Texture-Preserving Abstraction

David Mould*
Carleton University

Abstract

Image abstraction traditionally eliminates texture, but doing so ignores the more elegant alternative of texture indication, e.g., suggesting the presence of texture through irregular silhouettes and locally chosen details. We propose a variant of geodesic image filtering which preserves the locally strongest edges, leading to preservation of both strong edges and weak edges depending on the surrounding context.

Our contribution is to introduce *cumulative range* geodesic filtering, where the distance in the image plane is lengthened proportional to the color distance. We apply the new filtering scheme to abstraction applications and demonstrate that it has powerful structure-preserving capabilities, especially regarding preservation and indication of textures.

CR Categories: I.3.3 [Computer Graphics]: Display Algorithms— [I.4.3]: Computing Methodologies—Image Processing and Computer Vision-Enhancement Filtering;

Keywords: non-photorealistic rendering, image processing, abstraction, texture, painterly rendering

1 Introduction

Since the beginnings of NPR, the field has been populated with algorithms for well-known artistic styles and media. At the same time, there has been substantial interest in pure abstraction techniques, i.e., methods for creating a version of an input scene with markedly less detail than a conventionally rendered image or a photograph. While early abstraction processes such as that of Haeberli [1990] sought a specifically painterly look with explicit strokes, the more cartoon-like or woodcut-like abstraction postulated by DeCarlo and Santella [2002] has become the norm for NPR abstraction. Image abstraction mechanisms based both on segmentation and on direct filtering sought to remove detail, yielding images characterized by large uniform regions and smooth gradients, free from texture.

Yet while it is often taken for granted that texture should be removed while undertaking image abstraction, it is far from clear that painted or otherwise artistically created images lack texture. Indeed, various NPR algorithms seek to introduce texture into images where it was formerly absent: for example, the watercolorization methods of Bousseau et al. [2006; 2007] introduce high-frequency texture to represent pigment granulation. Various example-based techniques (e.g., Zhao and Zhu's "Sisley" [2010]) introduce extra texture also, seeking to recreate large-scale image structures by placing large numbers of textured strokes. In both of these examples, texture is introduced without reference to the textural properties of the original image. In this work, we attempt to abstract input images not by removing detail entirely, but by using a limited smoothing process that preferentially avoids smoothing across image edges, even very weak edges. Our mechanism preserves the *locally strongest* details, eschewing the flattening effect common in image-based abstraction techniques. The abstracted images are textured if and where the original image was textured. Very small features, whether texture elements or image details, fade into their surroundings without vanishing; sometimes this gives the impression of a 'glazing' effect resembling watercolor.

*e-mail:mould@scs.carleton.ca

Numerous dedicated algorithms for image abstraction have been proposed over the years, some of which we discuss in more detail below. This paper presents a variation on the geodesic filter specifically designed for abstraction of textured images, where the textured regions are abstracted but retain their irregular shapes and ragged edges. The algorithm builds a dedicated mask for each image pixel, taking the nearest n pixels according to a new "cumulative range" variation of geodesic distance. The problem of texture indication has been of longstanding interest to the NPR community, and our mask customization process offers some insight into how texture indication can be achieved.

The contribution of this work is the *cumulative range geodesic filter* (CRGF) and a discussion of some of its properties. The bulk of the paper is devoted to an exposition of the definition and characteristics of the filter and a comparison of its output to that of other generic abstraction methods. Before we discuss the CRGF in detail, however, we first review some of the existing methods for image abstraction.

2 Background

Some work in image abstraction depends on an initial segmentation of the image; for example, DeCarlo and Santella use an automatic hierarchical segmentation [DeCarlo and Santella 2002], and Wen et al. [2006] make use of an interactive segmentation pass following an initial mean shift segmentation [Comaniciu and Meer 2002]. However, since our approach is filter-based, we will chiefly emphasize filter-based approaches in our survey.

The two concepts most relevant to this paper are the bilateral filter [Tomasi and Manduchi 1998] and geodesic filtering [Criminisi et al. 2010]. The bilateral filter involves computing a custom arrangement of weights for each pixel, where the distance of the central pixel to each other pixel is a combination of spatial distance and colorspace distance. Geodesic filtering involves treating a 2D image as a 3D surface and computing distances from a pixel or group of pixels over the manifold [Bai and Sapiro 2007]. Our proposed filter uses a geodesic distance where the incremental horizontal distance from a starting pixel is proportional to the colorspace distance of the current pixel to the original pixel.

The bilateral filter has been used explicitly for abstraction purposes, notably by Winnemöller et al. [2006] whose iterated approximation allowed real-time video processing; Winnemöller et al. also noted texture indication as a goal, though this effort was only partly successful. The domain transform presented by Gastal and Oliviera [2011] allows real-time edge-aware operations; they presented it as an acceleration of bilateral filtering and demonstrated a wide range of effects. Real-time video processing was also an advantage of the geodesic-based formulation presented by Criminisi et al. [2010], one aspect of which was image abstraction. Our naive implementation is far from real time, but we believe that the visual effect we present cannot readily be obtained by other known techniques.

Orzan et al. undertook edge-preserving filtering [Orzan et al. 2007] based on the Poisson equation, with the explicit intention of removing small-scale, weak edges. We share their goal of faithful edge preservation; like ours, their output images also contain gradients, which are suppressed in approaches favoring flattening. However, weak edges make frequent appearances in image texture, and elimination of weak edges is a point of departure between our goals and theirs. Strong edges are generally preserved in the flow-based process of Kang et al. [Kang et al. 2009], in which custom smoothing kernels align to local edge tangents, but weak edges are eliminated and region boundaries are simplified. GradientShop [Bhat et al. 2010] is a versatile system operating in the gradient domain; when employed for abstraction purposes, weak edges and textures (e.g., hair) are abstracted away, replaced by longer-range gradients. A related endeavour was reported by Olsen and Gooch [2011], who used a sequence of linear filters to prepare for the creation of an edge image for subsequent stylization and compression; again, weak edges are eliminated and eventually replaced by gradients in this approach.

Much work in edge-preserving abstraction has been undertaken based on the Kuwahara filter [Papari et al. 2007; Kyprianidis et al. 2009; Kyprianidis and Kang 2011; Kyprianidis 2011], a nonlinear edge-preserving filter. Such work has not sought to preserve texture; in fact, Papari et al. [2007] have the stated goal of eliminating texture. As we argued above, texture removal should not be held out as the sine qua non of abstraction: only certain styles of imagery are texture-free. In the remainder of this paper, we demonstrate a style of abstraction in which the textures are muted and abstracted to a degree but, by design, preserved sufficiently to be communicated to the viewer.

3 Algorithm

Bilateral filtering uses a fixed mask shape with custom weights for each pixel within the mask, depending on the mask's location. In contrast, we propose a filtering process which creates a customized mask shape for each individual pixel but with uniform weights within the mask. The mask consists of the n pixels nearest to the centre, where *nearest* is with respect to a particular distance formulation, explained in detail below, that incorporates aspects of both bilateral and geodesic filters. Edge preservation, even of weak edges, is a natural outcome of our setup.

Let I be an image, whose pixel values are greyscale or color; we refer to "intensity" below without loss of generality, but in practice compute color distances in RGB space. Distances between arbitrary pixels are computed as the infimum of path costs among possible paths connecting them. More formally, using notation adapted from Criminisi et al. [2010], for pixels a and b, the distance $d(a, b)$ is defined as follows:

$$d(a,b) = \inf_{\Gamma \in P(a,b)} \int_0^{l(\Gamma)} C(\Gamma, s)\ \mathrm{d}s \tag{1}$$

where the integration occurs over the arc length s of the path, from 0 to the total path length $l(\Gamma)$. The function $C(\Gamma, s)$ is the infinitesimal cost of proceeding along path Γ at s and is given by

$$C(\Gamma, s) = \left[|I(a) - I(\Gamma(s))| + \gamma |\nabla I(\Gamma(s)) \cdot \Gamma'(s)| \right]. \tag{2}$$

In equation 2 we wrote $I(a)$ to indicate the intensity at the beginning of the path, but could equivalently have written $I(\Gamma(0))$.

In equation 1, $P(a, b)$ refers to the ensemble of paths linking a and b; Γ is one such path, parameterized by arc length s. Note that $\Gamma(s) \in \Re^2$ is a location in the image plane. The image intensity at a location x is given by $I(x)$. The formulation $\nabla I(\Gamma(s)) \cdot \Gamma'(s)$ is the component of the image intensity gradient parallel to the path direction. The parameter γ weights the relative importance of local edge-crossing versus deviation from the original pixel color.

In practice, we do not use equations 1 and 2 directly. Rather, we use a front propagation method to add pixels to the mask one by one. The incremental distance when proceeding from pixel g to pixel h is

$$|I(h) - I_0| + \gamma |I(h) - I(g)| \tag{3}$$

where $I_0 = I(a)$ is the intensity at the first pixel of the path, or equivalently, the intensity at the centre of the mask. The formulation of equation 3 makes obvious the role of the original pixel intensity in controlling the overall region shape.

Once the mask has been computed, we use a box filter over the defined domain to obtain the output color value: formally, for every pixel a, the output $I_{\text{filtered}}(a)$ is given by

$$I_{\text{filtered}}(a) = 1/n \sum_{k \in M_a} I(k). \tag{4}$$

where the mask M contains n pixels, and M_a is the mask customized for pixel a.

Better intuition for the mask shape customization can be gleaned from Figure 1, which shows some example masks (n=180,γ=1). The original pixel is shown with a black circle and the region boundary with a heavy red line. Notice how the mask avoids crossing strong edges (left hand example) and how the mask can become very irregular in order to conform to highly irregular structures in the image (right hand example).

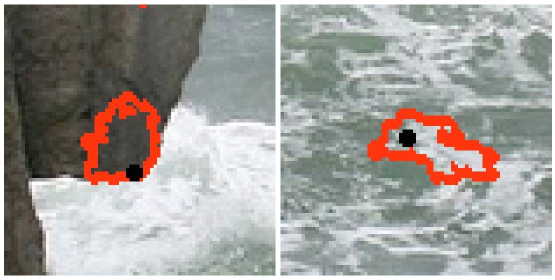

Figure 1: *Irregular masks.*

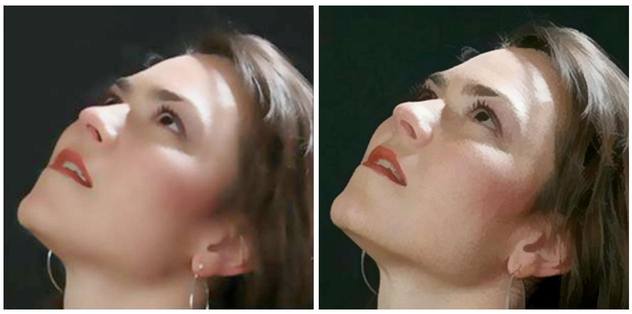

Figure 2: *Comparison with geodesic flattening. Left: geodesic output; right: our result. Former image provided by Criminisi et al. [2010].*

Figure 3: *Some large-scale results. Clockwise from upper left: stranger; cliff; forked tree; ranch; autumn.*

Figure 2 shows a visual comparison to geodesic flattening. (The original image is included as part of Figure 4.) Our result better preserves texture, as intended; pay particular attention to the hair above and behind the ear, which has been significantly blurred by geodesic flattening but is more gracefully abstracted with our technique. While the lack of smoothing of texture over the face may not be desirable in this instance, the texture is indeed present in the original, highlighting that the abstraction style chosen must be consonant with the user's aims.

In the following, we demonstrate the practical effect of the filter by applying it to several test images and showing comparisons to existing abstraction methods.

4 Results and Discussion

4.1 Examples

Several examples can be seen in Figure 3. The original images for our examples and comparisons can be seen in Figure 4. Since appreciating the results depends on careful attention to small-scale details, we recommend that these images be viewed on screen and at a high zoom level.

We chose sample images that contain large textured regions. The different types of texture are generally recognizable in the abstracted images. In the cliff image, perserving the texture is especially important since it offers a natural way to depict the motion of the water, apparent even in this still image. Areas which lack texture, such as the house in the ranch image, are more conventionally flattened. We devote the remainder of this subsection to discussing specific details of some of these images.

Figure 5 shows the basic texture abstraction and edge preservation properties of the algorithm. In the top image pair, notice how the texture is muted without being obliterated: the dynamic range has been locally compressed, but the details are still subtly present. In particular, irregular structures remain visible: e.g., the foam boundaries on the upper right. In the middle pair, the corrugated bark texture is still apparent, and the silhouette of the tree has been preserved. The bottom image has been selected to demonstrate strong edge preservation: the complex silhouette of the fabric is unchanged while the higher-frequency details within the body of the cap are abstracted.

Thin but extended linear features can be difficult to preserve using other approaches; segmentation-based approaches find them particularly challenging. However, our method quite naturally maintains linear features. As long as enough similar-colored pixels lie in a contiguous region, regardless of shape, the masks can stretch to collect them. Figure 6 illustrates preservation of linear features. While the masses of leaves and the main part of the beard are abstracted, individual tree branches and hairs remain visible, hinting at the overall structure without portraying it explicitly: this is at the heart of texture indication. For a non-texture example of linear feature preservation, look at the whiskers of the cat in Figure 12. The yellow leaves in Figure 6, combined with the slight paling of

Figure 4: *Original images for the abstractions used throughout this paper.*

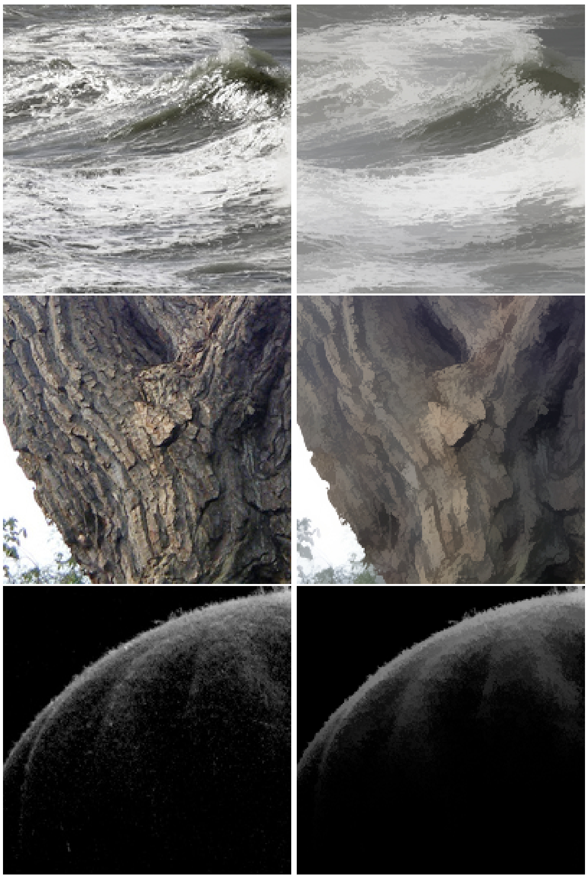

Figure 5: *Texture abstraction and edge preservation. Top: wave detail from cliff image; middle: bark detail from forked tree image; bottom: detail of cap from stranger image. Left: originals; right: filtered.*

Figure 6: *Linear feature preservation. Above: detail from autumn image; below: detail of stranger's beard. Left: originals; right: filtered.*

the smaller branches, provides a striking example of the watercolor 'glazing' effect alluded to earlier. The leaves of Figure 7 provide another example. Edge-preserving fading is discussed in more detail in the folowing subsection.

Figure 7: *Subtle texture edges and irregular regions.*

While Figure 5 has examples of preserving strong edges, we are also interested in *weak* edge preservation. Because of the custom mask shape, even weak edges can be preserved if they are locally the strongest. This is sometimes a subtle effect, but it is apparent in Figure 7, especially the spray near the cliff in the middle image. The water contains many low-intensity edges that are nonetheless maintained in the abstracted output. A larger-scale example of this is the shadow of the cliff, seen in full in Figure 3. Although the intensity of the shadow edge varies considerably from place to place, it always represents a sufficient change in contrast that the masks rarely straddle it. Figure 7 also contains further examples of irregular texture regions in the water and foliage. Particularly note the muted irregular structures in the abstracted mass of leaves, indicating the details but leaving it to the mind of the viewer to fill them in. The ragged silhouette helps considerably in furthering the illusion of detail.

4.2 Variations

Figure 8 shows the effect of changing the filter size n. Unsurprisingly, larger masks produce more abstracted images: features of size larger than the mask can straightforwardly be preserved, while smaller features begin to disappear. However, the small features do not become blurred, but instead gradually fade. This effect is most apparent in the dark rock in the upper right of Figure 8.

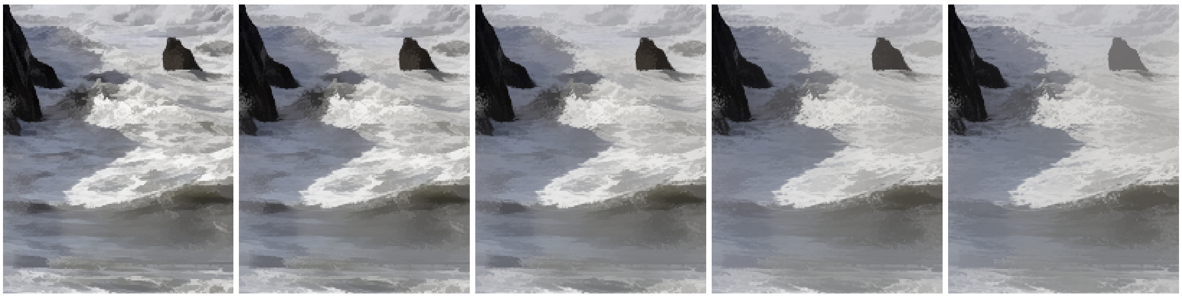

Figure 8: *Effect of changing size of region. Left to right: region sizes 40, 80, 160, 480, and 800 pixels.*

Figure 9: *Effect of changing γ. Left to right: original, γ=1, γ=8, γ=64.*

The fading phenomenon is due to the asymmetry inherent in our formulation. If pixel b is part of the mask for pixel a, there is no guarantee that pixel a will belong to b's mask. In fact, in general $d(a, b) \neq d(b, a)$. In the context of the black rock, the behavior manifests as follows: when possible, the black pixels form masks which occupy only black pixels, but when the mask size is larger than the rock, part of the mask necessarily extends into the water, lightening the box filter output. Conversely, the lighter pixels near the rock have a huge area of light-colored pixels in which to form their masks, and they never need to extend into the black rock. Thus, though the rock fades, it has little influence on the color of the water pixels, preserving the shape of its outline.

Figure 9 shows the effect of changing γ. With low γ, the deviation from the origin dominates, and we have high detail preservation. With high γ, the mask tends to extend into nearby smooth regions regardless of whether they are similar to the original pixel's color, so detail tends to be lost. In general we prefer the look of low γ, but for specific images or more aggressive detail removal, higher γ may be preferred. Unless otherwise stated, all images in this paper were created with γ=1.

4.3 Comparisons

In the following, we provide visual comparisons to some recent and relevant abstraction techniques: structure-preserving photo manipulation [Orzan et al. 2007]; the multiscale Kuwahara filter [Kyprianidis 2011]; and bilateral filtering [Tomasi and Manduchi 1998]. We previously gave a comparison to geodesic flattening [Criminisi et al. 2010], in section 3. Original images for the results in this section are part of Figure 4.

Figure 10 compares our filter with structure-preserving photo manipulation [Orzan et al. 2007]. Like us, Orzan et al. intend to faithfully preserve strong edges; the two methods are approximately equally successful at this. However, Orzan et al. deliberately seek to eliminate weaker edges, with the consequence that the detail on the petals, the texture on the fruit, and even an entire faint leaf (above the upper white flower) are largely removed. These textures are not particularly well defined in the original image, yet our method somewhat suggests them. A subtle point is the texture on the object behind the fruit: entirely smoothed by photo abstraction, nonetheless our result conveys a delicate sense of half-glimpsed surface detail.

Figure 10: *Comparison with structure-preserving photo manipulation. Left: result from Orzan et al. [2007]; right: our result.*

Figure 11 shows our attempt to deal with an especially difficult image, presented by Kyprianidis [2011] as a failure case for multiscale Kuwahara filtering. While we agree that this is a challenging image and our result is not completely clear, we are able to maintain some separation between the bush and the middle ground and avoid the overblurring visible in the output of the Kuwahara filter. The textures on the distant mountains have been abstracted nicely.

Figure 12 compares a successful outcome from Kuwahara filtering to our approach. We have also included the bilateral filtering result in this figure. Compared to both single-scale and, especially, multiscale anisotropic Kuwahara filtering, our result better suggests the underlying texture. Our filter flattens the image but the resulting mostly-uniform regions have irregular boundaries and the silhouettes are ragged, better conveying a sense of the fur. The whiskers are also better preserved, particularly in the image's upper

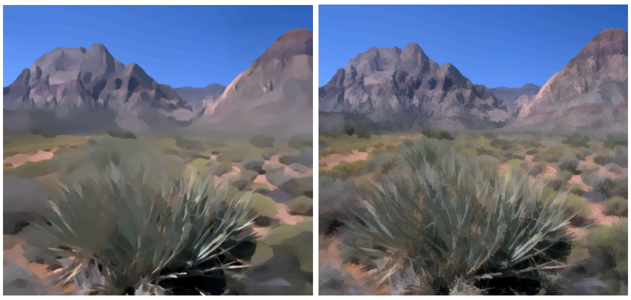

Figure 11: *Comparison with multiscale Kuwahara filter. Left: Kuwahara filtering; right: our result. Former result provided by Kyprianidis [2011].*

Figure 12: *Comparison with multiscale Kuwahara filter. Above: bilateral filter; anisotropic Kuwahara filtering. Below: multiscale anisotropic Kuwahara filtering; proposed filter. The first three images are provided by Kyprianidis [2011].*

left where the contrast with the cat's fur is relatively low.

With respect to the bilateral filter, our result appears more muted, having removed small high-contrast elements such as the black spots around the cat's nose and mouth. Similarly, contrast has been reduced in the fur across the cat's forehead. At the same time, our texture detail is in many places superior to the result from the bilateral filter: for example, subtle distinctions have been maintained on the sides of the cat's face. This attests to the difficulty of using a single parameter setting for the bilateral filter to preserve edges of different scales.

This difficulty is further illustrated in Figure 13, which compares the proposed filter and the bilateral filter. The original synthetic image is a series of ideal vertical step edges corrupted with uniform noise. We show a horizontal cross-section; the original data is in light grey. With the settings used, the bilateral filter recovers large intensity discontinuities perfectly, but smaller edges are smeared out. In general, the bilateral filter offers a tradeoff between preserving edges and preserving noise or fine detail. Conversely, the proposed filter has better edge localization for the smaller edges; while to some extent it treats the noise as edges to be preserved, the noise pixels are incoherent and its ability to preserve them is limited by the unavailability of sufficient numbers of pixels of similar color: i.e., incoherent noise is always attenuated. Note that while the noise removal applications of the proposed filter may be limited, we are not proposing the method for noise removal proper, but rather abstraction; the small-scale intensity changes that are best preserved are those which are coherent over a boundary size large enough to enclose the mask. Coherent intensity changes are exactly those which we do want to preserve.

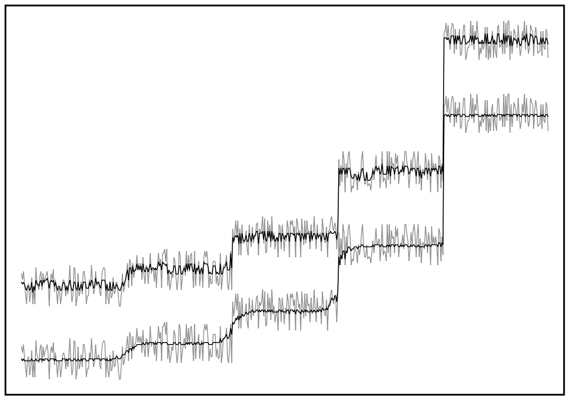

Figure 13: *Response to synthetic data. Above: proposed filter; below: bilateral filter. Original noisy data shown in light grey.*

4.4 Limitations

The proposed filter has one main limitation, which is its speed. Bilateral and geodesic filtering have benefitted from recent advances which make them extremely fast, but our unoptimized single-threaded prototype implementation is orders of magnitude slower: the speed is $O(kn)$ for an n-pixel image with mask size k. For typical images in this paper (roughly one megapixel), with $k = 160$, our processing time is approximately 30 seconds.

The property of weak edge preservation is beneficial when weak edges represent real structure, but the algorithm also preserves noise to some extent. In practice this would be ameliorated by lightly preprocessing the input with another filter.

Finally, not all textures can be adequately preserved using the proposed approach. Very high frequency textures still tend to be suppressed. Figure 14 shows an example: the noise-like texture of the sand in the foreground is almost entirely removed, leaving only the shapes and shadows of the driftwood in the middle ground. This is quite like a traditional image abstraction result, but we consider it a failure case for our method given our objective of texture preservation.

5 Conclusion

We have presented a novel variant of geodesic filtering, in which horizontal distance over the image manifold is locally stretched by the range distance to the origin. We made use of this distance to build custom masks; box filtering over such masks yields a texture-preserving abstraction effect. This is an effect rarely seen in past abstractions, which mainly concentrated on flattening the image and removing texture.

We showed images demonstrating the properties of the filter: its adherence to strong edges such as silhouettes; its ability to convey weak and irregular edges; its preservation of extended linear

Figure 14: *Left: original; right: filtered. The sand texture is not preserved.*

features; and its attenuation of isolated small features. A rarity in having been designed for texture abstraction, this method produces images visually distinct from those of other methods.

The main drawback to the proposed approach is its slow speed, so the obvious goal in future work is to address this. The existing approach can straightforwardly be parallelized to take advantage of multiple cores, and we would also like to investigate a GPU implementation and alternatives to naive front propagation. Applying the approach to video is another obvious direction and would be made more feasible by improving the speed.

Future work also includes creating an adaptive mask size, exploring a multiscale version of the proposed filter, and using the technique for texture and edge enhancement as well as abstraction. Finally, we would like to consider other distance functions to extend the range of stylization effects achievable within the geodesic framework.

Acknowledgements

Thanks to Hua Li and Herb Yang for suggestions as this work was being developed. Also thanks to the photographers who contributed images through Flickr: jocelyn (stranger), Ian Carroll (cliff), Ben Sollis (autumn), and Al_HikesAZ (ranch). This work was supported by NSERC and by Carleton University.

References

BAI, X., AND SAPIRO, G. 2007. A geodesic framework for fast interactive image and video segmentation and matting. In *ICCV*, 1–8.

BHAT, P., ZITNICK, C. L., COHEN, M., AND CURLESS, B. 2010. Gradientshop: A gradient-domain optimization framework for image and video filtering. *ACM Trans. Graph. 29*, 2 (Apr.), 10:1–10:14.

BOUSSEAU, A., KAPLAN, M., THOLLOT, J., AND SILLION, F. 2006. Interactive watercolor rendering with temporal coherence and abstraction. In *International Symposium on Non-Photorealistic Animation and Rendering (NPAR)*, ACM.

BOUSSEAU, A., NEYRET, F., THOLLOT, J., AND SALESIN, D. 2007. Video watercolorization using bidirectional texture advection. *ACM Transaction on Graphics (Proceedings of SIGGRAPH 2007) 26*, 3.

COMANICIU, D., AND MEER, P. 2002. Mean shift: A robust approach toward feature space analysis. *IEEE Trans. Pattern Anal. Mach. Intell. 24*, 5, 603–619.

CRIMINISI, A., SHARP, T., ROTHER, C., AND PÉREZ, P. 2010. Geodesic image and video editing. *ACM Trans. Graph. 29*, 5 (Nov.), 134:1–134:15.

DECARLO, D., AND SANTELLA, A. 2002. Stylization and abstraction of photographs. *ACM Trans. Graph. 21* (July), 769–776.

GASTAL, E. S. L., AND OLIVEIRA, M. M. 2011. Domain transform for edge-aware image and video processing. *ACM Trans. Graph. 30*, 4 (Aug.), 69:1–69:12.

HAEBERLI, P. 1990. Paint by numbers: abstract image representations. In *Proceedings of SIGGRAPH*, ACM, SIGGRAPH '90, 207–214.

KANG, H., LEE, S., AND CHUI, C. K. 2009. Flow-based image abstraction. *IEEE Trans. Vis. Comput. Graph. 15*, 1, 62–76.

KYPRIANIDIS, J. E., AND KANG, H. 2011. Image and video abstraction by coherence-enhancing filtering. *Computer Graphics Forum 30*, 2, 593–602. Proceedings of Eurographics 2011.

KYPRIANIDIS, J. E., KANG, H., AND DÖLLNER, J. 2009. Image and video abstraction by anisotropic Kuwahara filtering. *Computer Graphics Forum 28*, 7, 1955–1963. Special issue on Pacific Graphics 2009.

KYPRIANIDIS, J. E. 2011. Image and video abstraction by multi-scale anisotropic Kuwahara filtering. In *Proceedings of the Symposium on Non-Photorealistic Animation and Rendering*, ACM, NPAR '11, 55–64.

OLSEN, S., AND GOOCH, B. 2011. Image simplification and vectorization. In *Proceedings of the Symposium on Non-Photorealistic Animation and Rendering*, ACM, NPAR '11, 65–74.

ORZAN, A., BOUSSEAU, A., BARLA, P., AND THOLLOT, J. 2007. Structure-preserving manipulation of photographs. In *Proceedings of the Symposium on Non-photorealistic Animation and Rendering*, ACM, NPAR '07, 103–110.

PAPARI, G., PETKOV, N., AND CAMPISI, P. 2007. Artistic edge and corner enhancing smoothing. *IEEE Transactions on Image Processing 16*, 10, 2449–2662.

TOMASI, C., AND MANDUCHI, R. 1998. Bilateral filtering for gray and color images. In *Proceedings of the Sixth International Conference on Computer Vision*, IEEE Computer Society, Washington, DC, USA, ICCV '98, 839–846.

WEN, F., LUAN, Q., LIANG, L., XU, Y.-Q., AND SHUM, H.-Y. 2006. Color sketch generation. In *Proceedings of the Symposium on Non-photorealistic animation and rendering*, ACM, NPAR '06, 47–54.

WINNEMÖLLER, H., OLSEN, S. C., AND GOOCH, B. 2006. Real-time video abstraction. *ACM Trans. Graph. 25* (July), 1221–1226.

ZHAO, M., AND ZHU, S.-C. 2010. Sisley the abstract painter. In *Proceedings of the 8th International Symposium on Non-Photorealistic Animation and Rendering*, ACM, NPAR '10, 99–107.