

I-046

垂直面への着雪を考慮した積雪情景のリアルタイム表現 Real Time Rendering of Wind-Driven Snowfall on Vertical Surfaces

森谷 友昭†
Tomoaki Moriya

高橋 時市郎†
Tokiichiro Takahashi

1. まえがき

近年、3次元コンピュータグラフィックスを用いて降雪・積雪情景を表現する手法が数多く研究されている。筆者らは Precomputed Radiance Transfer (PRT) を使用して、風向きおよび遮蔽物を考慮した積雪量を近似的に算出することで、風向きを考慮した地形への積雪のリアルタイムレンダリングを可能とした[1]。本提案手法では、従来手法に加えて、法線マップを基に地形垂直面への積雪量を算出することで、壁面などへの雪粒の付着（以下、着雪と呼ぶ）を表現する。これによって、より現実に近い積雪情景をリアルタイムに表現することができる。

2. 従来手法

筆者らが提案した従来手法において、積雪量は以下に述べる方法で算出される。

まず、積雪の対象となる地面および建造物を含んだ地形メッシュを UV マップ展開し、作成された UV マップ上で積雪量算出を行う。事前に、各地点における遮蔽物を考慮した拡散反射光を算出し、これを各地点での積雪量と見立てる。ただし、算出された拡散反射光は PRT の方法により圧縮される。

地形上のある位置 \vec{x} における単位時間あたりの積雪量 $D(\vec{x})$ は、風向きベクトルを \vec{w} 、そこでの法線ベクトルを \vec{N} とするとき、次式 (1) によって算出される。

$$D(\vec{x}) = s \cdot L(\vec{x}, \vec{w}) (\vec{N} \cdot \vec{G}_n) \max\{1 - E(\vec{x}), 0\} \quad (1)$$

ここで、 s は地形のスケールに応じて積雪量を調整するためのスカラー値、 $L(\vec{x}, \vec{w})$ は PRT により復元された拡散反射光、 \vec{G}_n は重力方向逆の向きとなるベクトル、 $E(\vec{x})$ は地形の縁に現れる積雪斜面を生成するための値である。図1に、従来手法により生成された積雪情景の例を示す。図1では、画面手前から奥へ、小屋内部へ雪が吹き込むような風向きが設定されているため、小屋内部の椅子などにも積雪していることが見て取れる。

ところで、従来手法では地形への積雪量を、式 (1) に示すとおり \vec{N} と \vec{G}_n の内積により調節している。これは、地形表面が垂直に近いほど積雪量は少なることを表し、壁など垂直面では積雪が全く起きないことを意味している。しかしながら、実際の積雪情景では、垂直面に存在する表面の細かい凹凸によって垂直面への積雪が発生する。これは、垂直面への着雪となって現れる。

3. 提案手法

垂直面への着雪を表現するため、本手法では、式 (1) における法線ベクトル \vec{N} を、法線マップから算出した法線ベクトル \vec{N}_m に置き換える。法線マップには、表面の凹

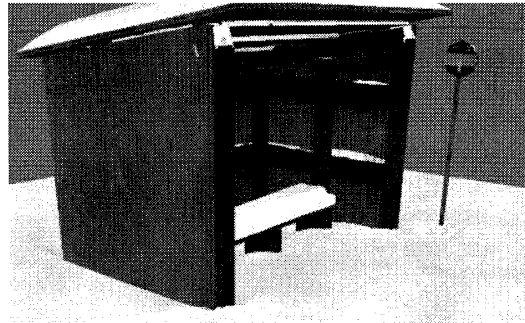
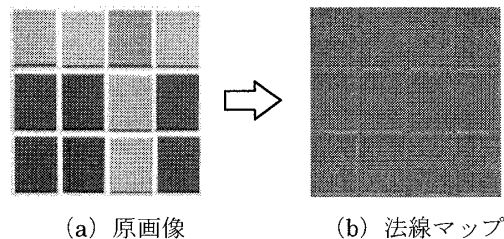


図1 従来手法による積雪情景生成結果例



(a) 原画像

(b) 法線マップ

図2 法線マップ生成例

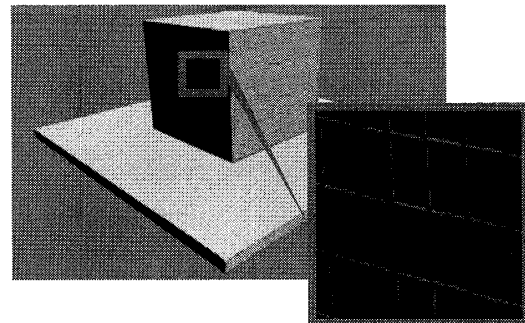


図3 法線マップからの法線算出例

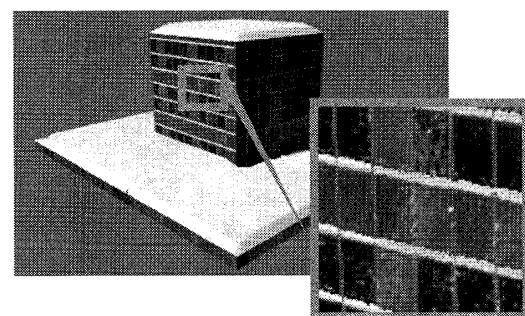


図4 レンダリング結果例 (単純地形)

†東京電機大学, Tokyo Denki University

凸を模したデータを与えておく。以下に提案手法詳細を述べる。

まず、地形にマッピングされるテクスチャを基に法線マップを生成する。今回、法線マップの生成にはグラフィック編集ソフトのプラグイン[2]を用い、テクスチャのRGB成分から法線マップを生成している(図2)。

また、法線マップを使用するため、地形メッシュの各頂点において接線、従法線を算出しておく。法線マップからサンプリングした法線を \vec{N}_s 、地形メッシュ頂点の法線 \vec{N} 、接線 \vec{T} 、従法線 \vec{B} とすると、 \vec{N}_m は次式(2)より求まる。

$$\vec{N}_m = \begin{bmatrix} Nm_x \\ Nm_y \\ Nm_z \end{bmatrix} = \begin{bmatrix} B_x & B_y & B_z \\ T_x & T_y & T_z \\ N_x & N_y & N_z \end{bmatrix} \begin{bmatrix} N_{sx} \\ N_{sy} \\ N_{sz} \end{bmatrix} \quad (2)$$

式(1)の \vec{N} を、算出された法線 \vec{N}_m に置き換え積雪量を算出する。図2(b)に示した法線マップを基に算出された、地形メッシュ法線を図3に示す。

図3は、各面、法線X方向の値を赤成分、Y方向を緑成分、Z方向を青成分で可視化した結果となる。上方を向く面は法線がY方向(0, 1, 0)であることから緑色で示している。垂直面に注目すると、法線マップから算出された法線により、垂直面にもY方向に近い法線が生成されている。

4. レンダリング結果

提案手法によりレンダリングした結果例を示す。

図4は、図3に示した法線を基に積雪量を算出した結果である。図4では、画面左上から右下に向かう風向きが設定されており、地形左側、雪が吹き付ける垂直面では着雪しており、逆に風に対して陰となる垂直面では、着雪していないことが分かる。

本手法では前述のとおりUVマップ上で積雪量計算を行うため、レンダリング品質はその解像度に大きく依存する。各UVマップ解像度におけるレンダリング結果を図5に示す。図5から、UVマップ解像度が高くなるにつれ、細部の表現が行われていることが見て取れる。また、表1に示す実行環境・各UVマップ解像度における平均FPSを表2に示す。

5. まとめ

法線マップを基に地形垂直面への積雪量を算出することで、壁面などへの着雪を表現した。これによって、より現実に近い積雪情景をリアルタイムに表現することができた。

参考文献

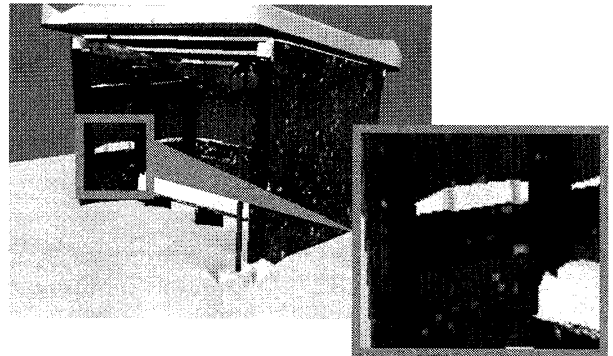
- [1] 森谷友昭, 高橋時市郎: “PRTを用いた積雪情景のリアルタイム生成”, Visual Computing/グラフィクスとCAD合同シンポジウム2010, No.4 (2010).
- [2] GIMP normalmap plugin
<http://code.google.com/p/gimp-normalmap/>

表1 実行環境

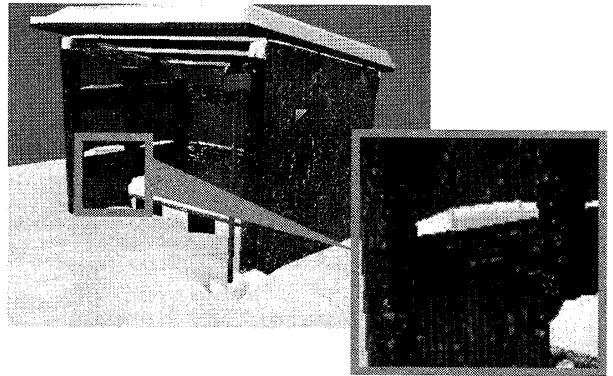
CPU	Intel Core2 Extreme Q6850
RAM	2GB
GPU	ATI Radeon HD 5870
画面サイズ	1280×1024画素
地形ポリゴン数	3424

表2 平均実行速度

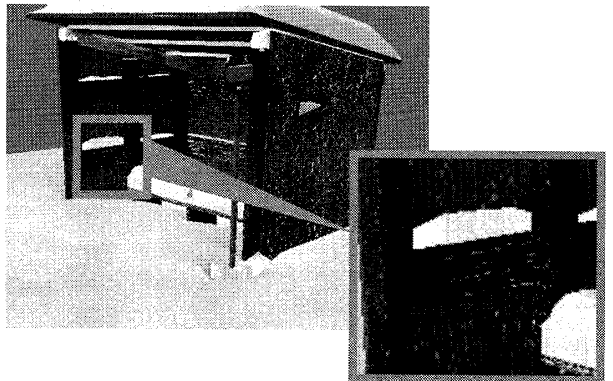
UVマップ解像度[画素]	平均実行速度[FPS]
512x512	57
1024x1024	43
2048x2048	6



(a) UVマップ解像度: 512x512



(b) UVマップ解像度: 1024x1024



(c) UVマップ解像度: 2048x2048

図5 レンダリング結果例(バス停)