HAL
open science

A Dynamic Noise Primitive for Coherent Stylization

Pierre Bénard, Ares Lagae, Peter Vangorp, Sylvain Lefebvre, George Drettakis, Joëlle Thollot

To cite this version:

Pierre Bénard, Ares Lagae, Peter Vangorp, Sylvain Lefebvre, George Drettakis, et al.. A Dynamic Noise Primitive for Coherent Stylization. Computer Graphics Forum, 2010, Proceedings of the Eurographics Symposium on Rendering 2010, 29 (4), pp.1497-1506. 10.1111/j.1467-8659.2010.01747.x . inria-00487986v2

HAL Id: inria-00487986

https://inria.hal.science/inria-00487986v2

Submitted on 15 Jul 2010 (v2), last revised 6 Aug 2011 (v4)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Eurographics Symposium on Rendering 2010
Jason Lawrence and Marc Stamminger
(Guest Editors)

Volume 29 (2010), Number 4

A Dynamic Noise Primitive for Coherent Stylization

P. Bénard[1] A. Lagae[2,3] P. Vangorp[3] S. Lefebvre[4] G. Drettakis[3] J. Thollot[1]

[1]Grenoble University [2]Katholieke Universiteit Leuven [3]INRIA Sophia-Antipolis [4]INRIA Nancy Grand-Est / Loria

Abstract

We present a new solution for temporal coherence in non-photorealistic rendering (NPR) of animations. Given the conflicting goals of preserving the 2D aspect of the style and the 3D scene motion, any such solution is a trade-off. We observe that primitive-based methods in NPR can be seen as texture-based methods when using large numbers of primitives, leading to our key insight, namely that this process is similar to sparse convolution noise in procedural texturing. Consequently, we present a new primitive for NPR based on Gabor noise, that preserves the 2D aspect of noise, conveys the 3D motion of the scene, and is temporally continuous. We can thus use standard techniques from procedural texturing to create various styles, which we show for interactive NPR applications. We also present a user study to evaluate this and existing solutions, and to provide more insight in the trade-off implied by temporal coherence. The results of the study indicate that maintaining coherent motion is important, but also that our new solution provides a good compromise between the 2D aspect of the style and 3D motion.

Categories and Subject Descriptors (according to ACM CCS): I.3.3 [Computer Graphics]: Picture/Image Generation—

1. Introduction

All stylized animation algorithms are confronted with the temporal coherence problem: the fundamental contradiction between conveying the 2D impression of the style depicted, while simultaneously following the 3D motion of the underlying scene. In this work, we address the case of region filling, where an animation is stylized using what we call a *pattern*, i.e., watercolor pigments, set of strokes, paper grain or any medium an artist may want to simulate.

We define three goals, inspired by previous work [Mei96, CTP*03, BSM*07], that must be achieved simultaneously for successful temporal coherent stylization:

Flatness by which we mean the ability to convey the 2D aspect of a given stylization. As pointed out by Meier [Mei96], the goal is to give the impression that each image is produced on a flat medium rather than being painted on a 3D object using surface *texture mapping*. 2D properties of the pattern must be respected: size of the features, their 2D distribution, homogeneous contrast or density. Breslav *et al.* [BSM*07] emphasized the popularity of these 2D patterns in traditional imagery because they provide greater abstraction.

Coherent motion represents the high correlation between the 3D scene motion field and the pattern motion. As stated by Cunzi *et al.* [CTP*03], the goal is to provide in 2D screen space a perceptual impression of motion as close as possible to the 3D displacement in object space. In contrast, when the pattern is static in screen space, the *shower door* effect [Mei96] occurs: the pattern seems to slide over the scene.

Temporal continuity is the quality of minimizing changes from frame to frame to ensure fluid animations. Perceptual studies [YJ84, SS09] have shown that human observers are very sensitive to sudden temporal incoherence such as popping. The goal is to keep the stylization as continuous as possible during the animation. This is especially important when zooming and at occlusion/disocclusion boundaries.

The conflicting nature of these goals means that any method addressing the problem of temporal coherence is *necessarily a compromise*. Any such method will be unable to achieve all the above goals perfectly, and thus will necessarily incur some artifacts, such as pattern deformation, sliding or popping. Previous approaches are based on *primitives* or *textures*; each solution presents a different compromise.

We present two contributions in this paper:

- First, we propose a noise primitive that provides a trade-off for the temporally coherent stylization of any

© 2010 The Author(s)
Journal compilation © 2010 The Eurographics Association and Blackwell Publishing Ltd.
Published by Blackwell Publishing, 9600 Garsington Road, Oxford OX4 2DQ, UK and 350 Main Street, Malden, MA 02148, USA.

PREPRINT

kind of 3D animation including deformable objects. By merging a large number of such primitives we produce a temporally coherent noise that exhibits the benefit of both primitive-based and texture-based approaches. This noise is then used in the same way as traditional procedural texture approaches to produce a wide variety of results with styles ranging from continuous (watercolor, ink) to discrete (hatching, painterly), all in the same framework.

- Second, we present a user study which evaluates the success of various solutions according to the three goals outlined above. The design of this study is based on a careful analysis of the trade-offs of different methods. The results of the study indicate that coherent motion is important and that our solution is a good compromise, providing a strong sense of flatness, and a reasonable trade-off for the other goals.

2. Related work

We classify related work into two categories: primitive-based methods and texture-based methods. We also identify a third category, many-primitive methods, establishing a connection between primitive-based methods in NPR and methods for noise in procedural texturing.

2.1. Primitive-based methods

Primitive-based methods are mostly used for patterns composed of uniformly distributed features, typically painterly rendering and stippling.

Meier [Mei96] introduced the idea of attaching anchor points to the 3D surface and using them to draw overlapping primitives in 2D. Since the primitives are usually small with respect to object size, their motion remains very close to the original motion field of the 3D scene. However, no control is provided over their 2D distribution which may produce stroke cluttering. To overcome this limitation, Daniels [Dan99] gave the control to the user who manually draws the strokes on the objects. This time-consuming process can be avoided by using dynamic distributions. Several authors generate such distributions automatically either by extending Meier's work, e.g. [CL06, VBTS07], or using the optical flow of an input video, e.g. [HE04]. Their main goal is to ensure a Poisson disk distribution in 2D, while still following the 3D scene motion. Kaplan *et al.* [KGC00] extended this range of styles by relaxing the distribution constraint and giving back more control to the user.

For such approaches the typical trade-off between flatness and temporal continuity involves either a dynamic distribution at the price of some popping (e.g. [VBTS07]), or no popping but some clutter artifacts or holes (e.g. [Mei96]).

2.2. Texture-based methods

Texture-based approaches are mostly used for continuous textures (canvas, watercolor) or highly structured patterns (hatching). Textures naturally ensure spatial continuity of the pattern, which allows for simpler and faster methods than primitive-based approaches in general.

Object-space approaches compromise flatness in favor of coherent motion by relying on texture mapping. To reduce perspective distortions and scale variations, these methods try to adapt the texture to maintain a near-constant size of the pattern in screen space. A specific set of mipmaps can be used to achieve this goal, such as art maps [KLK*00] or tonal art maps [PHWF01]. It leads to smooth transitions but global oscillations in terms of sharpness when only two mipmap levels are blended. Texture fractalization [BBT09] achieves a very good texture continuity, but also has a tendency to degrade the original pattern [BTS09].

Image-space approaches compromise motion coherence in favor of flatness by directly rendering the texture in screen space and modifying it according to the scene motion. Bousseau *et al.* [BNTS07] locally deform the texture according to the optical flow of a video. This creates a good compromise for continuous media but introduces contrast oscillations and degrades structured patterns. Cunzi *et al.* [CTP*03] approximate the scene motion by a global 2D transformation of the texture, creating a strong sliding artifact. This artifact is reduced but still visible in [CDH06, BSM*07] where they cut the texture in a small number of patches transformed independently. Again, these methods rely on mipmaps or fractalization to obtain smooth transitions when zooming.

2.3. Many-primitive methods

Primitive-based methods in NPR typically use relatively few primitives, in which case individual primitives can be discriminated and popping is noticeable. Our key insight is that when using many primitives, individual primitives merge and form a texture, which allows to fade primitives in or out with much less noticeable popping, and these methods become highly related to sparse convolution noise.

In procedural texturing, sparse convolution noise [Lew84, Lew89], including spot noise [vW91] and Gabor noise [LLDD09], can be classified as many-primitive methods. For an overview of procedural noise functions, see Lagae et al. [LLC*10]. These methods allow to create a wide variety of textures using procedural texturing [EMP*02] but do not address the problem of temporal coherence. In NPR, the only two previous methods that can be classified as many-primitive methods are the dynamic canvas of Kaplan and Cohen [KC05] and the interactive watercolor rendering of Bousseau *et al.* [BKTS06]. These methods take into account the problem of temporal coherence, but are limited to a specific medium: canvas fibers and watercolor respectively. In both cases the connection with sparse convolution noise is not made explicit, even though Bousseau *et al.* use Perlin noise [Per85].

Our dynamic noise primitive for coherent stylization ad-

© 2010 The Author(s)
Journal compilation © 2010 The Eurographics Association and Blackwell Publishing Ltd.

PREPRINT

dresses the problem of temporal coherence and leverages procedural texturing to create a wide variety of styles. To further extend the range of patterns we can produce, we plan to construct a version of spot noise that takes into account temporal coherency. As a many-primitive method, NPR Gabor noise bridges primitive-based methods and texture based methods. As we shall show in our results and user experiment, it features good motion coherence, similar to few-primitive methods, and a good sense of flatness, similar to image-space texture-based methods.

3. A noise primitive for NPR

In procedural texturing, a wide variety of textures is created using a noise primitive (typically Perlin noise) as the basic component [Per85, EMP*02]. To fulfill the goals we determined in the introduction, we define a noise primitive for NPR that is temporally coherent. Our new noise primitive is based on Gabor noise [LLDD09, LLD09]. We chose Gabor noise because it features precise local spectral control, which allows us to maintain the 2D aspect of the noise, and to create a wide variety of different styles (see Sec. 4).

3.1. Gabor noise

Gabor noise n is a sum of randomly weighted and positioned Gabor kernels,

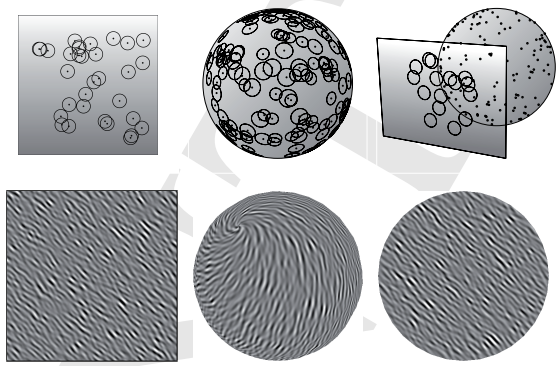
$$n(x,y) = \sum_i w_i g(a_i, F_{0i}, \omega_{0i}; x - x_i, y - y_i), \qquad (1)$$

where $\{w_i\}$ are the random weights, g is the Gabor kernel, and $\{(x_i, y_i)\}$ are the random positions. The Gabor kernel is parametrized by a bandwidth a, which determines the kernel radius r, a frequency F_0 and an orientation ω_0. The random positions $\{(x_i, y_i)\}$ are distributed according to a Poisson distribution with mean λ. We call the random positions the point distribution, and λ the point density. Depending on how a_i, F_{0i} and ω_{0i} vary for different kernels, anisotropic, isotropic, or even more general kinds of noise are obtained.

The number of overlapping kernels is relatively large. The mean λ is expressed as $N'/\pi r^2$, where N' is the average number of overlapping kernels. The choice of N' is a speed versus quality trade-off, since both the evaluation time and the quality of the noise are proportional to N'. Typical values of N' range from 16 to 128.

3.2. NPR Gabor noise

None of the existing kinds of Gabor noise addresses the problem of temporal coherence for NPR. 2D Gabor noise (see Fig. 1(a)) is inherently 2D, and therefore behaves like the *shower door* effect, while surface Gabor noise (see Fig. 1(b)) and solid Gabor noise are inherently 3D, and therefore behave like regular texture mapping. We present a new kind of Gabor noise, which we call NPR Gabor noise (see Fig. 1(c)), that takes into account the problem of temporal coherence.



(a) 2D (b) Surface (c) NPR

Figure 1: NPR Gabor noise. (a) 2D Gabor noise. (b) Surface Gabor noise. (c) NPR Gabor noise. NPR Gabor noise has a 2D aspect, similar to 2D Gabor noise, and moves coherently with the object (see video), similar to surface Gabor noise.

Gabor noise can be seen as a primitive-based method, where the kernel is the primitive. As for other primitive-based methods in NPR, the basic principles of NPR Gabor noise follow from the goals formulated in Sec. 1:

- To obtain a noise with a 2D aspect, we define the noise parameters F_0, a, ω_0 and λ in 2D screen space, and also evaluate the noise in 2D screen space.
- To ensure a coherent motion of the noise, we define the point distribution used by the noise on the surface of the 3D model.

Primitive-based methods in NPR typically use splatting [KC05, BKTS06]. Noise based on kernels can be generated using splatting, as in [vW91, FW07], or procedurally, as in [Lew89, LLDD09], two conceptually opposite approaches. Procedural approaches are usually more general, while splatting is usually faster and maps better to the GPU. Since we target interactive applications, we use splatting.

3.3. Dynamic point distribution

Assuming a triangle model and a perspective camera, we first need to generate a Poisson distribution for our Gabor noise. We then need a way to guarantee temporal continuity of the noise. To achieve this while minimizing popping, we introduce an LOD mechanism which ensures a continuous and constant density of point distributions in 2D screen space throughout time. A key aspect of this LOD is the blending weight, which we determine based on the statistical properties of the noise.

Poisson distribution for NPR Gabor noise. In contrast to most few-primitive methods in NPR, which require a Poisson-disk distribution, Gabor noise only requires a Poisson distribution. We obtain this distribution per triangle using a method inspired by [KC05] and [LLDD09]. We generate the points in a triangle using a pseudo-random number

© 2010 The Author(s)
Journal compilation © 2010 The Eurographics Association and Blackwell Publishing Ltd.

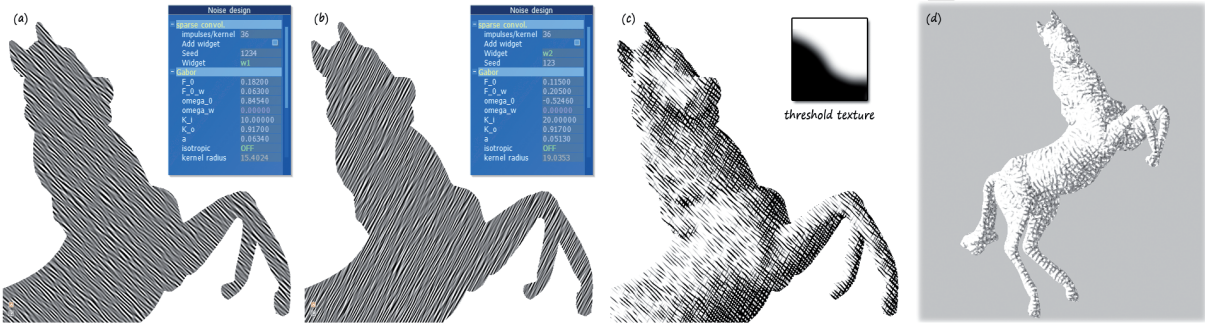


Figure 2: Style design. (a, b) Two anisotropic noises are designed. (c) The thresholded results are multiplied to produce a cross-hatching. The inset shows the x-toon texture used for thresholding. (d) Adding a third high-frequency noise layer allows to produce a graphite style.

generator (PRNG), with a different seed for each triangle to ensure a random point distribution. We use the PRNG to generate uniformly distributed barycentric coordinates that correspond to the random kernel positions $\{(x_i, y_i)\}$. Since points in a Poisson distribution are independent, these positions are guaranteed to follow a Poisson distribution over edges, even if they are generated per-triangle. We always use the same seed for the same triangle, which ensures temporal continuity during the animation. If the expected number of points N on the triangles is less than 1, both generating a point or not will result in an incorrect point density. We solve this problem by generating a point and weighting it by N to account for this discretization error.

Dynamic point distribution using LOD. Since the density of the point distribution is constant in 2D screen space, the number of points on a triangle in 3D object space should vary. More specifically, the required number of points N on a triangle is λA, where A is the 2D screen space area of the triangle. Note that N changes when A changes, for example, when the triangle moves away from the viewer. This means that NPR Gabor noise requires a dynamic point distribution, i.e., a distribution that changes over time. One way to achieve this would be to add and remove individual points in a temporally consistent manner, to maintain the required point density λ in 2D screen space, similar to [KC05, BKTS06]. Unfortunately, we have experienced that, with our noise, this fine-grained level of control can still lead to popping, since a single kernel can still appear relatively fast.

Instead, we halve or double the number of points, and weight the second half of the points to account for this operation. More specifically, rather than using N points, we use $2M$ points, where:

$$M = 2^{\lfloor \log_2 N \rfloor} \tag{2}$$

and we weigh the first M points with a weight of 1, and the second M points with a weight of α. Note that M and $2M$ are the powers of 2 that bracket N. This coarse-grained level of control further reduces popping, since kernels now appear at a slower rate.

Choice of blending weight using statistical properties. Since Gabor noise is a summation, we can express the noise n_{2M} based on $2M$ points as a linear interpolation between two noises n_M and n'_M based on M points:

$$n_{2M}(x,y) = n_M(x,y) + \alpha n'_M(x,y). \tag{3}$$

Therefore, it might seem as if we just trade popping artifacts for blending artifacts, in a manner similar to texture-based methods in NPR. However, this is not the case, since we do not affect the appearance of the noise. Our approach preserves the statistical properties of Gabor noise because we blend between two noises with the same parameters. In contrast, most existing approaches in NPR that blend between two noises blend noises with different parameters, for example two octaves of Perlin noise, which have a different principal frequency.

We derive a weighting function α that preserves the statistical properties of the noise n_M, based on the power spectrum and the variance. In the Appendix, we derive the relation between the power spectrum and variance of noise n_{2M} and n_N. From Eqn. 6 and Eqn. 8 we see that n_{2M} preserves the variance and the power spectrum of n_N when $(1+\alpha^2)M/N$ equals 1. We obtain the weighting function α by solving this equation for α and substituting Eqn. 2:

$$\alpha(N) = \sqrt{\frac{N}{2^{\lfloor \log_2 N \rfloor}} - 1}. \tag{4}$$

3.4. Implementation

Since we target interactive applications, we implement NPR Gabor noise efficiently on the GPU, using geometry and fragment shaders.

We use a geometry shader to generate point sprites on-the-fly from incoming triangles. It is important for performance to discard invisible point primitives before rasterization. Therefore, we do not generate primitives for back-facing and invisible triangles. We resolve visibility using a

© 2010 The Author(s)
Journal compilation © 2010 The Eurographics Association and Blackwell Publishing Ltd.

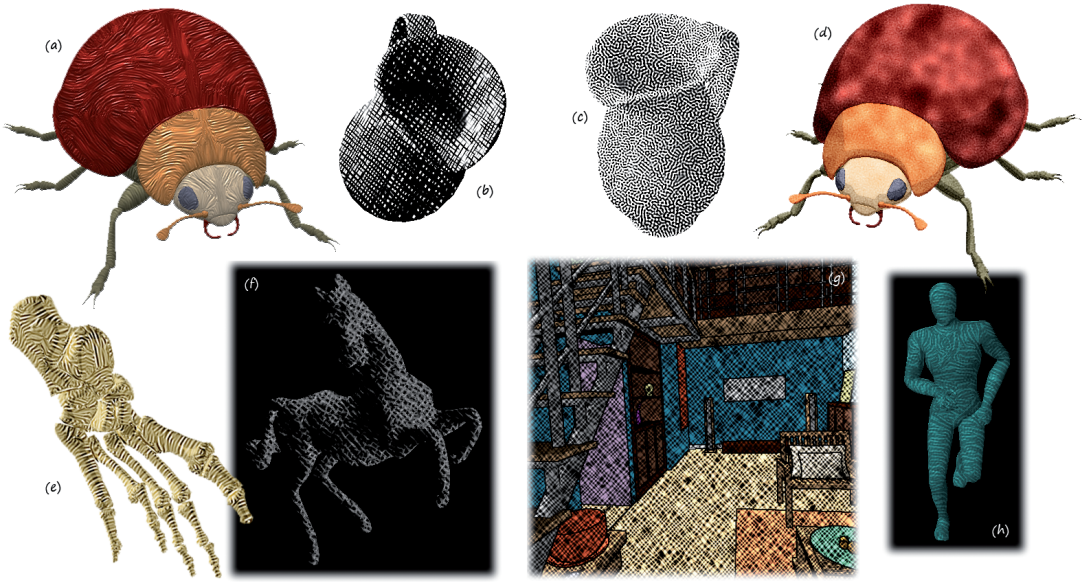


Figure 3: A variety of styles produced with our method. See the accompanying video to see these styles in motion.

depth test, or using a per-object ID buffer and a per-triangle ID buffer. In order to avoid popping (see Sec. 6), we use a fuzzy depth test, such as a blurred depth test [LD06] or a partial visibility test [CF08]. The number of point primitives that can be emitted by the geometry shader for each triangle is limited (in our setup to 64). We work around this limitation by subdividing large triangles, currently in a manual preprocess. Future GPU's with relaxed hardware limitations and the programmable tessellation unit will most likely remove this limitation.

We use a fragment shader to perform per-fragment visibility tests and to evaluate the Gabor kernel. The visibility tests include an object ID test, a triangle ID test, and a potentially blurred depth test. Each test affects the appearance of the noise, and can be enabled or disabled by the user. The object ID test, triangle ID test and depth test reject fragments based on object ID, triangle ID and depth, trimming the noise to object boundaries, triangle boundaries and depth edges.

Our GPU implementation runs at interactive rates, ranging from 8 to 50 fps for models of moderate complexity (4k to 50k tris), using off-the-shelf hardware (Intel Core 2 Duo 2.4GHz CPU and GeForce GTX 260 graphics card). Our GPU implementation scales well with scene complexity. Although every triangle is processed by the geometry processor, the total number of point sprites is approximately constant. Our GPU implementation supports anti-aliasing using super-sampling for higher image quality at the expense of a lower frame rate.

4. Styles

Our approach inherits all the advantages of noise primitives, while addressing the problem of temporal coherence. This means that we can use standard techniques from procedural texturing and modeling (see e.g., [EMP*02]) to generate a large variety of patterns that are well suited for NPR. We provide a "cookbook" with a precise description of the styles we designed in the additional material.

We model binary styles by thresholding a noise using a smooth step function (see Fig. 2), in the same spirit as Durand *et al.* [DOM*01]. This produces hatches in the case of an anisotropic noise, and stipples in the case of an isotropic noise (see Fig. 3(c)). The position of the step controls the overall intensity of the binary pattern, and the smoothness of the step controls the smoothness of the edges of the hatches and stipples. The frequency of the noise controls the average distance between stipples or hatches, and the bandwidth of the noise controls the regularity of the pattern. We achieve local control by linking these parameters to scene attributes: examples include noise orientation linked to geometric curvature (see Fig. 3(a,e)) or noise frequency linked to shading, allowing for various hatches or stipple sizes (see Fig. 3(b,c)). We obtain cross-hatching by multiplying two thresholded anisotropic noises, and graphite by modulating hatching with a high-frequency noise layer (see Fig. 2).

We produce color styles by compositing the noise with the scene color in various ways. We produce a watercolor style by modeling pigments and paper textures using noise, similar to [BKTS06] (see Fig. 3(d)). This approach uses scene color in an overlay blending mode that we also use to produce felt-tip patterns (see Fig. 3(g)). A painterly style is produced using large hatches overlaid with an high frequency anisotropic noise that simulates brush fibers. We add a bump mapping effect to emphasize the fibers (see Fig. 3(a)).

© 2010 The Author(s)
Journal compilation © 2010 The Eurographics Association and Blackwell Publishing Ltd.

		Flatness	Coherent motion	Temporal continuity	
Naive	Shower door	++	−−	++	
	Texture mapping	−−	++	++	
Few-primitive	[Mei96]	−/+	+	++	
	[VBTS07]	++	+	−−	*popping*
Object-space textures	[PHWF01]	−	++	−/+	*mipmap*
	[BBT09]	−	++	+	*fractalization*
Image-space texture	[BNTS07]	++	++	−/+	*regeneration*
	[CDH06, BSM*07]	+	−	−/+	*mipmap*
	[CTP*03]	++	−−	+	*fractalization*
Many-primitive	[KC05, BKTS06]	++	+	−	*flickering, oscillations*
	Ours	++	+	−/+	*secondary motion, residual popping*

Table 1: Summary of the trade-offs made by various solutions for the temporal coherency problem.

5. User study

The goals we set out in the introduction are contradictory, and therefore no perfect solution exists. As a consequence, evaluating the success of any given trade-off is very complex. Previous approaches only relied on speculative visual inspection. However, each goal has specific artifacts which a user can observe. A user study can thus provide significant insight into how well each solution performs for each goal. It also provides an indication of overall success as well as relative importance of the different criteria involved.

Note that in this user study we are only evaluating the trade-off of different methods with respect to temporal coherence. Other aspects, such as the range of patterns and the interactivity, will be discussed in Sec. 6.

5.1. Procedure

Evaluated techniques. Tab. 1 summarizes the relative merits of the compromises presented by each method, as discussed in Sec. 2. Among them, the extreme cases of shower door (SD) and texture mapping (TM) are logical choices for reference comparisons. We include the following state-of-the-art texture-based methods in our study:

- Dynamic Solid Textures (DST) [BBT09], an object-space approach,
- Bidirectional Texture Advection (Adv) [BNTS07], a local image-space approach,
- Dynamic 2D Patterns (D2D) [BSM*07], a global image-space approach.

For many-primitive approaches, we only used our method, since [KC05, BKTS06] can be seen as special cases of our solution applied to canvas and watercolor. We did not include few-primitive approaches in our comparisons for several reasons: (i) there is no way to produce the same pattern in a texture-based and a few-primitive approach; (ii) the severe popping produced by these methods is a highly disturbing, and thus obvious, artifact which probably does not require a user test. We show an example from [VBTS07] in our accompanying video (used with permission).

Setup. The study was split into two parts involving simple and complex stimuli respectively. A total number of 15 volunteers participated in both. Participants were asked to rank the stimuli, presented as still images or video loops, according to several criteria. Ranking was performed by dragging and dropping them on a dual monitor setup in a controlled environment as shown in Fig. 4(a). A detailed description of the experimental procedure and setup are provided in the additional materials. A sequence of a participant performing the study is also presented in the video.

Analysis methodology. We first determine appropriate assessment criteria for each goal defined in Sec. 1. We use them as guidelines to choose the stimuli and to formulate the questions we pose to the participants of the study. In the following sections we describe the details and results of the separate ranking tasks, in the order they were presented to the participants. The statistical significance of observed trends is confirmed by the Wilcoxon rank-sum hypothesis test [Wil45] and is indicated by the p-value. Fig. 5 provides the ordinal scales (interval scales are provided in the additional materials). and the similarity groups at the 95% significance level.

5.2. Simple stimuli

Simple stimuli were created to selectively test the separate goals in controlled conditions. Each stimulus consisted of a single object rendered with a black-and-white hatching pattern (see Fig. 4), chosen to best reveal the differences and possible artifacts of each selected method. This style corresponds to an extreme case; we thus believe that our conclusions generalize more readily to other styles. Lighting is disabled to avoid additional shape or motion cues [LMJY95, WFGS07].

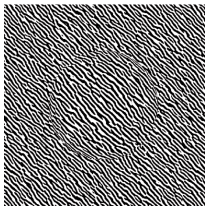
Flatness. The stimuli consist of still images depicting a sphere in front of a planar background, both rendered with the same pattern (Fig. 4(c,d)). Participants are asked to *rank the images according to how flat they appear.* Since lighting is disabled, the only remaining shape cues are the silhouette of the sphere and the perspective distortion of the pat-

© 2010 The Author(s)
Journal compilation © 2010 The Eurographics Association and Blackwell Publishing Ltd.

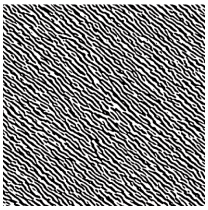
PREPRINT

(a) A participant performing the study

(b) Indoor scene



(c) Less flat



(d) More flat

(e) Pitcher

Figure 4: Some examples of the simple stimuli (c,d) from the flatness task and (e) from the coherent motion and temporal continuity tasks, and (b) a complex stimulus.

tern [Pal99, Ch. 5]. In practice, the sphere completely blends in with the flat background for methods that do not provide these shape cues in the pattern.

The results show a significant amount of variance, indicating the complexity of this question. People are not accustomed to relying on shape-from-texture cues in isolation. Additionally, several methods produce similar flatness by design: Shower Door and Dynamic 2D Patterns both produce perfectly flat images without perspective distortion, while Dynamic Solid Textures and Texture Mapping both produce the correct perspective. Of the hybrid 2D/3D methods, ours seems slightly closer to the 2D methods than Advection.

Coherent motion. The stimulus videos show a pitcher on a white background (Fig. 4(e)). We chose a pitcher as a simple, recognizable, everyday object, to make it easier for the observer to concentrate on the pattern. Nevertheless, the pitcher is sufficiently complex to exhibit some self-occlusion, which is handled differently by the compared methods. Three separate basic motions were ranked, in random order: translation parallel to the image plane, rotation, and zoom. We need to evaluate the correlation of the motion of the 3D scene and the motion of the pattern after stylization. Insufficient correlation creates the impression that the pattern is sliding over the depicted scene. Participants were asked to *rank the videos according to how coherently the pattern moves with the object.*

As expected, object-space methods such as Texture Mapping and Dynamic Solid Textures are significantly more coherent than the other techniques ($p < 0.05$), while Shower Door is consistently the least coherent ($p < 0.05$). The three

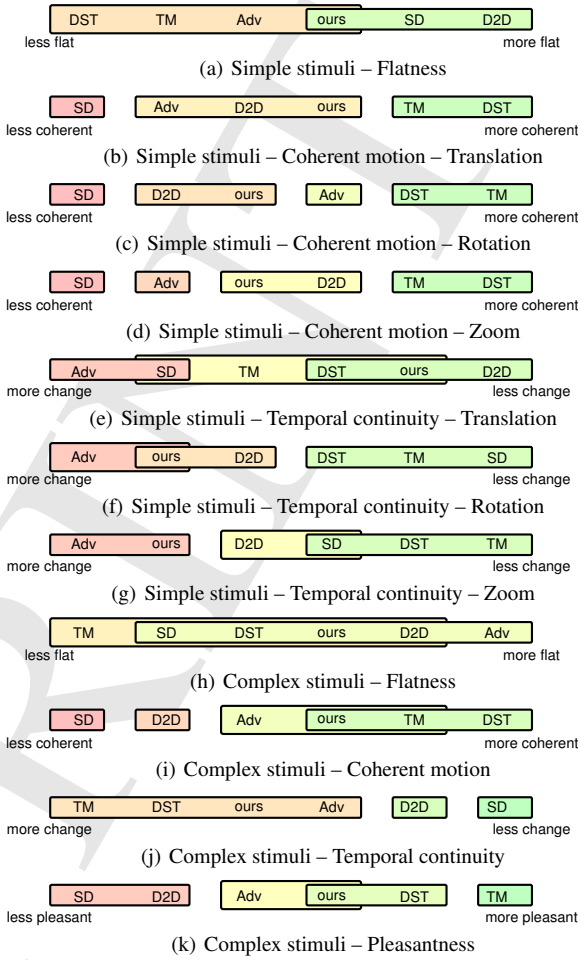


Figure 5: Ordinal scales indicating the relative merits of each method. The methods are classified into similarity groups based on the Wilcoxon rank-sum hypothesis test at 95% significance level. Within a group all methods perform similarly. Conversely, methods that are not grouped together are significantly different. A method can belong to multiple groups, shown here by overlapping rectangles.

other approaches which compromise flatness for motion coherence are perceived almost equally. Depending on the motion, the ranking varies slightly: Advection performs better for rotation ($p < 0.05$), whereas Dynamic 2D Patterns and our method perform better for the zoom ($p < 0.05$).

Temporal continuity. Given a stylized version of an animation, we want to estimate the fluidity or continuity of the animation. Temporal discontinuities such as popping strongly distract the attention of human observers [YJ84, SS09], and thus are easily identified. Temporally continuous artifacts, which are more subtle and consequently harder to notice, include contrast or intensity oscillations, as well as local residual motion of the patterns. For the same pitcher stimuli, participants were asked to *rank the videos according*

© 2010 The Author(s)
Journal compilation © 2010 The Eurographics Association and Blackwell Publishing Ltd.

to how much the pattern changes over time, regardless of the coherence of the motion of the object and the pattern, which they had already evaluated in the coherent motion task.

The results are not as clear-cut as for the previous question. This may be due to the complexity of the question. For example Shower Door, for which the pattern is constant over time, clearly shows that many participants had difficulty to both ignore the principal object translation and to concentrate on residual motions.

Overall, Advection introduces the largest changes to the pattern in the form of a residual motion called "swimming" in [BNTS07, Sec. 5], which are made more apparent by subsequently applying the binary threshold. For this kind of pattern, our approach exhibits similar effects during rotations and zoom.

We constrained the orientation and scale of the pattern in 2D but chose to attach the primitives to the 3D surface. Accommodating these conflicting constraints in an animation can result in artifacts such as popping or residual motion. This may be why our method performs poorly on the temporal continuity scale for rotation and zoom. Advection, Dynamic 2D Patterns, and our method are the only ones that actually try to satisfy all these constraints, whereas the other methods do not support some 2D constraints (TM and DST) or 3D constraints (SD).

5.3. Complex stimuli

We created complex stimuli to study the behavior of the different approaches for a more aesthetically pleasing experience in a complex walkthrough. The viewer moves inside an indoor scene which contains a mix of large planar surfaces, small details, grazing angles, and depth discontinuities. To make the different objects and surfaces more apparent, lighting is enabled, black outlines are drawn, and the scene is rendered with colored cross hatching patterns on a white paper background (Fig. 4(b)).

We kept the same questions as in the first stage. Flatness is evaluated on still images; coherent motion and temporal continuity are evaluated on a complex motion path. To assess the overall aesthetic appreciation of the different approaches, participants were additionally asked to *rank the videos according to how pleasant they find them in the context of cartoon animation.*

Once again, the variance of the answers while estimating the flatness of the scene shows the complexity of this question. This is more pronounced in the complex scene, due to the additional shape and depth cues.

The responses concerning motion coherence are consistently in favor of object-space approaches. Our noise ranks with the best of the image-space methods for this goal.

The results for motion coherence and pleasantness (Fig. 5(i) and (k)) are strongly correlated (Kendall rank correlation $\tau = 0.58$ [Ken38]). This indicates that motion coherence is probably the most important quality to preserve in the overall temporal coherence compromise. Both Dynamic Solid Textures and our method perform well on the motion coherence scale, and trade off better temporal continuity (DST) versus better flatness (ours).

6. Discussion

6.1. Popping

The GPU implementation of our noise primitive based on splatting is subject to residual popping artifacts, which may be reinforced by the stylization. This popping is caused by aliasing in the per-point-sprite visibility tests, due to the mismatch between the continuous point sprite position and the discretized visibility. This problem is typically alleviated using *fuzzy* visibility tests, such as the blurred depth test [LD06]. We use a partial visibility test based on the work of Cole and Finkelstein [CF08]. It works relatively well in practice, although some popping may remain according to the mesh and style. Moreover, partial visibility avoids binary visibility decisions at (dis)occlusion boundaries, which reduces the residual motion of the pattern near silhouettes.

We believe it might be possible to completely remove the remaining popping artifacts by using a procedural approach, as in [LLDD09]. However, initial experiments were not encouraging in terms of performance. Because we target interactive rates, which is especially important for the artistic design process, we believe that a splatting approach is currently more suited for NPR Gabor noise It would be interesting to develop such a high quality offline approach which could use the current method as a previsualization tool. Appropriate guarantees on similarity of visual results must of course be provided.

6.2. Styles

Our goal is not to precisely reproduce a specific style, which explains why our results do not always look like their real-world equivalent (e.g., hatching and stippling). Instead, we provide a noise primitive that is temporally coherent, such that the artist can focus on style creation rather than on the dynamic behavior of the stylization.

Our noise primitive does not offer direct control at the stroke level. Consequently, we cannot create styles with clearly independent primitives. This limitation is shared with other texture-based and many-primitives methods. However, in contrast to most other texture-based approaches, our noise primitive does offer local control, for example by varying the noise parameters according to scene attributes.

Our current method for style design is not entirely trivial. Nevertheless, we were able to create a wide variety of styles, with reasonable effort, and without an advanced user interface (see supplemental material and the video for a style-creation sequence). Of course, more intuitive methods for artistic control would significantly facilitate this process. In this context, it is important to note that procedural texturing

© 2010 The Author(s)
Journal compilation © 2010 The Eurographics Association and Blackwell Publishing Ltd.

and noise, the concepts we use for style design, have been used by artist for decades.

6.3. User study

Our user study is necessarily limited by the choices we have made. For example, the choice of hatching as a stimulus, which makes the fractalization artifacts introduced by dynamic solid textures hard to see, or the choice for naive users, which could respond differently than expert users such as professional artists, and fundamentally the choice of the goals we evaluated. Of course, a more elaborate user study would be very interesting, particularly to refine the relative importance of these goals.

Our user study provides a comparison of several methods, but has to be interpreted with care:

First, as explained in the introduction, the problem of temporal coherence in NPR involves intrinsically contradictory constraints. Therefore, any possible solution to this problem will be a trade-off, and no method can have the best performance for all of the criteria simultaneously. We believe a good trade-off should provide a satisfactory compromise for as many criteria as possible, and should not completely break down for one of them. Overall, our new method achieves this goal well.

Second, the user study only covers some aspects of the methods. Our approach has several advantages over the others that are not or only partially revealed by the user study. Compared to dynamic 2D Patterns [BSM*07], our method is not subject to sliding, and is better at handling complex scenes. Compared to dynamic solid textures [BBT09], our method is better in flatness, and does not degrade the pattern. Compared to bidirectional texture advection [BNTS07], our method runs at interactive rates, does not require knowledge of the entire animation, and preserves the pattern better.

7. Conclusion

We have presented a new dynamic noise primitive for coherent stylization, and a user study investigating temporal coherence. Our new noise primitive allows the use of standard techniques from procedural texturing to create various temporally coherent styles for interactive NPR applications. Our user study provides more insight into the problem of temporal coherence, for example, that motion coherence is one of the most important qualities to preserve in the overall temporal coherence compromise, and that our new solution provides a good compromise between the 2D aspect of the style and 3D motion.

Our noise primitive for NPR makes the connection between primitive-based methods in NPR and methods for sparse convolution noise in procedural texturing more explicit. This places existing work in a different perspective. For example, the watercolor of Bousseau et al. [BKTS06] is actually a sparse convolution noise [Lew89] with a Gaussian-like kernel, and the dynamic canvas of Kaplan and Cohen [KC05] is actually a spot noise [vW91] with a fiber as the spot shape. This connection is likely to be productive. For example, to further extend the range of patterns we can produce, we plan to construct a version of spot noise [vW91] that takes into account temporal coherency, using the ideas of NPR Gabor noise.

Our user study is a first attempt to better understand the trade-offs involved in temporal coherence, and a first step towards a more formal evaluation procedure for this problem. We plan to do more user studies, taking into account what we have learned so far, but also to explore objective measures to quantify these trade-offs, for example, statistical texture measures to quantify flatness, in the spirit of [BTS09], and optical flow to quantify motion coherence and temporal continuity. This is likely to lead to deeper insight in the evaluation of solutions for the temporal coherence problem.

Acknowledgments

We would like to thank the anonymous reviewers, Kartic Subr and Adrien Bousseau. We are especially grateful to Laurence Boissieux for 3D modeling and for producing the advection sequences, to Marcio Cabral for producing the video, and to the participants of the user study. Ares Lagae is a Postdoctoral Fellow of the Research Foundation - Flanders (FWO), and acknowledges K.U.Leuven CREA funding (CREA/08/017).

References

[BBT09] BÉNARD P., BOUSSEAU A., THOLLOT J.: Dynamic solid textures for real-time coherent stylization. In *Proc. 2009 Symp. Interactive 3D graphics and games* (2009), pp. 121–127. 2, 6, 9

[BKTS06] BOUSSEAU A., KAPLAN M., THOLLOT J., SILLION F.: Interactive watercolor rendering with temporal coherence and abstraction. In *Proc. 4th Int. Symposium on Non-Photorealistic Animation and Rendering* (2006), pp. 141–149. 2, 3, 4, 5, 6, 9

[BNTS07] BOUSSEAU A., NEYRET F., THOLLOT J., SALESIN D.: Video watercolorization using bidirectional texture advection. *ACM Trans. Graphics 26*, 3 (2007), 104:1–104:8. 2, 6, 8, 9

[BSM*07] BRESLAV S., SZERSZEN K., MARKOSIAN L., BARLA P., THOLLOT J.: Dynamic 2d patterns for shading 3d scenes. *ACM Trans. Graphics 26*, 3 (2007), 20:1–20:6. 1, 2, 6, 9

[BTS09] BÉNARD P., THOLLOT J., SILLION F.: Quality assessment of fractalized NPR textures: a perceptual objective metric. In *Proc. 6th Symp. Applied Perception in Graphics and Visualization* (2009), pp. 117–120. 2, 9

[CDH06] COCONU L., DEUSSEN O., HEGE H.-C.: Real-time pen-and-ink illustration of landscapes. In *Proc. 4th Int. Symposium on Non-Photorealistic Animation and Rendering* (2006), pp. 27–35. 2, 6

[CF08] COLE F., FINKELSTEIN A.: Partial visibility for stylized lines. In *Proc. 6th Int. Symposium on Non-Photorealistic Animation and Rendering* (2008), pp. 9–13. 5, 8

[CL06] CHI M.-T., LEE T.-Y.: Stylized and abstract painterly rendering system using a multiscale segmented sphere hierarchy. *IEEE Trans. Visualization and Computer Graphics 12*, 1 (2006), 61–72. 2

© 2010 The Author(s)
Journal compilation © 2010 The Eurographics Association and Blackwell Publishing Ltd.

[CTP*03] CUNZI M., THOLLOT J., PARIS S., DEBUNNE G., GASCUEL J.-D., DURAND F.: Dynamic canvas for immersive non-photorealistic walkthroughs. In *Proc. Graphics Interface 2003* (2003), pp. 121–130. 1, 2, 6

[Dan99] DANIELS E.: Deep canvas in disney's tarzan. In *ACM SIGGRAPH 99 Conference abstracts and applications* (1999), p. 200. 2

[DOM*01] DURAND F., OSTROMOUKHOV V., MILLER M., DURANLEAU F., DORSEY J.: Decoupling strokes and high-level attributes for interactive traditional drawing. In *Proc. 12th Eurographics Workshop on Rendering Techniques* (2001), pp. 71–82. 5

[EMP*02] EBERT D. S., MUSGRAVE F. K., PEACHEY D., PERLIN K., WORLEY S.: *Texturing and Modeling: A Procedural Approach*, 3rd ed. Morgan Kaufmann Publishers, Inc., 2002. 2, 3, 5

[FW07] FRISVAD J. R., WYVILL G.: Fast high-quality noise. In *Proc. GRAPHITE'07* (2007), pp. 243–248. 3

[HE04] HAYS J., ESSA I.: Image and video based painterly animation. In *Proc. 3rd Int. Symposium on Non-Photorealistic Animation and Rendering* (2004), pp. 113–120. 2

[KC05] KAPLAN M., COHEN E.: A generative model for dynamic canvas motion. In *Proc. 1st Eurographics Workshop on Computational Aesthetics in Graphics, Visualization and Imaging* (2005), pp. 49–56. 2, 3, 4, 6, 9

[Ken38] KENDALL M. G.: A new measure of rank correlation. *Biometrika 30*, 1-2 (1938), 81–93. 8

[KGC00] KAPLAN M., GOOCH B., COHEN E.: Interactive artistic rendering. In *Proc. 1st Int. Symposium on Non-Photorealistic Animation and Rendering* (2000), pp. 67–74. 2

[KLK*00] KLEIN A. W., LI W. W., KAZHDAN M. M., CORREA W. T., FINKELSTEIN A., FUNKHOUSER T. A.: Non-photorealistic virtual environments. In *Proc. ACM SIGGRAPH 2000* (2000), pp. 527–534. 2

[LD06] LUFT T., DEUSSEN O.: Real-time watercolor illustrations of plants using a blurred depth test. In *Proc. 4th Int. Symposium on Non-Photorealistic Animation and Rendering* (2006), pp. 11–20. 5, 8

[Lew84] LEWIS J.-P.: Texture synthesis for digital painting. In *Computer Graphics (Proc. ACM SIGGRAPH 84)* (1984), vol. 18, pp. 245–252. 2

[Lew89] LEWIS J. P.: Algorithms for solid noise synthesis. In *Computer Graphics (Proc. ACM SIGGRAPH 89)* (1989), vol. 23, pp. 263–270. 2, 3, 9

[LLC*10] LAGAE A., LEFEBVRE S., COOK R., DEROSE T., DRETTAKIS G., EBERT D., LEWIS J., PERLIN K., ZWICKER M.: State of the art in procedural noise functions. In *EG 2010 - State of the Art Reports* (2010). 2

[LLD09] LAGAE A., LEFEBVRE S., DUTRÉ P.: *Improving Gabor Noise*. Report CW 569, Department of Computer Science, K.U.Leuven, 2009. 3

[LLDD09] LAGAE A., LEFEBVRE S., DRETTAKIS G., DUTRÉ P.: Procedural noise using sparse Gabor convolution. *ACM Trans. Graphics 28*, 3 (2009), 54:1–54:10. 2, 3, 8, 10

[LMJY95] LANDY M. S., MALONEY L. T., JOHNSTON E. B., YOUNG M.: Measurement and modeling of depth cue combination: in defense of weak fusion. *Vision Research 35*, 3 (1995), 389–412. 6

[Mei96] MEIER B. J.: Painterly rendering for animation. In *Proc. ACM SIGGRAPH 1996* (1996), pp. 477–484. 1, 2, 6

[Pal99] PALMER S. E.: *Vision Science: Photons to Phenomenology*. MIT Press, 1999. 7

[Per85] PERLIN K.: An image synthesizer. In *Computer Graphics (Proc. ACM SIGGRAPH 85)* (1985), vol. 19, pp. 287–296. 2, 3

[PHWF01] PRAUN E., HOPPE H., WEBB M., FINKELSTEIN A.: Real-time hatching. In *Proc. ACM SIGGRAPH 2001* (2001), pp. 581–586. 2, 6

[SS09] SCHWARZ M., STAMMINGER M.: On predicting visual popping in dynamic scenes. In *Proc. 6th Symp. Applied Perception in Graphics and Visualization* (2009), pp. 93–100. 1, 7

[VBTS07] VANDERHAEGHE D., BARLA P., THOLLOT J., SILLION F.: Dynamic point distribution for stroke-based rendering. In *Proc. 2007 Eurographics Symp. Rendering* (2007), pp. 139–146. 2, 6

[vW91] VAN WIJK J. J.: Spot noise texture synthesis for data visualization. In *Computer Graphics (Proc. ACM SIGGRAPH 91)* (1991), vol. 25, pp. 309–318. 2, 3, 9

[WFGS07] WINNEMÖLLER H., FENG D., GOOCH B., SUZUKI S.: Using npr to evaluate perceptual shape cues in dynamic environments. In *Proc. 5th Int. Symposium on Non-Photorealistic Animation and Rendering* (2007), pp. 85–92. 6

[Wil45] WILCOXON F.: Individual comparisons by ranking methods. *Biometrics Bulletin 1*, 6 (1945), 80–83. 6

[YJ84] YANTIS S., JONIDES J.: Abrupt visual onsets and selective attention: evidence from visual search. *J. Experimental Psychology 10*, 5 (1984), 601–621. 1, 7

Appendix

To determine the weight α in Eqn. 4, we need to determine the power spectrum and variance of n_{2M} in terms of n_N.

Power spectrum. First, we take the magnitude of the Fourier transform of both sides of Eqn. 3. Then we simplify, noting that the Fourier transform is a linear operator, and that n_M and n'_M have the same expected Fourier transform,

$$|\mathcal{F}(n_{2M})|^2 = (1+\alpha^2)|\mathcal{F}(n_M)|^2. \tag{5}$$

Note that with expected Fourier transform we mean the Fourier transform of the stochastic processes corresponding to n_M and n'_M rather than the Fourier transform of specific instances of n_M and n'_M. Finally, we apply equation 10 of [LLDD09], noting that $\lambda_M = M/A = M/N\lambda_N$,

$$|\mathcal{F}(n_{2M})|^2 = (1+\alpha^2)\frac{M}{N}|\mathcal{F}(n_N)|^2. \tag{6}$$

This establishes a relation between the power spectrum of the noises n_N and n_{2M}.

Variance. First, we take the variance of both sides of Eqn. 3. Then we simplify, noting that n_M and n'_M are uncorrelated and have the same expected variance,

$$\sigma^2_{2M} = (1+\alpha^2)\sigma^2_M. \tag{7}$$

Finally, we apply Eqn. 10 of [LLDD09],

$$\sigma^2_{2M} = (1+\alpha^2)\frac{M}{N}\sigma^2_n. \tag{8}$$

This establishes a relation between the variance of the noises n_N and n_{2M}.

© 2010 The Author(s)
Journal compilation © 2010 The Eurographics Association and Blackwell Publishing Ltd.