

単一画像からの汚れシミュレーション

張 英夏 (正会員) 吉田 隼人 向井 信彦 (正会員)

東京都市大学大学院工学研究科

Weathering Simulation based on a Single Image

Youngha Chang Hayato Yoshida and Nobuhiko Mukai

Graduate School of Engineering, Tokyo City University

{chang, mukai} @ vgl.cs.tcu.ac.jp

概要

人間によって作られた建築物は、時間の経過と共に劣化していく。この経年変化は環境や時間などの様々な条件によって多様な表れ方をする。景観シミュレーションを行う際にはこのような複雑な経年変化を表現することで現実感を増すことができる。従来は熟練されたアーティストが手作業によって経年劣化を表現していた。しかしながら大規模なシーンを対象とした場合、非常に時間と手間がかかるという問題が生じており、このため近年、汚れを簡易的に生成する様々なアプローチが提案されている。本稿では、雨水による汚れを対象とし、一枚の画像から生成される汚れの層を複数重ね合わせることによって、経年による複雑な汚れを表現する手法の提案を行う。また、水の汚れ表現のみではなく、壁に付着した水分量に基づくコケの発生もシミュレーションすることにより、より多様な経年変化の表現を可能とする。

Abstract

Weathering affects urban scenes and forms complex patterns that vary with time, environment, and the surface of architectures such as buildings. This causes dramatic changes in the appearance. Thus, it is very important to consider weathering effects in the rendering process for virtual cities. In previous methods, artists have represented weathering effects by hand; however, it takes huge amount of time. Therefore, in recent years, many works have been proposed to generate weathering effects with simplified simulations. In this paper, we propose an image-based method to generate complex patterns caused by rain drops, which represents the aging effect by combining multiple layers representing weathering effects, and also represents the moss generated according to the amount of water attached on the wall.

1 はじめに

近年、コンピュータの性能向上によって高精度かつ大規模な景観シミュレーションが可能になり、景観デザインのような実務分野からゲームなどのエンターテインメント分野まで多岐に渡って利用されている。これまでのコンピュータシミュレーションは基本的には全ての対象物を真新しい見栄えでレンダリングしていた。一方、現実世界において人間によって作られた構造物は劣化によって外観が変化する。この経年変化は時間の経過や環境の変化によって非常に多様な表現を成しており、現実感を実現するための重要な要素であると考えられる。

従来は、熟練されたアーティストが手作業で経年変化を表現していた。しかしながら膨大な景観シーンをレンダリングする必要がある場合には、多大な時間と労力がかかるという問題があり、近年、汚れを付加するためのウェザリング技術に注目が集まっている。中でも景観シーンで頻繁に現れ、かつ生成することが難しい汚れとして、流体による経年変化が挙げられる。流体は物体形状に変化をもたらすだけでなく、流れた痕跡や汚れの伝搬、沈着、融解、さらにはコケやカビの発生などに関わる。

本稿では、水による複合的な経年変化を表現する手法を提案する。まず表面形状を考慮した水の落下による水汚れを表現し、水汚れ画像をレイヤーとして複数枚重ねることによって経年変化による複雑な水汚れ模様を生成する手法を提案する。さらに、壁に付着した水分量に応じてフラクタルを用いたコケ模様を表示することによって、より多様な経年劣化を表現する手法についても提案する。

2 関連研究

従来、経年劣化を表現するために様々なアプローチが提案されている [1, 2]。ひび割れ [3, 4]、コケの生長 [5]、木材の経年変化 [6] など劣化表現に関する手法が提案されているが、本節では主に水を原因とした汚れ生成手法について述べる。

Dorsey らは水滴を粒子によって表現し、簡略化した物理シミュレーションを行うことによって 3 次元表面上に水汚れや錆の汚れを表現した [7]。粒子の動きは物体の表面形状や水滴の属性によって制御される。時間ステップ毎に水滴と物体表面間における水の移動量、壁面への水の沈着量および融解量の変化を計算する。粒子の初期位置は雨量や風の方向などのユーザパラメータに基づいて計算され、雨がビルなどの壁面に付着することによって汚れのシミュレーションが始ま

る。水滴は表面を伝って流れながら沈着と融解を繰り返し、最終的に汚れ画像はテクスチャとして保存され、レンダリングに用いられる。

一方、Chen らはこのアプローチに基づき、汚れ現象を一般化した概念に発展させた手法を提案している [8]。彼らは経年変化を誘発する粒子をガンマトンと命名し、フォントレーシングに似た方法でシーン中にある大量のガンマトンを追跡し、ガンマトンに保存された汚れ要因に基づいて経年劣化を表現している。アーティストが目的に応じて汚れ要因を設定することによって、様々な種類の劣化表現が可能になり、さらには複数要因による経年劣化の効果も表現できる。

さらに、Xue らは二次元画像を基にした汚れの表現手法を提案した [9]。画像における物体の色を物体反射と光源色の積とみなし、繰り返し処理によって経年変化による反射の多様性成分を抽出することで、経年変化を後退または進行させることに成功している。しかしながら物体表面形状に依存した要因による経年変化は表現できないという問題がある。

このため遠藤らは物体表面の凹凸を画素の輝度情報から推測することによって、表面形状に応じた雨の流れを表現した [10]。汚れ生成手法としては Dorsey らの手法を 2 次元画像に適用している。また、パラメータ調整によって所望する形状の汚れを直感的に生成することが可能になっている。

そこで本稿では、遠藤らの手法にさらに改良を加え、より複雑な経年劣化を表現する手法を提案する [11]。そのために、生成された水汚れ画像を時系列にレイヤーとして保存する。この時、各レイヤーにおける水汚れの性質、例えば水滴の質量や位置、水滴の個数を変化させることによって、レイヤーを合成した際に複雑な水汚れ模様が表現できる手法を提案する。一つのレイヤーはある期間の汚れを表しており、ある期間のみの汚れを可視化できる。また、レイヤーを一つずつ重ねていくことで、汚れの変遷を可視化することができる。

3 水汚れの表現手法

本節では、水汚れ量の計算手法を述べる。3.1 では、建物の表面上に付着した雨水の壁面への染み込み、しみ出し、および表面上での水の伝搬方法について述べる。また 3.2 では、3.1 で説明した水汚れが時間経過と共に積層される様子を再現する手法について述べる。

3.1 水汚れの表現

遠藤らの手法を基にして一層の水汚れを生成する．遠藤らの手法は Dorsey らの 3 次元モデルに対する手法 [2] を基に，2 次元画像を用いたインタラクティブな水汚れの生成手法を提案している．本節では遠藤らの手法と，本研究で行った修正点について明記する．

水汚れは壁と水滴という 2 つの要素によって生成される．壁面には凹凸があり，壁面上を粒子で表現された水滴が速度を持って流れ落ちる．水滴が壁面を流れると，壁と水滴の間で水分量の移動が発生し，同時に壁面における水汚れの沈着および水の溶解量が定まる．以下に，各項目の計算方法を述べる．

まず最初に，壁面上の凹凸を粗さマップとして表現する．3 次元を対象とした Dorsey らの手法とは異なり，遠藤らの手法は 2 次元画像を対象としているため，表面上の凹凸情報は正確には計算できない．そこで画像上の輝度情報を用いて，周辺画素との輝度差が大きい場合には表面の凹凸（粗さ）が大きいと推測し，水平および垂直方向の粗さマップ $M_x(p)$ および $M_y(p)$ を以下の式で作成する．

$$M_x(p) = |B(p_x, p_y) - \frac{B(p_{x+1}, p_y) + B(p_{x-1}, p_y)}{2}| / 255$$

$$M_y(p) = |B(p_x, p_y) - B(p_x, p_{y+1})| / 255 \quad (1)$$

ここで， $p = (p_x, p_y)$ であり，画像上での画素位置を意味する． $B(p_x, p_y)$ は p における輝度値 ($0 \leq B(p_x, p_y) \leq 255$) である．

次に，各水滴粒子の流速の変化量を求める．各水滴粒子の速度 $v = (v_x, v_y)$ は x および y 方向の粗さマップ $M_x(p)$ および $M_y(x)$ の値に応じて求まる．ここで，垂直方向の粗さ $M_y(p)$ に応じて，水平方向に粒子が散乱するよう， x 方向への散乱度合い $v_x^{diffuse}$ を導入する．なお，粒子の速度について基本的には遠藤らの数式に基づくが，本稿では建物や外壁などの垂直な表面への汚れ発生を仮定しているため，垂直方向の落下加速度 $\frac{\partial v_y}{\partial t}$ に重力加速度を追加する．この結果，速度の変化量は次式によって計算できる．

$$v_x^{diffuse} = br\xi_1 M_y(p) \quad (2)$$

$$\frac{\partial v_x}{\partial t} = v_x^{diffuse} - v_x \frac{M_x(p)}{r_{max} - r + 1} \quad (3)$$

$$\frac{\partial v_y}{\partial t} = g - v_y \rho \frac{M_y(p)}{r_{max} - r + 1} \quad (4)$$

ここで， b は定数， r はユーザによって調整可能なパラメータで， r の値が大きくなると散乱の度合いが大きくなる． ξ_1 は $[-1, 1]$ の一様乱数， r_{max} は r の最大値であり， g は重力加速度， ρ は壁の抵抗率を示す．

次に，壁を流れる水滴と壁との水分量の移動について説明する．この移動に関する壁および水滴のパラメータを図 1 に示す．水滴は粒子として表現し，各粒

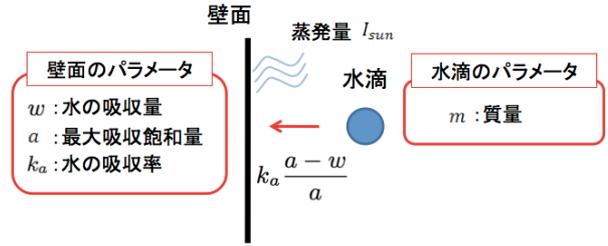


図 1: 水の移動量

子は質量 m ，位置 p ，速度 v ，溶解している汚れ量 S のパラメータを持つ．物体表面は，水の吸収量 w ，水の吸収率 k_a ，最大吸収飽和量 a ，沈着している汚れ量 D のパラメータを持つ．シミュレーションのタイムステップ毎に，水滴の質量と壁面における水の吸収量が変化する．遠藤らの手法では，壁面への汚れの沈着量を直感的に操作するためのパラメータを導入したが，本稿では沈着量は水滴の質量に比例すると考え，このパラメータを省略し，質量 m と吸収量 w の変化量を以下のように定義する．

$$\frac{\partial m}{\partial t} = -k_a \frac{a-w}{a} m \quad (5)$$

$$\frac{\partial w}{\partial t} = k_a \frac{a-w}{a} m - I_{sun} \quad (6)$$

ここで， t は時刻， I_{sun} は水の蒸発量を表す．

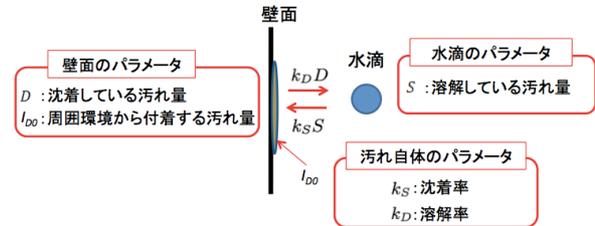


図 2: 汚れ量の沈着と吸収

さらに，運搬された水分の壁への移動により，汚れの壁面への沈着量と汚れの水滴への溶解量の変化を計算する．図 2 に，関連するパラメータを示す．遠藤らは沈着率 k_S と溶解率 k_D を用いて，水滴中の汚れ量 S および壁面中の汚れ量 D の変化量を次式のように定義した．

$$\frac{\partial S}{\partial t} = -k_S S + k_D D \quad (7)$$

$$\frac{\partial D}{\partial t} = k_S S - k_D D + I_{D0} \quad (8)$$

なお， I_{D0} は時間ステップごとに周囲環境から付着される汚れ量を表し， $k_S = 1 - k_D$ が成り立つものとする．

最後に、壁内部における水分の拡散を考慮する．壁面上のある地点 $p = (p_x, p_y)$ で吸収された水分量 w_p は、周囲近傍への移動を繰り返しながら次第に拡散していく．繰り返し回数 i 回目に位置 p で保持される水分量を w_p^i 、 p の 8 近傍を構成する各画素 $n(p)$ に拡散する水分量を $w_{n(p)}^i$ とし、次式によって水分の拡散を表現する．

$$w_{n(p)}^i = \tau w_p^{i-1} \quad (9)$$

$$w_p^i = w_p^{i-1} - 8w_{n(p)}^i \quad (10)$$

ただし、 $w_p^0 = w_p$ 、 τ は伝搬量を調整するためのパラメータである．また、繰り返し回数 $i+1$ 回目の計算では、8 近傍を構成する画素 $n(p)$ から同様の手順で水分が拡散される．なお、質量 m がある一定のしきい値以下になった粒子は、シミュレーションから除外される．

以上で、単位時間ステップにおける水分の移動と、それに伴う汚れの沈着および溶解について説明した．なお、各処理で説明したように、式 (4), (5), (6), (9) および (10) が遠藤らの手法から本稿で改良した式である．

3.2 水汚れの多層化

壁の水汚れは一度に生成されるものではなく、壁が乾湿を繰り返すことで新たな汚れが沈着し、複雑な汚れの様態を示す [12]．本節ではこのような汚れの変化を表現するために、3.1 で生成された汚れの層を複数枚積み重ねる手法について説明する．3.1 の手法を用いて生成した汚れ画像をレイヤー $(i-1)$ として保存し、新たにレイヤー (i) を計算する方法を説明する．レイヤー $(i-1)$ はある時点で雨が降った時の汚れの付着量であり、レイヤー (i) はその後雨が降った時の汚れの付着量であると考え、新たなレイヤー (i) が生成されるまでに、壁面に残留している水分量はある程度蒸発していると考えられる．また、壁面に沈着した汚れの一部も風や外部物体との接触により減少したり、周囲環境から新たな汚れが一定量付着したりしているはずである．そこで、新しいレイヤー (i) を計算する前に、次式を用いて壁面の水分量および前レイヤー $(i-1)$ における汚れ量 D_p^{i-1} の更新を行う．

$$w_p = \xi_2 w_p \quad (11)$$

$$D_p^{i-1} = \xi_2 D_p^{i-1} + I_{D0} \quad (12)$$

ここで、 ξ_2 は $[0, 1)$ の一様乱数であり、 I_{D0} は周囲環境からの新たに付加される汚れ量、 D_p^{i-1} はレイヤー $(i-1)$ の画素位置 p における汚れ量である．

レイヤー間における汚れ量の移動を図 3 に示す．

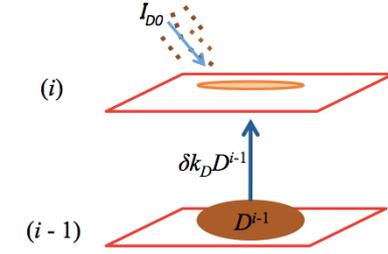


図 3: レイヤー間における汚れ量の移動

図 3 に示すように、レイヤー (i) において水滴以外から受ける汚れ量は、周囲環境からの汚れ量 I_{D0} と少し前の時点における汚れ量 (レイヤー $(i-1)$ の汚れ量) からの移動量の加算である．このため、3.1 で説明した式 (8) における周囲環境からの汚れ量 I_{D0} を次式の I_{D^i} に置き換える．

$$I_{D^i} = I_{D0} + \delta k_D D^{i-1} \quad (13)$$

ここで、 I_{D0} は周囲環境からの汚れ量、 δ はレイヤー $(i-1)$ からレイヤー (i) への汚れの引き継ぎ率を表す定数、 $\delta k_D D^{i-1}$ はレイヤー $(i-1)$ からの汚れの移動量である．この結果、レイヤー $(i-1)$ における汚れ量 D^{i-1} は、次式のようにレイヤー (i) に移動した量だけ減少する．

$$D^{i-1} = D^{i-1} - \delta k_D D^{i-1} \quad (14)$$

また、式 (1) に示す粗さマップ $M(p)$ は、輝度値 $B(p)$ に基づいて求めていたが、汚れの付着による高さの変化を粗さマップに反映するために、式 (1) に用いる変数 $B^i(p)$ を次式のように更新する．

$$B^i(p) = \min\left(\varepsilon \frac{D^{i-1}}{D_{max}^{(i-1)}} + B^{i-1}(p), 255\right) \quad (15)$$

ここで、 $\varepsilon \frac{D^{i-1}}{D_{max}^{(i-1)}}$ の項は、少し前の時点 (レイヤー $(i-1)$) における汚れ量による高さの変化を反映するために導入した項である． ε は定数である． D_{max}^i はレイヤー (i) の最大汚れ量であり、汚れ量を $[0, 1]$ に正規化するために導入した．したがって、 $B^i(p)$ の値は少し前の時点 (レイヤー $(i-1)$) における汚れ量に基づき、 $B^{i-1}(p)$ より 0 から最大 ε まで増加する．ただし、 $B^i(p)$ は式 (1) で説明した通り、 $[0, 255]$ の範囲を仮定しているため、汚れの高さを付加することにより $B^i(p)$ の値が 255 を超える場合、255 までに丸め込む．

なお、各レイヤーにおける水滴の質量 m^i や水滴が発生する位置 p^i は次式によってランダムに割り当てられる．

$$m^i = \sqrt{-2\ln\xi_3 \sin(2\pi\xi_4)\sigma} + m^{i-1} \quad (16)$$

$$p^i = \sqrt{-2\ln\xi_3 \sin(2\pi\xi_4)\sigma} + p^{i-1} \quad (17)$$

ただし, ξ_3 および ξ_4 は $[0,1]$ の一様乱数である. 式 (16) および (17) は平均が m^{i-1} あるいは p^{i-1} , 標準偏差が σ の正規分布を用いた正規乱数であり, レイヤー (i) における水滴の質量 m^i はレイヤー ($i-1$) における水滴の質量 m^{i-1} を平均とする正規乱数を用いて求められる. なお, 標準偏差 σ を用いて質量 m^i と位置 p^i の分布を調整する.

4 コケの表現手法

複数の汚れをインタラクティブに表現するために, 計算コストの低減が課題である. そこで本章では, 3. で計算した表面形状の粗さマップや汚れ量などを再利用することで, 計算コストのかからないコケの付加手法を提案する.

コケは水分の残留量が多く, また, 壁の劣化が激しい場所や汚れの多い場所に発生しやすい [13]. そこで, 水汚れの生成で用いた壁面の汚れ量 D と水の吸収量 w , および垂直方向の粗さマップ $M_y(x)$ によるコケの発生確率を, 次式を用いて求める.

$$f(p) = \frac{1}{2} \left(\frac{w}{a} + \frac{D}{D_{max}} \right) M_y(p) \gamma \quad (18)$$

ここで, $\gamma (0 \leq \gamma \leq 1)$ は定数であり, γ が大きいほどコケの発生確率は高くなる. w/a が水分の残留量に対応した項, D/D_{max} が汚れに対応した項である. 分母にある a および D_{max} は w/a および D/D_{max} を $[0, 1]$ に正規化するために導入した. ここで, $0 \leq \frac{1}{2} \left(\frac{w}{a} + \frac{D}{D_{max}} \right) \leq 1$, $0 \leq M_y(p) \leq 1$ であるため, $0 \leq \gamma \leq 1$ と設定することにより, $f(p)$ は $[0, 1]$ の範囲に収まる. ある画素 p においてコケが発生するか否かは以下の条件式によって定まる.

$$\text{画素 } p \text{ におけるコケの発生} = \begin{cases} \text{true} & \text{if } f(p) \geq \xi_5 \\ \text{false} & \text{otherwise} \end{cases}$$

ここで, ξ_5 は $[0,1]$ の一様乱数である. 上記条件を用いることで, $f(p)$ が高いほどコケは発生しやすいが, 他の箇所にも発生するようになる.

次に, コケのレンダリング手法について述べる. 本稿では, 関東地方のコンクリートによく見られるダイダイゴケ (図 4(a)) を表現する. 図 4(a) から分かるように, ダイダイゴケは, 小さな胞子の集合体として表現できる. 始めは数点の胞子が壁面上に点在するだけである. 胞子が生長して周辺領域に新たな胞子を放つと, 着床した後に生長, 胞子の排出を繰り返すことで徐々にコケ群は生長する. 生長したコケ群は, 胞子が密に存在するため, 点の集合として表現することができる. そこで本研究では, コケの表現として, 点を基にしたフラクタル画像の生成手法 [14] を用いる. 以

下, フラクタルを用いたコケの表現手法について説明する. まず, 図 4(b) に示すような山状のフラクタルジェネレータを 1 つ用意し, 次に, 直線の基線を仮定して, 繰り返し処理によってフラクタルを作成する. しかしながら, このままでは点集合ではないためコケの画像にはほど遠い. そこで, フラクタルの線分を描く代わりに, 線分を構成する点のみを描く. このように, フラクタルジェネレータを構成する点群の描画を行うことで, コケに類似した画像が生成できる.

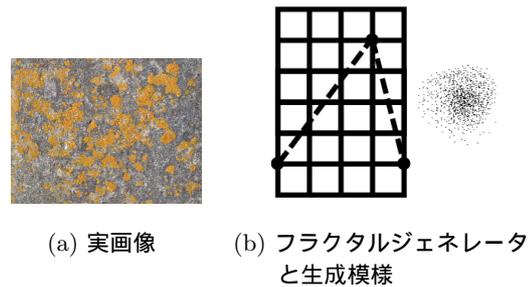


図 4: ダイダイゴケ

本論文で対象としているダイダイゴケの色はオレンジから黄緑まで多様であるため, ダイダイゴケの代表的な色としてオレンジに近い RGB=(180,130,0) と黄緑に近い RGB=(140, 130, 80) を定め, その選択はユーザに任せる. ただし, 全てのコケが同色では立体感がでないため, R, G, B の各値をランダムに変更することで多様性を持たせる.

フラクタルの大きさはジェネレータの基線長を壁領域幅の $1/40$ として生成する.

5 シミュレーション結果

本手法によるシミュレーション結果について述べる. 表 1 に, 図 5, 6, 7 および 8 での画像生成で用いたパラメータ値を示す. 全ての図で同じパラメータ値を用いていたが, 図 6 の k_s, k_a のみ, 他の図とは異なる値を用いた. また, コケの形状には図 4(b) を利用した.

図 5(a), (b), (c), (d) および (e) はそれぞれ入力画像, 4 レイヤーの重畳画像, 8 レイヤーの重畳画像, 12 レイヤーの重畳画像および 10 レイヤーの重畳とコケの重畳画像である. レイヤー化を行うことで, 初期には汚れの色は薄く, また形状も細いが, 徐々に太く, また濃くなっていく様子や汚れが周辺の色と馴染んで広がって行く様子など, 汚れの変遷を表現することができている.

表 1: シミュレーションに使われたパラメータ

記号	意味	値
m	水粒子の質量	0 ~ 120(ランダム)
I_{D0}	初期汚れ量	0 ~ 5(ランダム)
n	粒子数	600
k_s	汚れの沈着率	0.7(図 6:0.3)
k_a	壁面水分吸収率	0.5(図 6:0.7)
r	壁面の粗さ	100
δ	汚れ量の引き継ぎ率	0.01
ϵ	輝度の汚れ率	0.3
σ	位置と質量の分散	10
γ	カビの発生確率	1.5

図 9 に実画像と従来研究 [10] による結果を示す。図 9(a) にはコンクリート壁によく見られる汚れの例を示している。このように、汚れが経年変化すると、太い紐状の汚れが現れ始める。また、コンクリート内部で水分が伝搬していき、全体的に滲んだような模様になるのも特徴である。図 9(b) には従来手法による結果例を示す。この結果は文献 [10] の著者らが配布しているソフトウェア [15] を用いてレンダリングしたものである。なお、このソフトウェアにはさまざまなパラメータがあり、パラメータを調整することによってその結果は大きく異なってくる。図 9(b) は、「太い紐状の水汚れ」を発生させることを目標に、可能な限り図 9(a) に近づけるように粒子サイズや水量、吸収量などのパラメータを調整した結果である。図 9(a) と図 9(b) を比較すると分かるように、従来手法では壁面内部における水の拡散を考慮していないため、一つ一つの水汚れの境界がくっきりと現れており、図 9(b) は図 9(a) と異なることがわかる。一方、図 5(d) に示す本手法による結果では、壁面内の汚れ拡散やレイヤーの積み重ねによって、より自然な「太い紐状の水汚れ」が実現できていることがわかる。

また、図 6 に他の例を示す。(a) が入力画像、(b)-(d) にレイヤーの重ね合わせ、および汚れ量によってコケの発生量を制御した画像であり、(e) には実画像を示す。図 6 から分かるように、実画像と同様、壁面の中で水分の残量が多い壁面上部にコケが集中し、徐々にコケの発生が少なくなっている現象が表現できている。

図 7 に、実画像と本シミュレーションの結果を合成した画像を示す。図 7(a) 中、ダイダイゴケが発生している塀のブロック 4 つの内、左の 2 つの部分を切り出し、汚れおよびダイダイゴケを手動で削除して入力画像とする(図 7(b))。生成された入力画像に本手法で汚れを付加し、再び実画像にはめ込んだ結果が図 7(c) である。本シミュレーションの結果と実画像

の双方において、コケは水分が多く溜まる上部に多く発生している。また、壁面の汚れは、経年劣化によってブロック内で拡散され、水が重力で垂れた部分だけではなく、周辺にも汚れが拡散している。一方、実画像のダイダイゴケは壁の上部に密集しており、また孢子群同士の距離もより近い。また、ダイダイゴケの群集形状も、実画像では多様である。そして、実画像では水汚れが多く、コケは少ないのに対し、シミュレーション結果では水汚れに比べ、コケの発生度合いが大きい。

図 8(a) は入力画像、(b) は本手法を右下壁面にのみ適用した結果である。本手法を画像の一部にのみ適用することにより、局所的に発生する汚れを表現することができる。

6 おわりに

本稿では、景観シミュレーションで用いる画像上で経年劣化を表現する手法を提案した。はじめに水汚れが壁を伝って流れる遠藤らの手法を改良し、壁内部における水の伝搬を加えた汚れの拡散手法を提案した。また、生成された画像を時系列にレイヤーとして保存することで、汚れの変遷を可視化することができた。さらに、点のフラクタルジェネレータを用いることによって多様な形状のコケを生成し、水汚れ量に応じたコケ生成を行うことによって複数要因による経年変化も可能とした。

しかしながら、現在のモデルでは水分量が多い部分を中心にランダムにコケが発生する箇所を決めているため、実画像のようにコケが密集した複数の群を表現することは困難であるため、今後検討を行う。また、カビやひびなど、他の汚れについても今後検討を行う予定である。

参考文献

- [1] Stéphane Merillou and Djamel Ghazanfar-pour: “A Survey of Aging and Weathering Phenomena in Computer Graphics”, *Computers & Graphics*, Vol.32, Issue 2, pp.159–174, 2008.
- [2] Julie Dorsey, Holly Rushmeier: “Advanced Material Appearance Modeling”, *SIGGRAPH 2009 Courses Notes*, Article No. 3, 2009.
- [3] 広田 光一, 星野 直昭, 金子 豊久: “物理的モデルに基づくひび割れパターンの発生”, *情処学論*, Vol.37, No.12, pp.2235-2242, 1996.

- [4] Brett Desbenoit, Eric Galin, Samir Akkouche: “Modeling Cracks and Fractures”, The Visual Computer, Vol.21, No.8-10, pp.717–726, 2005.
- [5] Brett Desbenoit, Eric Galin, Samir Akkouche: “Simulating and Modeling Lichen Growth”, Computer Graphics Forum, Vol.23, No.3, pp.341–350, 2004.
- [6] Xin Yin, Tadahiro Fujimoto, Norishige Chiba: “CG Representation of Wood Aging with Distortions, Cracking and Erosion”, The Journal of the Society for Art and Science, Vol. 3, No. 4, pp.216–223, 2004
- [7] Julie Dorsey, Hand Kϕhling Pedersen, Pat Hanrahan: “Flow and Changes in Appearance”, Proc. of ACM SIGGRAPH, pp.411–420, 1996.
- [8] Yanyun Chen, Lin Xia, Tien-Tsin WONG, Xin Tong, Hujun Bao, Baining Guo, Heung-Yeung Shum: “Visual Simulation of Weathering by Gamma-ton Tracing”, ACM Transactions on Graphics, Vol.24, No.3, pp.1127–1133, 2005.
- [9] Su Xue, Jiaping Wang, Xin Tong, Qionghai Dai, Baining Guo: “Image Based Material Weathering”, Computer Graphics Forum, Vol. 27, Issue 2, pp.617-626, 2008.
- [10] Yuki Endo, Yoshihiro Kanamori, Jun Mitani, Yukio Fukui: “An Interactive Design System for Water Flow Stains on Outdoor Images”, Proc. of Smart Graphics 2010, Vol. 6133, pp.160–171, 2010.
- [11] 吉田隼斗, 向井信彦: “経年変化を伴う水の落下とカビによる汚れ画像の生成手法”, 映情学技報, Vol.34, No.45, pp.1–6, 2010.
- [12] 清田雄介, 長谷川拓哉, 千歩修: “乾湿繰り返し凍結融解が各種外装材の雨筋汚れ評価に及ぼす影響”, 学術講演梗概集, A-1, 材料施工, pp.325–326, 2005.
- [13] 柴田彩子, 湯浅昇, 大島明, 松井勇: “竣工後 33 年を経過した打ち放しコンクリート外壁に付着した生物”, 学術講演梗概集, A-1, 材料施工, pp.325–326, 2005.
- [14] 森野昌, 向井信彦, 小杉信: “フラクタルを利用した汚れ画像の生成手法”, 映情学技報, Vol.31, No.59, pp.29–34, 2007.
- [15] 筑波大学非数値処理アルゴリズム研究室: <http://www.npal.cs.tsukuba.ac.jp/projects/waterstain/>

張 英夏



1998年, 韓国梨花女子大学卒業。2004年, 東京工業大学大学院情報理工学研究科博士後期課程修了。Ph.D. 同年, 同大研究員。2006年, 同大学院情報理工学研究科助手。2007年, 同大学院情報理工学研究科助教。2012年, 東京都市大学知識工学部講師。現在に至る。画像処理の研究に従事。正会員。

吉田 隼人



2009年, 武蔵工業大学(現東京都市大学)工学部コンピュータ・メディア工学科卒業。2011年, 同大学院工学研究科修了。同年, 三晃印刷(株)入社。在学中は画像処理の研究に興味を持つ。

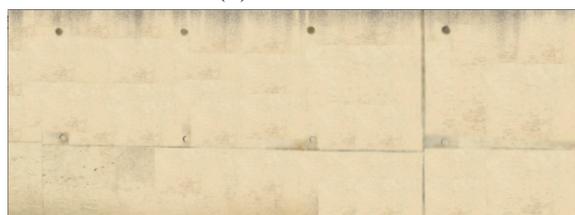
向井 信彦



1983年, 大阪大学基礎工機械卒業。1985年, 同大学院博士前期課程修了。同年, 三菱電機(株)入社。1997年, コーネル大コンピュータサイエンス修士課程修了。2001年, 大阪大学大学院博士後期課程修了。博士(工学)。2002年, 武蔵工業大学(現東京都市大学)工学部電子情報工学科助教授。2007年, 同大知識工学部情報科学科教授。画像工学に関する研究に従事。正会員。



(a) 入力画像



(b) 4 レイヤーの重畳画像



(c) 8 レイヤーの重畳画像



(d) 12 レイヤーの重畳画像



(e) 10 レイヤーの重畳とコケの重畳表示画像

図 5: シミュレーション結果例 (その 1)



(a) 入力画像



(b) 5 レイヤー重畳とコケの重畳画像



(c) 10 レイヤー重畳とコケの重畳画像



(d) 15 レイヤー重畳とコケの重畳画像



(e) 実画像

図 6: シミュレーション結果例 (その 2)



(a) 実画像



(b) シミュレーション対象領域



(c) 結果画像

図 7: シミュレーション結果例 (その 3)



(a) 入力画像



(b) シミュレーション結果

図 8: シミュレーション結果例 (その 4)



(a) 実画像



(b) 従来研究 [10] による結果

図 9: 実画像および従来研究による結果