

## フラクタル圧縮を用いたテクスチャの超解像化

小南 悠人<sup>†</sup>  
東京都立大学<sup>†</sup>

向井 智彦<sup>†</sup>  
東京都立大学<sup>†</sup>

## 1 はじめに

近年、ディスプレイの高解像度化に伴い、リアルタイム 3DCG においてもテクスチャの高精細化が求められている。高解像度なテクスチャでは転送負荷が高くなり、必要メモリ量も増大するため、映像表示時に即座に超解像度化する技術が望ましい。しかし、バイリニア補完やバイキュービック補完のような従来の超解像度化技術では、拡大するに従い不鮮明になり、また SRCNN[3] などの深層学習を用いた超解像度化では、多くの計算を要するという課題がある。本手法では、従来手法よりも高品質な結果を高速に得るために、フラクタル符号化 [1] を応用することでレンダリング時に超解像度化を行う。提案法が備える反復関数系の特性を利用して、より鮮明な超解像度化を実現する。

## 2 提案法

## 2.1 フラクタル符号化

本研究では GPU 上にも実装可能な超解像化を実現するため、ミップマップピラミッドを用いたフラクタル符号化を行う。具体的には Barnsley の手法 [1] と同様に、まず画像全体をドメインと呼ばれるブロックに分割する。その後、各ドメインと類似するブロックを探索し、それをレンジとする。なお、今回は Stachera らの研究 [2] を参考に、ドメインブロックのサイズを  $2 \times 2$ 、レンジブロックの探索範囲をミップマップレベル 1 のテクスチャとした。

また、図 1 に示すように、ミップマップレベル 0 における座標を  $(p, q)$ 、ミップマップレベル 1 における座標を  $(r_0, s_0)$  と表す。また、 $L_i(x, y)$  をミップマップレベル  $i$  における座標  $(x, y)$  のピクセル値とする。このと

き、ドメイン-レンジ間の類似度を式 (1) に示す差分二乗和として定義する。

$$\sum_{i=0}^1 \sum_{j=0}^1 (L_0(p+i, q+j) - L_1(r_0+i, s_0+j))^2 \quad (1)$$

そして、任意の  $(p, q)$  に対して、式 (1) を最小化するような座標  $(r_0, s_0)$  を総当たり探索する。このとき、 $(p, q)$  と  $(r_0, s_0)$  はドメイン-レンジの対応関係にあるとみなす。

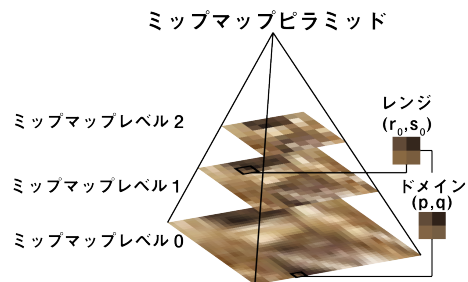


図 1 ミップマップレベル 1 のテクスチャから式 (1) を最小化するレンジを探索

ここで、ドメイン、レンジのサイズがそれぞれ  $2 \times 2$  であることから、 $(p, q)$  と  $(r_0, s_0)$  はいずれも 2 の倍数である必要がある。このとき、すべての  $(p, q)$  に対して走査すると以下に示す式 (2) が成り立つ。

$$\mathbf{D}(p, q) = \mathbf{R}(r_0, s_0) + \text{diff}_{r_0, s_0} \quad (2)$$

ただし、 $\mathbf{D}(p, q)$  はミップマップレベル 0 での  $2 \times 2$  ピクセルブロック、 $\mathbf{R}(r_0, s_0)$  はミップマップレベル 1 での  $2 \times 2$  ピクセルブロックを表す。また、単一の色値  $\text{diff}_{x, y}$  はドメイン-レンジ間の色誤差を補正するための差分項であり、次の式 (3) で求められる。

$$\text{diff}_{x, y} = L_1\left(\frac{p}{2}, \frac{q}{2}\right) - L_2\left(\frac{r_0}{2}, \frac{s_0}{2}\right) \quad (3)$$

この式 (3) は、求めた  $(p, q)$ 、 $(r_0, s_0)$  に対してそれぞれミップマップレベルを 1 上げることで、単一のピクセル値を参照する操作に対応する。

Super-Resolution of Textures Using Fractal Compression

<sup>†</sup> Yuto Kominami

<sup>†</sup> Tomohiko Mukai

<sup>†</sup> Tokyo Metropolitan University

以上の操作を行うことで、ミップマップレベル0の座標  $(p, q)$  におけるピクセル値は、それぞれ  $r_0, s_0, \text{diff}_{r_0, s_0}$  で表すことができる。

### 2.2 フラクタル超解像度化

上記のドメイン-レンジ対応関係を用いてリアルタイム超解像度化を行う。フラクタル符号化で圧縮されたテクスチャを展開する際、 $(p, q)$  に対応する座標  $(r_0, s_0)$  と差分  $\text{diff}_{r_0, s_0}$  は、ドメイン-レンジ対応関係により取得できるため、ミップマップレベル0のピクセルブロックを復号できる。この一連の処理を全てのドメインに対し適用することで、ミップマップレベル0のテクスチャ全体を復号化する。

また、取得した座標  $(r_0, s_0)$  を用いて新たに  $p_1 = 2r_0, q_1 = 2s_0$  とすると、座標  $(p_1, q_1)$  に対応する  $(r_1, s_1)$  が存在し、次の式 (4) が成り立つ。

$$\mathbf{D}(e, f) = \mathbf{R}(r_1, s_1) + \text{diff}_{r_0, s_0} + \text{diff}_{r_1, s_1} \quad (4)$$

ただし、 $\mathbf{D}(e, f)$  はミップマップレベル0の2倍の解像度をもつテクスチャの座標を入力とした  $2 \times 2$  ピクセルブロックである。この式を用いることにより、 $L_0$  テクスチャの2倍の解像度を持つ画像を生成できる。また、 $(r_1, s_1)$  から新たに  $p_2 = 2r_1, q_2 = 2s_1$  が求まり、 $L_0$  テクスチャの4倍のテクスチャが得られる。

このような、式 (2)、式 (4) の繰り返しを一般項として表すと以下の式 (5) となる。

$$\mathbf{D}_i(x, y) = \mathbf{R}(r_i, s_i) + \sum_{k=0}^i \text{diff}_{r_k, s_k} \quad (5)$$

$\mathbf{D}_i(x, y)$  は、ミップマップレベル0の解像度を  $2^i$  倍したテクスチャの  $(x, y)$  座標を入力とした  $2 \times 2$  ピクセルブロックである。この式を用いて、任意の  $i$  に対して  $2^i$  倍のテクスチャ超解像度化を行う。

### 2.3 実装

GPUでのデータの扱いを容易にするために、ここでは  $r_0, s_0, \text{diff}_{r_0, s_0}$  をテクスチャデータとして保存する。すなわち、ピクセル座標  $(r_0, s_0)$  を以下に示す式 (6) のように  $(p, q)$  に対しての相対値である  $(\delta p, \delta q)$  で表す。

$$r_0 = \frac{p}{2} + \delta p, s_0 = \frac{q}{2} + \delta q \quad (6)$$

このとき、 $\delta p$  と  $\delta q$  はともに  $[-125, 126]$  の整数となり、 $r_0, s_0$  を RG 各 8bit の Position テクスチャで保存できる。また、 $\text{diff}_{r_0, s_0}$  は RGB 各 8bit の Diff テクスチャとして保存する。展開する際には、解像度  $2^i$  倍に対して、

Diff と Position テクスチャから合計  $2 \times i + 3$  回サンプリングを行う。

## 3 実験結果

既存の超解像度化手法と提案法を比較した。図3は原画像、図4から図6は図2の原画像をバイキュービック、LapSRN、及び提案法で、それぞれ8倍に拡大したテクスチャの一部を示す画像である。



図2 原画像



図3 原画像

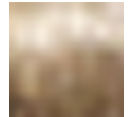


図4 Bicubic



図5 LapSRN



図6 提案法

図6の提案法では、原画像に存在する繊維質の特徴が残されていることが分かる。

## 4 まとめ

本研究では、GPUで展開可能なフラクタル超解像度化のためのアルゴリズムを考案した。提案法の主な利点は、レンダリング時の計算量や、テクスチャサンプリング数が少なく、ランダムアクセスが可能な点である。今後の課題としては、より自然なテクスチャを生成するため、隣接ピクセルの情報を考慮した探索方法の改良や、デノイザーの導入、より多くのミップマップレベルを用いることなどが挙げられる。

## 謝辞

本研究はプラチナゲームズ株式会社の支援を受けた。

## 参考文献

- [1] M.F.Barnsley, Fractals Everywhere. Academic Press, 1988.
- [2] J.Stachera, P.Rokita, Fractal-based hierarchical mip-pyramid texture compression, Machine Graphics & Vision International Journal, 2006
- [3] C.Dong, C.C.Loy, K.He, X.Tang. Learning a Deep Convolutional Network for Image Super-Resolution, European Conference on Computer Vision, 2014