise in chroma values (as shown in Fig. 2). In order to analyze the relationship between the weathering degree error and the chroma noise, we formulate the simple scheme as $D(x,y) = F(I_c(x,y))$, where the mapping F is determined by the appearance manifold. Therefore, the error of the weathering degree $\Delta D(x,y)$ caused by chroma noise can be estimated by $\Delta D(x,y) = F'(v) * \Delta I_c(x,y)$, where $F'(v)$ refers to the gradient of the appearance manifold at point v. As a result, the error in weathering degree computation depends on two aspects. On one hand, due to the increase in noise in a RGB image as the pixel brightness decreases [LFSK06], the chroma noise $\Delta I_c(x,y)$ exponentially increases as the luminance decreases. As a result, dark pixels in shadow suffer more chroma noise than other pixels. On the other hand, as shown in Fig. 6(a), the gradient of the appearance manifold would also scale up the chroma noise in the weathering degree computation.

Based on this analysis, we developed a confidence propagation scheme to progressively infer the weathering degree map $D(x,y)$ and reduce the error in weathering degree computation. The key idea of our scheme is to determine the weathering degrees of pixels with high confidence first and then propagate the result outward to help infer the weathering degrees of neighboring pixels with low confidence.

For this purpose, we represent the probability distribution function (PDF) of weathering degrees at each pixel by a discrete histogram, denoted by $Histo(x,y)$. The mean of $Histo(x,y)$ is taken as the estimation of weathering degree $D(x,y)$, while the variance of $Histo(x,y)$ indicates the confidence of $D(x,y)$. The greater variance of $Histo(x,y)$, the lower confidence of $D(x,y)$, and vice versa. To initialize $Histo(x,y)$, we find a point v in the appearance manifold whose chroma value is closest to $I_c(x,y)$. We then randomly sample nearby appearance manifold points with k chroma values coming from a window centered at $I_c(x,y)$ and build the histogram $Histo(x,y)$ with k weathering degrees. The radius of the sampling window centered at $I_c(x,y)$ is deter-

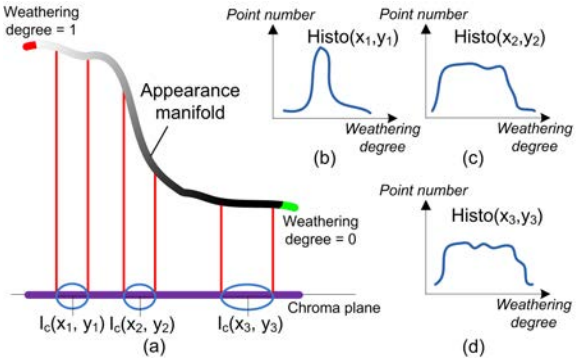

Figure 6: *The impact of appearance manifold gradient in weathering degree computation. (a)Different appearance manifold gradients lead to different errors in weathering degree computation for equal chroma noise. (b)Histogram for $I_c(x_1,y_1)$ has high confidence. (c)Histogram for $I_c(x_2,y_2)$ has low confidence resulting from a large manifold gradient. (d)Histogram for $I_c(x_3,y_3)$ has low confidence due to larger sampling region of darker pixels.*

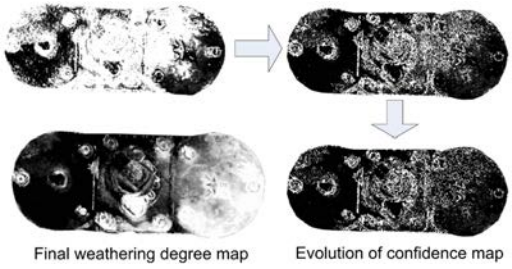

Figure 7: *Confidence propagation. The darker regions in the confidence map represent pixels with higher confidence, while the lighter regions represent pixels with lower confidence. Note that the overall confidences of the weathering degrees increase during the propagation process.*

mined by an decaying exponential function of the pixel luminance $r(x,y) = pe^{-ql_l(x,y)}$. In our implementation, k is set to 500, and p and q are specified so that $r = 1$ for the brightest pixels and $r = 20$ for the darkest pixels. Therefore, $Histo(x,y)$ models the probability distribution of weathering degrees caused by both chroma noise (with different window radius) and the gradient of the appearance manifold, as shown in Fig. 6.

After the initialization, we iteratively update $Histo(x,y)$ by maximizing confidences of all histograms to compute the weathering degree map with the highest overall confidence. In each iteration, we calculate a new function $JointHisto(x,y)$ for each pixel by multiplying all histograms of pixels within a local window centered at the current pixel. Specifically, we first normalize each histogram into a real PDF which sums to 1. Then, we multiply all PDFs and then normalize and discretize the resulting PDF back to a histogram. The window radius is set to two pixels in our implementation. We then replace $Histo(x,y)$ with $JointHisto(x,y)$ if the confidence of $JointHisto(x,y)$ is greater than that of $Histo(x,y)$. Otherwise, we keep $Histo(x,y)$ unchanged. This iteration are executed until all $Histo(x,y)$ converge. Finally, the weathering degree $D(x,y)$ in each pixel is assigned as the estimation of $Histo(x,y)$. Using this confidence propagation scheme, the histograms of pixels with high confidence (low variance) tend to survive and propagate outward. As illustrated in Fig. 7, stable histograms with high confidences is achieved for most pixels after iterations. Weathering degrees of pixels in challenging regions, e.g., shadows, are effectively inferred while the error caused by chroma noise is remarkably alleviated.

With the weathering degree map $D(x,y)$, we then compute the weathering component $W(x,y)$ for each pixel by finding a point v in the appearance manifold whose weathering degree is $D(x,y)$ and its chroma value is closest to $I_c(x,y)$. Finally, we obtain the shading map $S(x,y)$ by computing shading $S(x,y) = I_l(x,y)/W_l(x,y)$ for each pixel.

3.3. Weathering effect editing and rendering

After shading decomposition, the weathering effects distribution in the input image is represented by the weathering degree map $D(x,y)$. To edit the weathering effects, we modify the weathering degree map D(x,y) with a simple function of time t: $D'(x,y) = D(x,y,t) = 1 - e^{-K(x,y)t}$, where $K(x,y)$ is a spatially variant speed factor that is determined by setting $D(x,y,1.0) = D(x,y)$. By adjusting t, we can easily deweather $(0 \le t < 1)$ or weather $(1 < t \le \infty)$ the input image. Besides this way, our system also allows the user to manually generate the desired $D'(x,y)$.

Weathering effect rendering

Given a weathering degree map $D'(x,y)$, we compute the new weathering component $W'(x,y)$ for each pixel. The final de/weathered result $I'(x,y)$ is rendered by combining $W'(x,y)$ with the shading map $S(x,y)$.

As shown in Fig. 9, a naive solution for computing the new weathering component $W'(x,y)$ is to "walk" along the shortest path from its starting point v_{s1} (with weathering degree $D(x,y)$) in the appearance manifold to the most/least weathered point sets. We then find an ending point v'_{e1} with the new weathering degree $D'(x,y)$ and assign its weathering component $W'(v'_{e1})$ to this pixel. However, this simple scheme cannot well preserve the secondary variations of the weathering component. As illustrated in Fig. 8, the rich chroma details tend to be lost in the result, especially for weathering effects with a highly nonlinear appearance manifold.

To solve this problem, we parameterize the chroma varia-

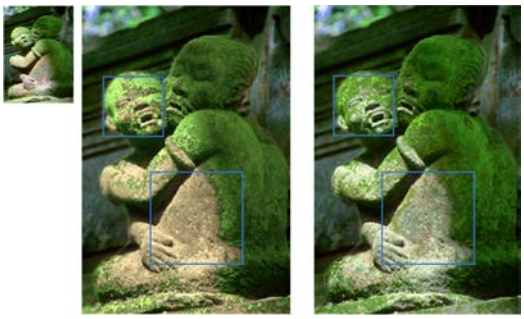

Figure 8: *Comparison between the results generated by shortest path interpolation(middle) and by our pixel walking(right). The significant advantage of our technique in maintaining detailed color variations can be observed.*

tions under weathering degree d by defining the *radial bias ratio* $Rbr(v)$ for all points v with the weathering degree d,

$$Rbr(v) = \phi(v, vc) \,/\, Ra(d), s.t. D(v) = d, \qquad (10)$$

where vc is a point in the appearance manifold that is closest to the geometric center of all points with the weathering degree d, and $\phi(.,.)$ is the geodesic distance between two points in the appearance manifold. $Ra(d) = \max_{\{v, D(v)=d\}} \phi(v, vc)$ is the maximal geodesic distance between point v and vc.

Based on this parameterization, we then present a new pixel-walking scheme for weathering component computation. For each pixel (x,y), starting from its starting point v_{s1} with $D(x,y)$, we find a point v_{e1} in the appearance manifold for the new weathering degree $D'(x,y)$ that satisfies:

$$v_{e1} = arg \min_{v} \{ |Rbr(v) - Rbr(v_{s1})| * \phi(v, v_{s1}) \} \qquad (11)$$

where $D(v) = D'(x,y)$ and the geodesic distance term is used to pick a solution from points with the same radial bias ratio. After that, the weathering component of v_{e1} is assigned to the pixel for rendering. As shown in Fig. 9, the new pixel walking scheme can well preserve the chroma variations in the result by maintaining the radial bias ratio in weathering component computation.

Weathering transfer between images

By parameterizing the appearance manifold with both weathering degree and radial bias ratio, we can build the correspondence between two appearance manifolds and transfer the weathering effects from one image to another. Specifically, given two appearance manifolds M and M', for each point $v' \in M'$, we let $W(v') = W(v) * C$, where $v \in M, D(v) = D(v')$ and $Rbr(v) = Rbr(v')$. $C = S'/S$ is the ratio of image-specific illuminance constants of the two input images. Since the images for weathering transfer in this paper share a similar illuminance condition (sunlight), we set $C = (1,1,1)$ for all weathering transfer results shown in the paper. Fig. 1 and Fig. 10 illustrate two transfer results.

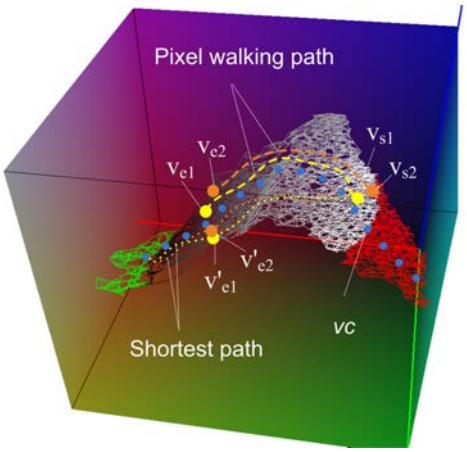

Figure 9: *The pixel walking scheme vs. the shortest path interpolation scheme. With the pixel walking scheme, different starting points v_{s1} and v_{s2} in the appearance manifold will move to v_{e1} and v_{e2}. In contrast, they will move to v'_{e1} and v'_{e2} by simple interpolation along the shortest path in the appearance manifold. The blue curve refers to vc at different weathering degrees. Note that the pixel walking scheme can well preserve the secondary color variations after weathering.*

Figure 10: *The weathering material transfer. Left: The input image of a lamp post with rust. Its weathering degree map is used to guide the material transfer. Middle: The image of the copper statue with patina. Right: The material transfer result. The copper with patina (from the middle image) is transferred to the lamp post.*

4. Experimental results

We implemented our image based material weathering system on a PC with a Pentium IV 3.0GHZ CPU and 2GB memory. The resolutions of all images presented in the paper range from 972×1296 to 2592×1944. For computational efficiency, the appearance manifold is constructed using 2,000 to 4,000 pixels with distinct pixel values, which requires about two minutes of computation in our current implementation. For images in this paper, about fifteen minutes are required for decomposition. A single de/weathered frame is synthesized within one minute.

Fig. 12 and Fig. 13 illustrate two compelling results generated by our approach, where the object geometry and the shading on objects are complicated. Note that in both weathered and de-weathered results, the shading and color details are well preserved.

Fig. 14 exhibits a very challenging case for de/weathering. The tree's complex geometrical details make geometry estimation and shadow removal thoroughly intractable. Nevertheless we produce rather natural looking results. The pixel walking technique demonstrates its advantages for preserving rich details here.

Our approach can be used to generate a consistent weathering image sequence from input image. In the supplementary video, we demonstrate several weathering sequences generated from input images. Another interesting application is to apply weathering editing in an image cut-and-paste system [JSTS06], in which users can adjust the weathering effects of the source object to match the target environment. Fig. 15 shows two results in which a street lamp is pasted into two different scenes and has different weathered appearances that are consistent with the target background.

A failure case of our approach is shown in Fig. 11, where the variation of material geometric details at different rusting stages cannot be well modeled by the appearance manifold. As a result, the shading effects caused by geometric details are decomposed into shading map and are applied to the original region after de-weathering, which lead to artifacts shown in the result.

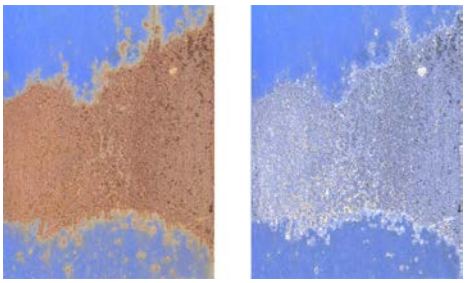

Figure 11: *Left: The input image of a rusting iron surface. Right: the failed de-weathering resulting generated by our method. The geometric variations caused by rusting cannot be correctly modeled by the appearance manifold as weathering components. After de-weathering, it generates artifacts in the result image.*

5. Conclusion

In this paper we propose a system for editing weathering effects of material in a single image. Based on the appearance manifold, our system can successfully decompose the shading from reflectance variations and generate compelling results for a wide scope of weathering effects. Since our approach requires no assumptions about the object geometry in the input image, it could well handle weathering effects on objects with complex geometric details, such as trees. Since our system is based on the appearance manifold, it can only handle weathering effects with progressive reflectance variations. One of our goals in future work is to extend the appearance manifold method to handle time-variant geometry variations caused by weathering, such as the aging of human skin (wrinkles). Modeling and editing the multiple weathering effects exhibited in a single image is another interesting direction for investigation. In addition, we also intend to develop new mathematical tools to represent weathering effects that cannot be modeled by the appearance manifold.

Acknowledgements

The authors thank Stephen Lin for discussions on shading decomposition and proofreading the paper, and the anonymous reviewers for their helpful suggestions and comments. Tianjia Shao helped us to generate some results and demos in the paper and accompanying video. The source images in Fig. 1, 8 and 11 are copied from MaYang's texture library (http://www.mayang.com/textures). We also thank Shuai Yuan, Chao Chen and Xiaoduan Feng for proofreading the paper. Su Xue and Qionghai Dai are supported by the Distinguished Young Scholars of NSFC (No.60525111), the key project of NSFC (No.60432030) and NSFC No. 60721003.

References

[BPMG04] BOSCH C., PUEYO X., MERILLOU S., GHAZANFARPOUR D.: A physically-based model for rendering realistic scratches. *Computer Graphics Forum 23*, 3 (2004), 361–370.

[CXW*05] CHEN Y., XIA L., WONG T.-T., TONG X., BAO H., GUO B., SHUM H.-Y.: Visual simulation of weathering by γ-ton tracing. *ACM Trans. Graph. 24*, 3 (2005), 1127–1133.

[DEJ*99] DORSEY J., EDELMAN A., JENSEN H. W., LEGAKIS J., PEDERSEN H. K.: Modeling and rendering of weathered stone. In *ACM SIGGRAPH '99* (1999), pp. 225–234.

[DGA04] DESBENOIT B., GALIN E., AKKOUCHE S.: Simulating and modeling lichen growth. *Computer Graphics Forum 23*, 3 (2004), 341–350.

[DH96] DORSEY J., HANRAHAN P.: Modeling and rendering of metallic patinas. In *ACM SIGGRAPH '96* (1996), pp. 387–396.

[DPH96] DORSEY J., PEDERSEN H. K., HANRAHAN P.: Flow and changes in appearance. In *ACM SIGGRAPH '96* (1996), pp. 411–420.

[FDL04] FINLAYSON G., DREW M., LU C.: Intrinsic images by entropy minimization. In *ECCV '04* (2004), vol. 3, pp. 582–595.

Figure 12: *The weathering process of a stone lion. On the top-left corner we display the input image.*

[FH04] FANG H., HART J. C.: Textureshop: texture synthesis as a photograph editing tool. *ACM Trans. Graph. 23*, 3 (2004), 354–359.

[FHD02] FINLAYSON G., HORDLEY S., DREW M.: Removing shadows from images. In *ECCV'02* (2002), vol. 4, pp. 823–836.

[GLX*05] GEORGHIADES A. S., LU J., XU C., DORSEY J., RUSHMEIER H.: *Observing and Transferring Material Histories*. Tech. Rep. 1329, Yale University, 2005.

[GTR*06] GU J., TU C.-I., RAMAMOORTHI R., BELHUMEUR P., MATUSIK W., NAYAR S.: Time-varying surface appearance: acquisition, modeling and rendering. *ACM Trans. Graph. 25*, 3 (2006), 762–771.

[HW95] HSU S.-C., WONG T.-T.: Simulating dust accumulation. *IEEE Comput. Graph. Appl. 15*, 1 (1995), 18–22.

[JSTS06] JIA J., SUN J., TANG C.-K., SHUM H.-Y.: Drag-and-drop pasting. *ACM Trans. Graph. 25*, 3 (2006), 631–637.

[KRFB06] KHAN E. A., REINHARD E., FLEMING R. W., BÜLTHOFF H. H.: Image-based material editing. *ACM Trans. Graph. 25*, 3 (2006), 654–663.

[LFSK06] LIU C., FREEMAN W. T., SZELISKI R., KANG S. B.: Noise estimation from a single image. In *CVPR '06* (2006), vol. 1, pp. 901–908.

[LGD*05] LU J., GEORGHIADES A. S., DORSEY J., RUSHMEIER H., XU C.: Synthesis of material drying history: Phenomenon modeling, transferring and rendering. *Euro. Workshop Nat. Phenomena* (2005).

[LGG*07] LU J., GEORGHIADES A. S., GLASER A., WU H., WEI L.-Y., GUO B., DORSEY J., RUSHMEIER H.: Context-aware textures. *ACM Trans. Graph. 26*, 1 (2007), 3.

[MDG01] MERILLOU S., DISCHLER J.-M., GHAZANFARPOUR D.: Corrosion: simulating and rendering. In *Graphics Interface 2001* (2001), pp. 167–174.

[Mil94] MILLER G.: Efficient algorithms for local and global accessibility shading. In *ACM SIGGRAPH '94* (1994), pp. 319–326.

[NB93] NAYAR S. K., BOLLE R. M.: Computing reflectance ratios from an image. *Pattern Recognition 26*, 10 (1993), 1529–1542.

[OCDD01] OH B. M., CHEN M., DORSEY J., DURAND F.: Image-based modeling and photo editing. In *ACM SIGGRAPH '01* (New York, NY, USA, 2001), ACM Press, pp. 433–442.

[PPD02] PAQUETTE E., POULIN P., DRETTAKIS G.: The simulation of paint cracking and peeling. In *Graphics Interface 2002* (2002), pp. 59–68.

[TAF06] TAPPEN M. F., ADELSON E. H., FREEMAN W. T.: Estimating intrinsic component images using non-linear regression. In *CVPR '06* (2006), vol. 2, pp. 1992–1999.

[TFA05] TAPPEN M. F., FREEMAN W. T., ADELSON E. H.: Recovering intrinsic images from a single image. *IEEE Trans. on Pattern Analysis and Machine Intelligence 27*, 9 (Sept. 2005), 1459–1472.

[WTL*06] WANG J., TONG X., LIN S., PAN M., WANG C., BAO H., GUO B., SHUM H.-Y.: Appearance manifolds for modeling time-variant appearance of materials. In *ACM SIGGRAPH '06* (New York, NY, USA, 2006), ACM Press, pp. 754–761.

Figure 13: *De/weathered jar at different stages of weathering. We tune only the middle jar while the other two are kept constant for comparison.*

Figure 14: *A sequence of a de/weathered pine tree with extremely complex geometry and shading. This example illustrates our capacity to handle challenging cases for which geometry estimation and shadow removal are intractable.*

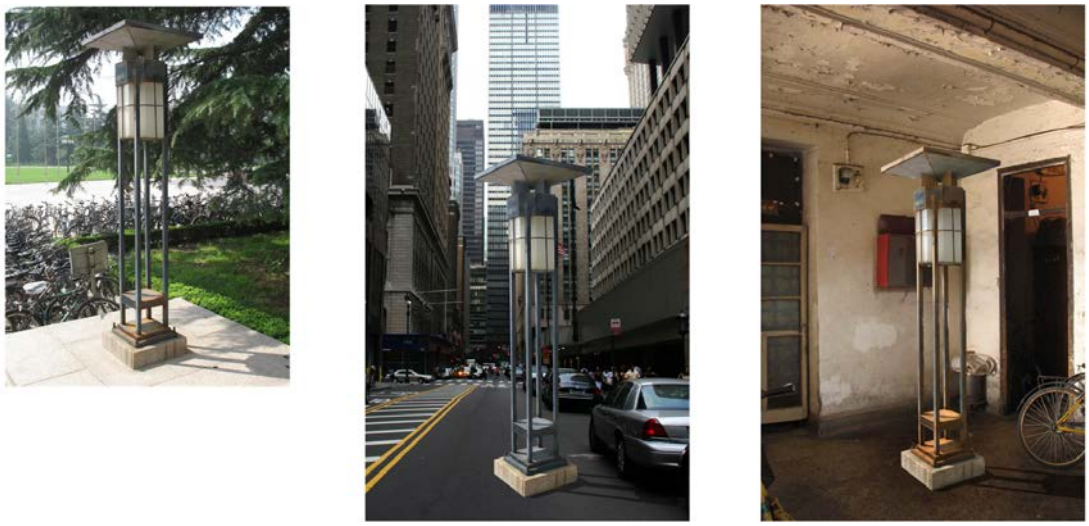

Figure 15: *Incorporating our technique into the cut-and-paste application. We tune the appearance of the source object to fit its stage of weathering harmoniously into the target environment.*