

領域画像マッチングのための初期探索法

山下 幸彦（東京工業大学），若原 徹（法政大学）

平成30年9月21日

内容

- 背景
- スライディング離散フーリエ変換によるガウス平滑化
- 特徴量記述
- フーリエ記述エッジ方向ヒストグラム
- マッチング
- 計算機実験
- まとめ

1 背景

- 画像マッチングは画像処理における最も重要な手法の一つ
- SIFTなどによる特徴点マッチング
- 相関を最大化する領域マッチング
 - 若原らによる GAT(Global Affine Transform)/GPT(Global Projection Transform) 相関マッチング法
 - 繰り返し法に基づく**正確なマッチング**が可能
 - GPT相関法に同一エッジ方向平均最小距離使った k NN による MNIST の手書き文字認識で、認識率 **99.7%** を達成
 - **問題点**：whole-to-part マッチングに適用すると、初期探索に計算時間がかかる。

2 三角関数のスライディング畳み込み和によるガウス平滑化

- 1次元離散点上の関数で記述する。
- ガウス関数は標準偏差の数倍で0に急速に近づくため、加算区間を $k \in [-K, K]$ で考える。
- $G[n] = e^{-(\pi n/K)^2/(2\sigma^2)}$ (画像上の標準偏差は $\sigma K/\pi$)
- GAT/GPTテンプレート作成を高速化するためには、関数 $f[n]$ に対して、次の値が求めれば良い。

$$\sum_{k=-K}^K G[k]f[n-k] \quad (\text{ガウス関数畳み込み})$$

$$\sum_{k=-K}^K kG[k]f[n-k] \quad (\text{微分ガウス関数畳み込み})$$

$$\sum_{k=-K}^K k^2G[k]f[n-k] \quad (\text{LoGの計算に必要})$$

- $\beta = \pi/K$ とおき，次の近似を考える。

$$G[k] \simeq \sum_{p=0}^P a_p \cos(\beta pk)$$

$$kG[k] \simeq \sum_{p=1}^P b_p \sin(\beta pk)$$

$$k^2G[k] \simeq \sum_{p=0}^P d_p \cos(\beta pk)$$

- P は DFT を打ち切る次数である。(計算量を減らすため。)
- 想定している P は，**2, 4程度**。

- 三角関数のスライディング畳み込み和を次式で定義する。

$$c_p[n] = \sum_{k=-K}^K f[n-k] \cos(\beta pk)$$

$$s_p[n] = \sum_{k=-K}^K f[n-k] \sin(\beta pk)$$

- ガウス関数やその微分と信号の畳み込み和は，次のように近似できる。

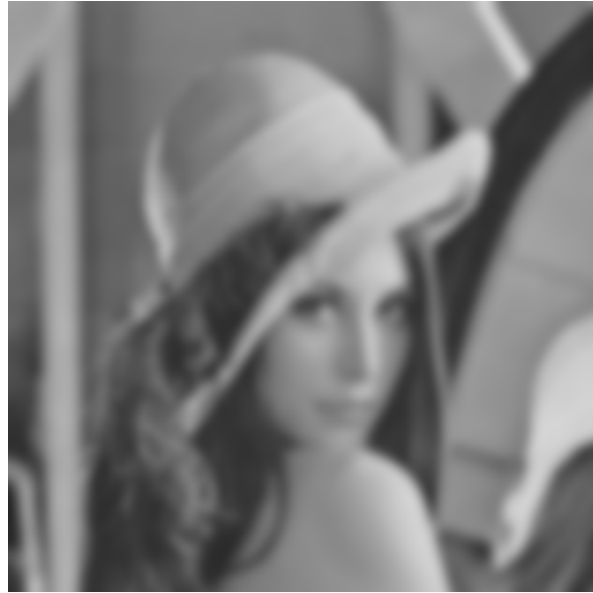
$$\sum_{k=-K}^K G[k] f[n-k] \simeq \sum_{p=0}^P a_p c_p[n]$$

$$\sum_{k=-K}^K k G[k] f[n-k] \simeq \sum_{p=1}^P b_p s_p[n]$$

$$\sum_{k=-K}^K k^2 G[k] f[n-k] \simeq \sum_{p=0}^P d_p c_p[n]$$



原画像



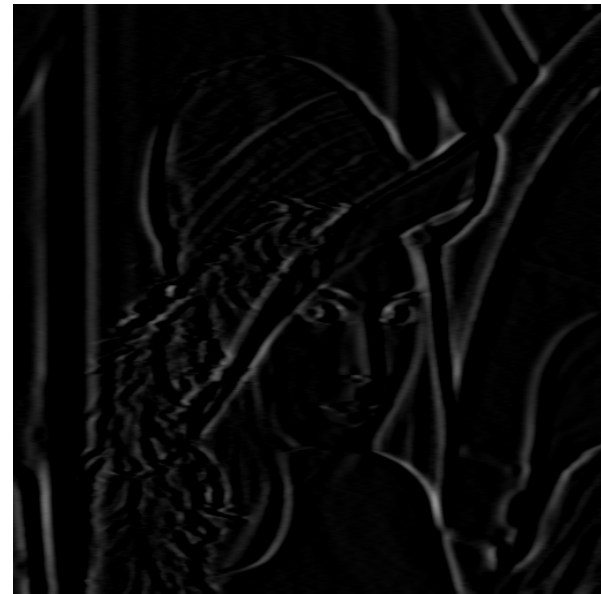
ガウスボケ画像



横方向微分画像



縦方向微分画像



LoG画像

2.1 三角関数のスライディング畳み込み和の高速計算

- カーネル積分 (複素数を使って記述) (Kober, 2011) :

$$u[n] = \sum_{k=1}^n f[k] e^{i\beta p k}$$

- $u[n]$ から, 三角関数のスライディング畳み込み和は次式で求まる。

$$c_p[n] + i s_p[n] = e^{-i\beta p n} (u[n + K] - u[n - K - 1])$$

- $u[n]$ は, 次の漸化式で計算できる。

$$u[n] = u[n - 1] + f[n] e^{i\beta p n}$$

- 三角関数を1周期分記憶して, 信号にかけていく必要がある。
- 少し変えて, **IIRフィルタ**を使う。

$$v[n] = e^{-i\beta p} v[n - 1] + f[n]$$

- **三角関数のスライディング畳み込み和は次式で求まる。**

$$c_p[n] + i s_p[n] = (-1)^p (v[n + K] - v[n - K] + x[n - K])$$

- 杉本(2017)らは2次のフィルタを使うことを提案している。

先の $v[n] = e^{-i\beta p}v[n-1] + f[n]$ の n に $n+1$ を代入すると,

$$v[n+1] = e^{-i\beta p}v[n] + f[n+1]$$

となり, 前の式に $e^{i\beta p}$ をかけて引くと, 次式が得られる。

$$v[n+1] = 2\cos(\beta p)v[n] + v[n-1] + f[n+1] - e^{i\beta p}f[n] \quad (1)$$

- \cos の項, \sin の項だけを計算することができる。
- 2階の微分みたいになっているため, 計算精度は?

積分値でなく, **直接, 区間和を計算**することもできる。

$$u'[n] = u'[n-1] + f[n]e^{i\beta pn} - f[k-2K-1]e^{i\beta p(n-2K-1)}$$

$$\tilde{v}'[n] = e^{-i\beta p}\tilde{v}'[n-1] + f[n] - f[n-2K]$$

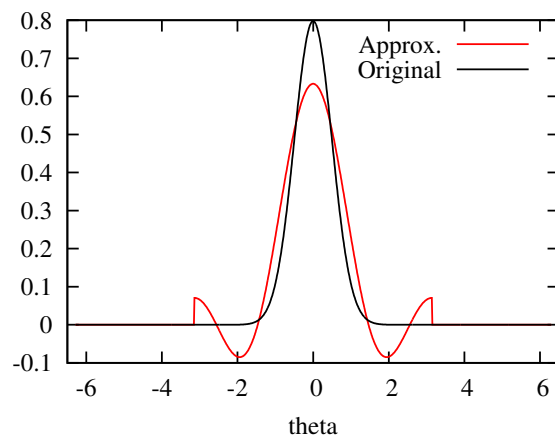
$$v'[n+1] = 2\cos(\beta p)v'[n] - v'[n-1] \\ + f[n+1] - e^{i\beta p}f[n] - e^{-i\beta p}f[n-2K] + f[n-2K-1]$$

- K, p を整数倍した場合への, 積分値の再利用はできなくなる。

3 計算誤差の検討

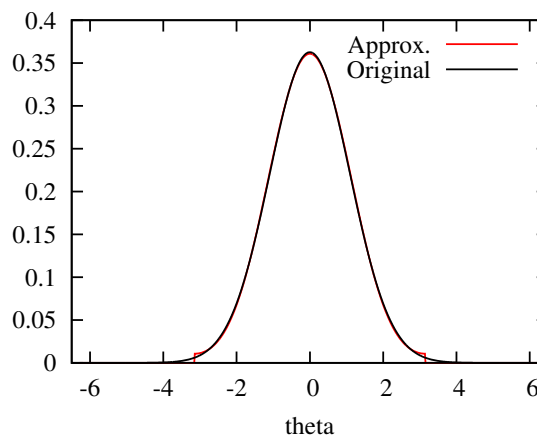
3.1 区間打ち切り誤差, 次数打ち切り誤差

- σ は, 区間 $[-K, K]$ を $[-\pi, \pi]$ としたときの標準偏差。
- 打ち切り誤差は σ に大きく依存し, K にはあまり依存しない。
- 画像上の平滑化の標準偏差は, K で調整できるため, σ は固定できる。
- σ を変えて打ち切り誤差を調べる。



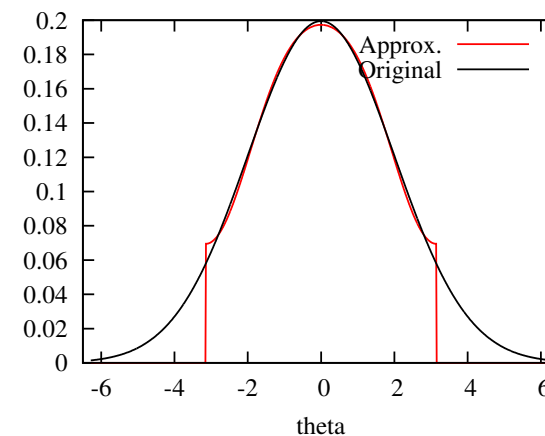
$$\sigma = 0.5$$

(次数打ち切り誤差 : 大)



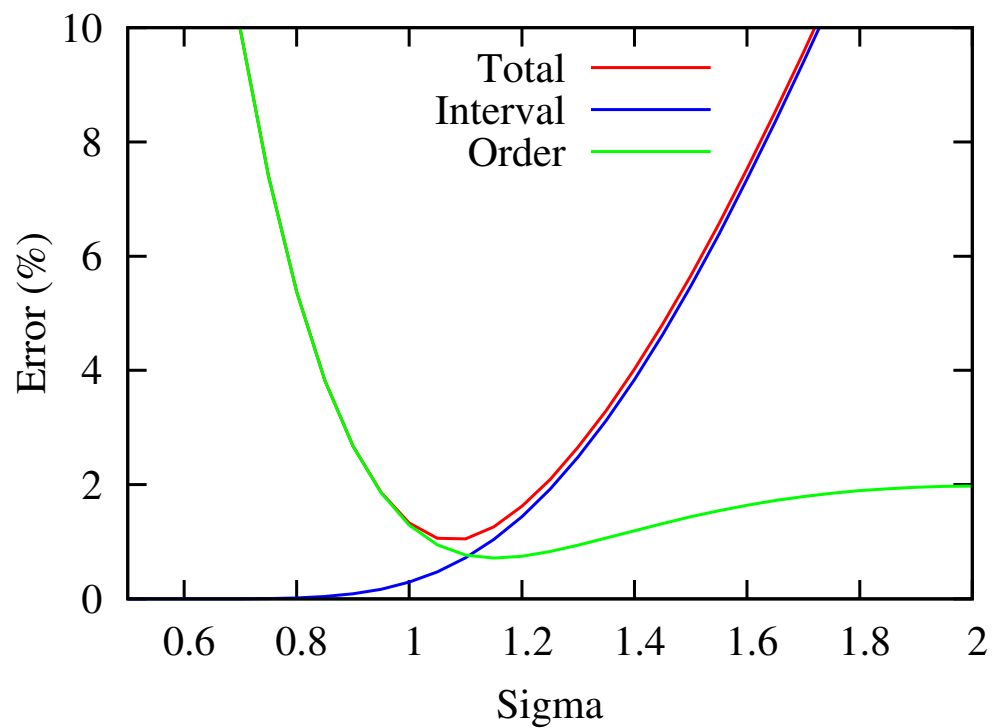
$$\sigma = 1.1$$

(区間打ち切り誤差 : 大)

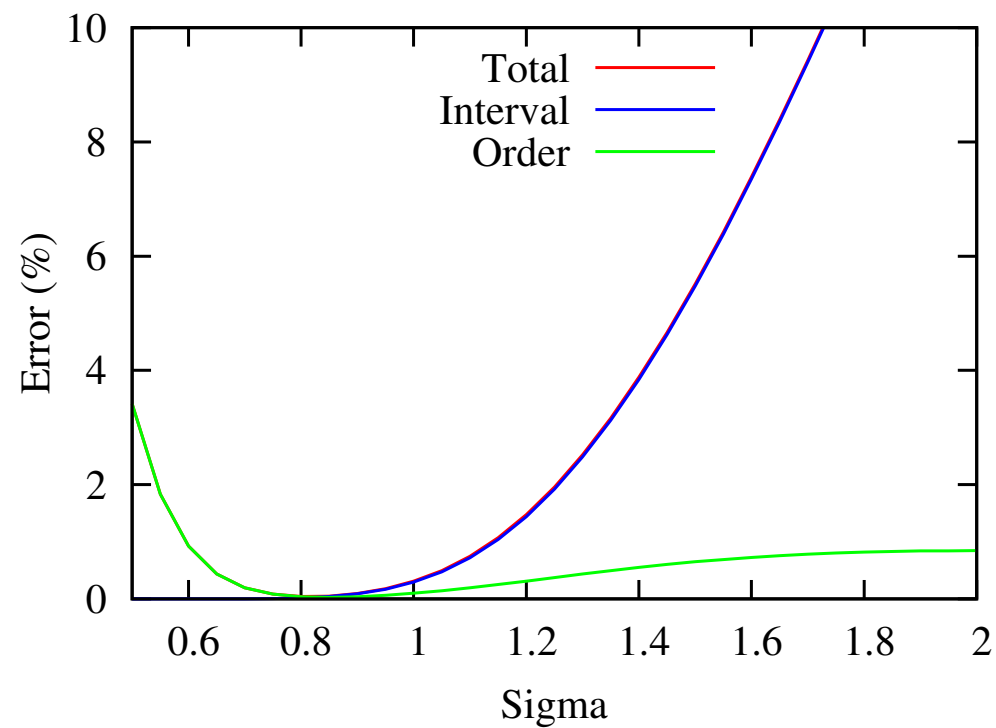


$$\sigma = 2.0$$

3.2 ガウス関数の区間打ち切り誤差, 次数打ち切り誤差

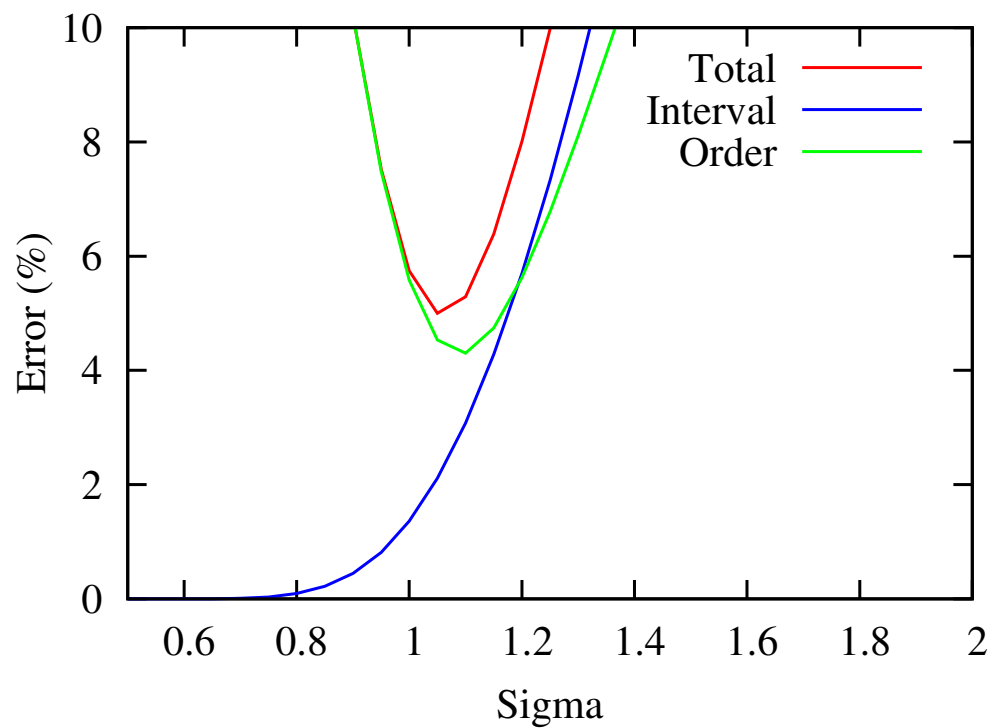


$$G[n], K = 256, P = 2$$

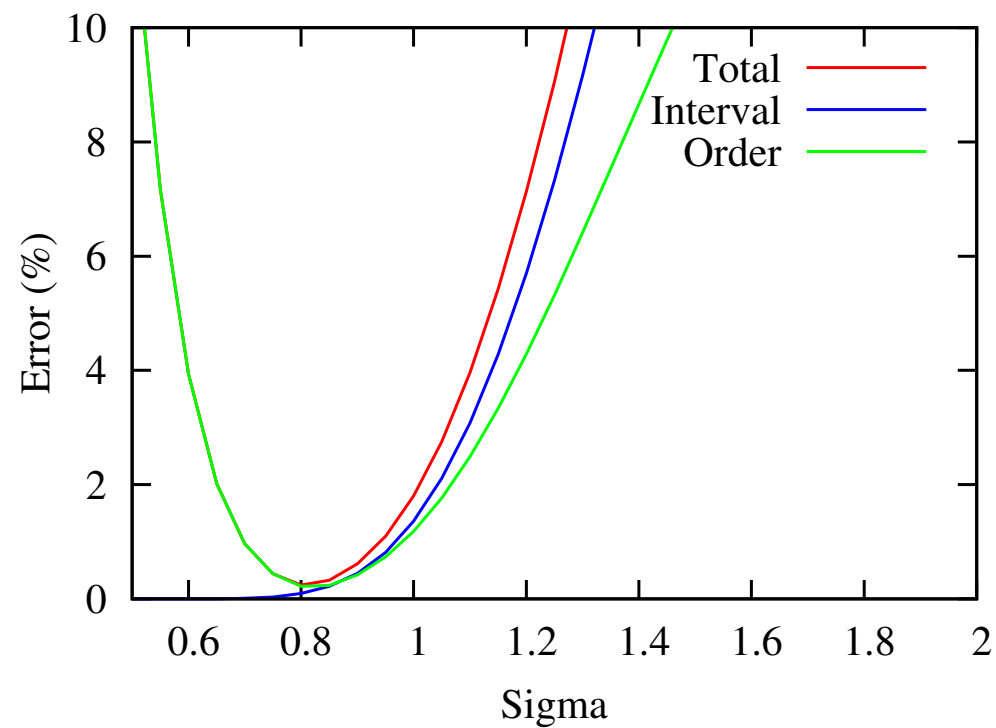


$$G[n], K = 256, P = 4$$

3.3 微分ガウス関数の区間打ち切り誤差, 次数打ち切り誤差

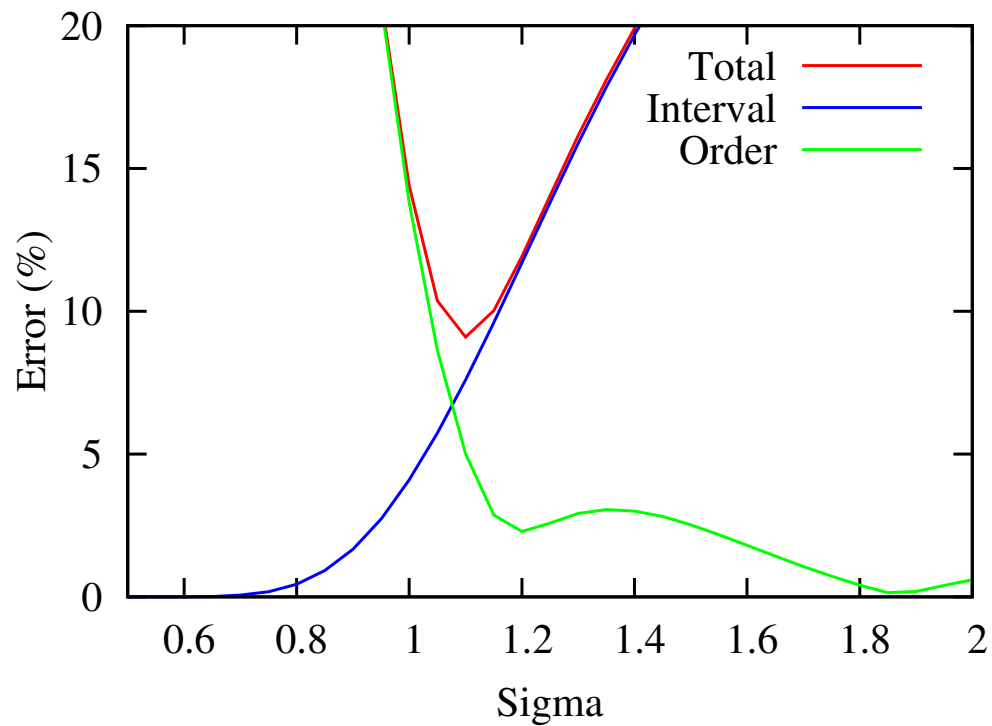


$$nG[n], K = 256, P = 2$$

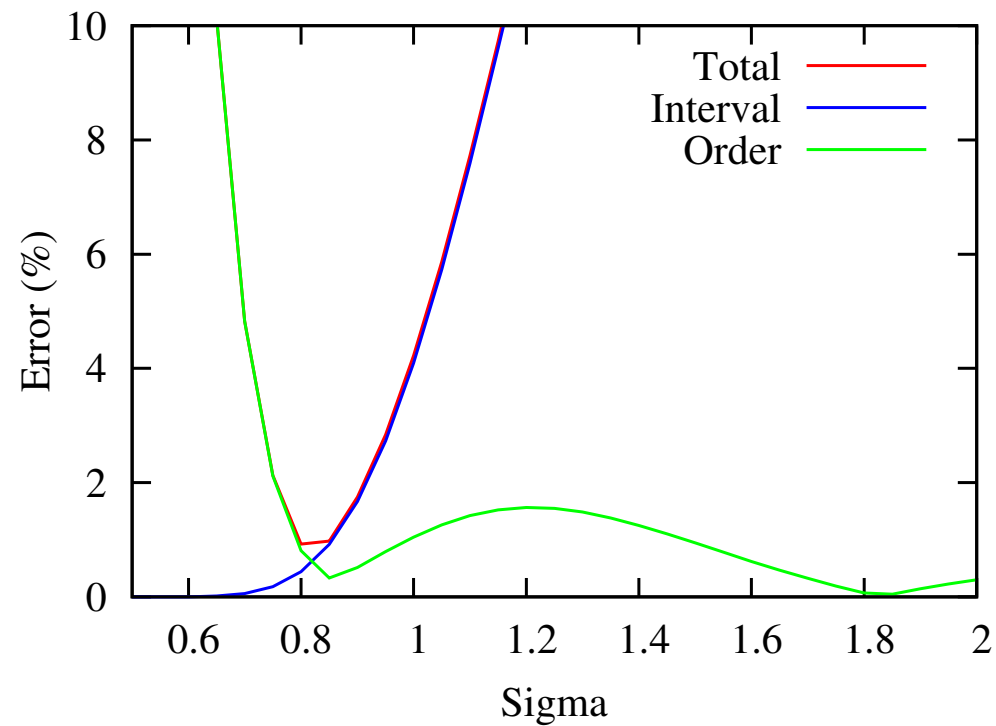


$$nG[n], K = 256, P = 4$$

3.4 2階微分ガウス関数の区間打ち切り誤差, 次数打ち切り誤差



LoG, $K = 256$, $P = 2$

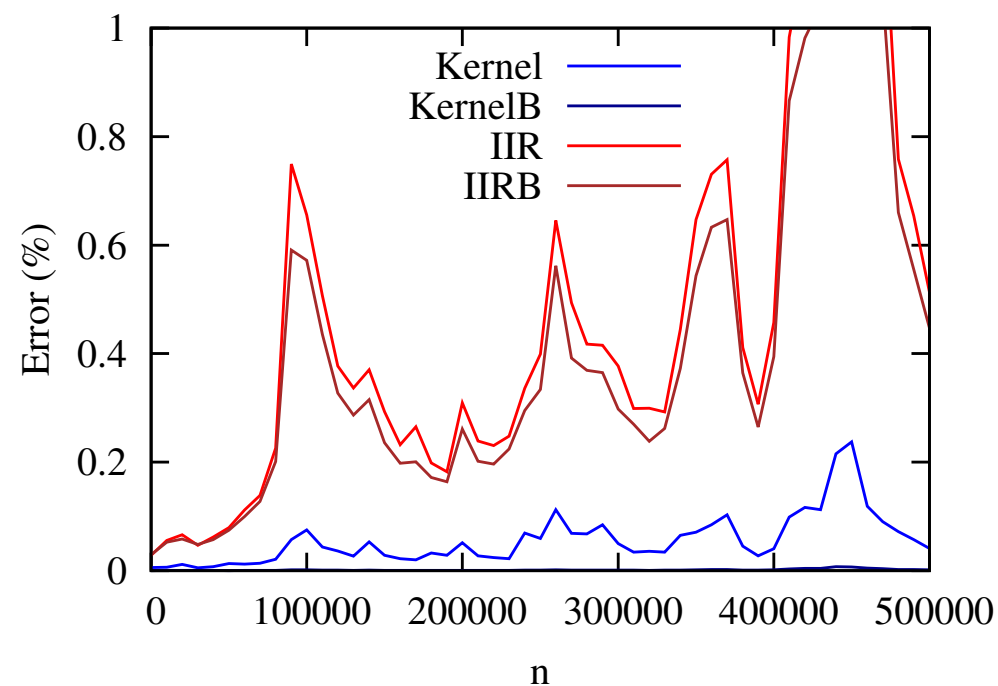
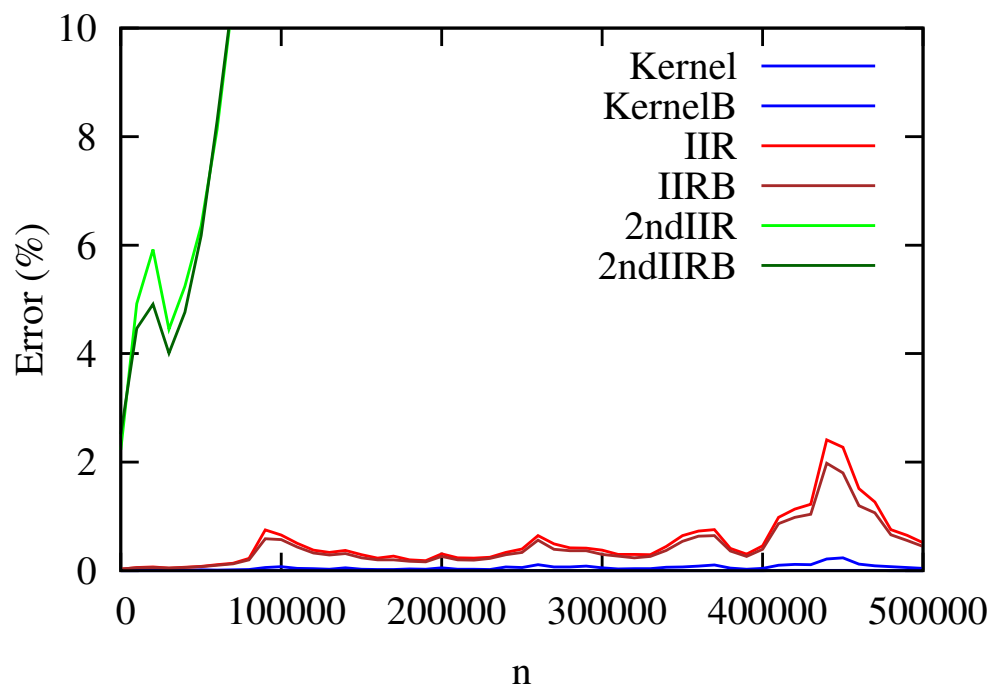


LoG, $K = 256$, $P = 4$

3.5 単精度浮動小数点演算誤差

- 安価なGPUは，単精度浮動小数点演算器しか積んでいない。
- この場合，倍精度との計算速度比は30倍程度。
- 原信号LenaとBarbaraを連結， $K = 256$ ， $P = 1$ のcos成分の

$$\frac{\text{区間長10,000での最大誤差}}{\text{区間長10,000での値の平方根2乗平均}} \times 100$$



(右グラフの縦スケールを拡大)

4 初期探索法

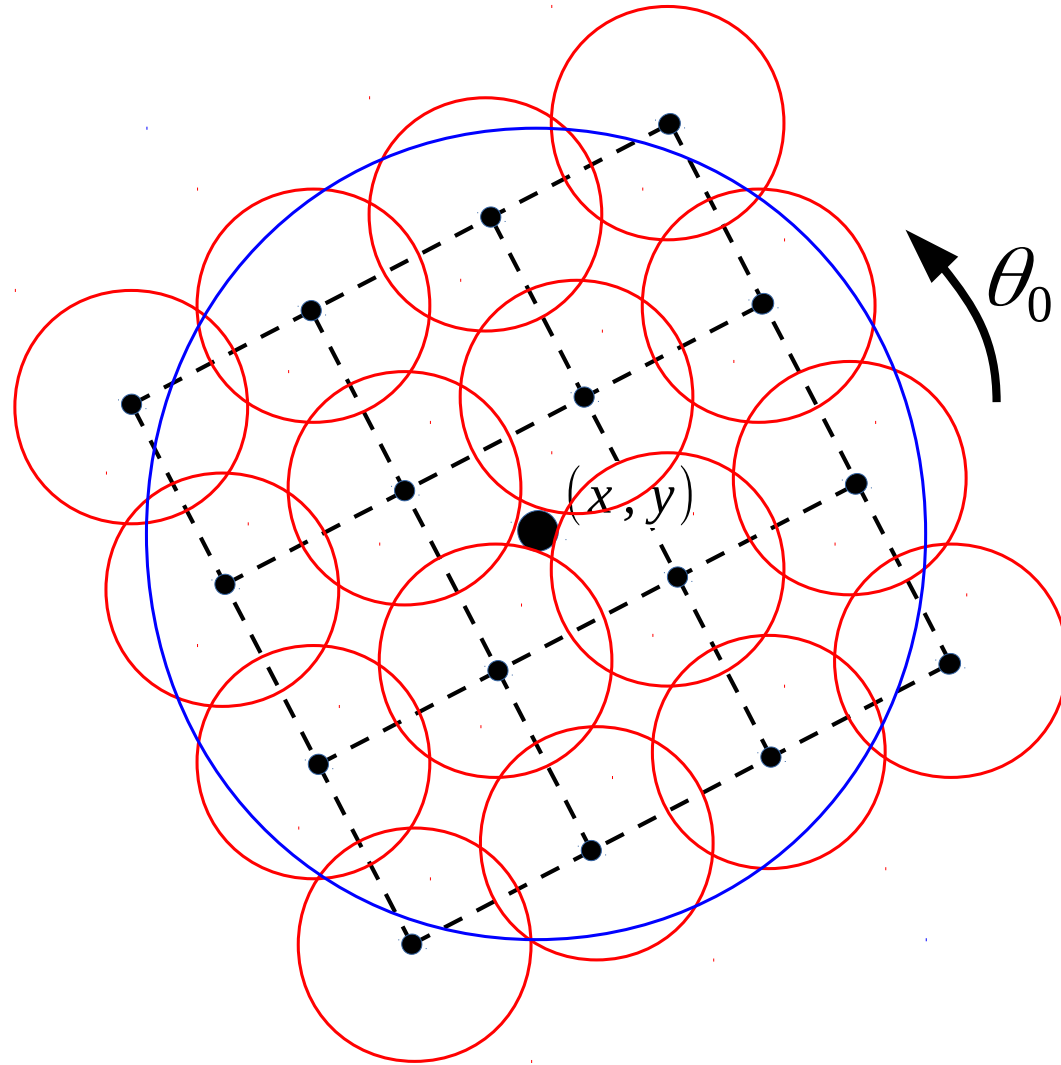
概要

- SIFTの特徴量に準拠した領域特徴量でマッチングを行う。
- エッジ方向ヒストグラムをフーリエ記述する。

4.1 特徴量記述アルゴリズムの概要

領域のスケールを S とする。

1. 画像からスケール $S/8$ でガウス平滑化した微分画像を求める。
2. 微分画像から各点のフーリエ記述エッジ方向ヒストグラムを求める。
3. 各点のフーリエ記述エッジ方向ヒストグラムをスケール S と $S/4$ でガウス平滑化する。
4. スケール S の平滑化フーリエ記述エッジ方向ヒストグラムから、各標本点における基準方向を求める。
5. 標本点周りの16点で、スケール $S/4$ のガウス平滑化フーリエ記述エッジ方向ヒストグラムを基準方向で回転し、並べてベクトルを作成する。
6. そのベクトルのノルムを1で正規化したものを特徴ベクトルとする。



標本点, 基準方向, 特徴量抽出点

- 特徴量の次元数 : $9 \times 16 = 144$

4.2 フーリエ記述エッジ方向ヒストグラム

- エッジ方向ヒストグラム $D(\theta)$ は、ある領域内の θ 方向のエッジの強さの（重み付き）和である。
- エッジ方向ヒストグラムを、 Q 次のフーリエ級数で記述する。

$$D(\theta) \simeq \sum_{q=-Q}^Q d_n e^{iq\theta}$$

- $d_{-q} = \overline{d_q}$ が成立するため、実数での自由度は $2Q + 1$ である。
- $I_X(x, y)$, $I_Y(x, y)$: 入力画像のガウス平滑化微分画像（横，縦）
- $I_1(x, y) = I_X(x, y) + iI_Y(x, y)$: 複素数表示
- 1 点 (x, y) におけるエッジ方向ヒストグラム :

$$|I_1(x, y)| \delta(\theta - \arg(I_1(x, y)))$$

- 1 点 (x, y) におけるフーリエ記述エッジ方向ヒストグラム :

$$d_q = \frac{1}{2\pi} \int_{-\pi}^{\pi} |I_1(x, y)| \delta(\theta - \arg(I_1(x, y))) e^{-iq\theta} d\theta$$

$$= \frac{1}{2\pi} |I_1(x, y)| e^{-iq \arg(I_1(x, y))} \simeq \frac{\overline{I_1(x, y)^q}}{2\pi (|I_1(x, y)|^2 + \varepsilon)^{(q-1)/2}}$$

ここで、 ε は正則化定数である（実験では、 $\varepsilon = 10^{-12}$ ）。

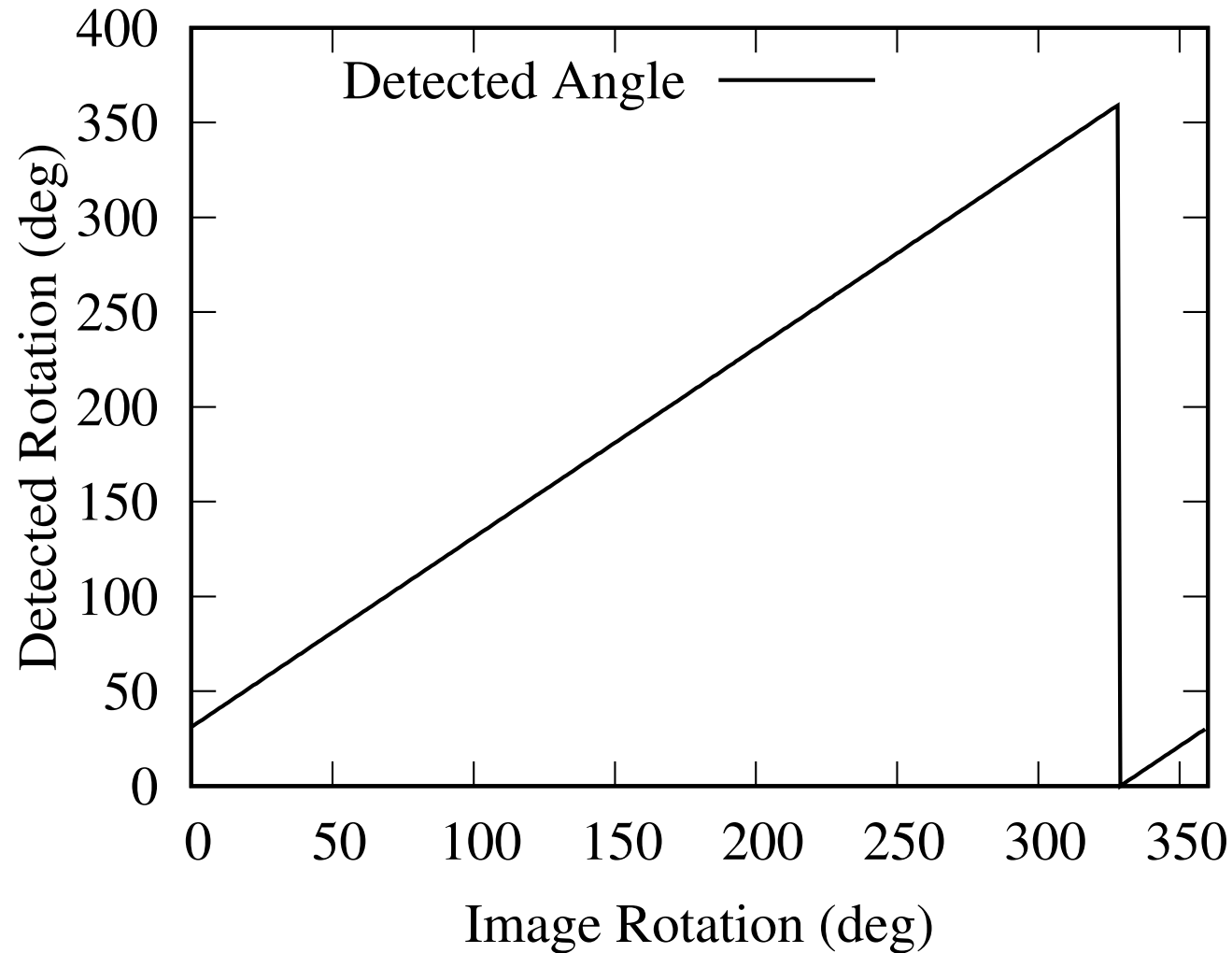
- この1点のフーリエ記述エッジ方向ヒストグラムを、ガウス平滑化し、領域のフーリエ記述エッジ方向ヒストグラムを求める。
- d'_q : エッジの基準方向を θ_0 回転したときの係数

$$d'_q = d_q e^{-iq\theta_0}$$

フーリエ記述の利点

- エッジ方向を量子化することなく、四則演算だけで計算できる。
- ヒストグラム最大点の画像回転共変性
 - 画像を1度回転させれば、検出最大位置が1度ずれる。
⇒ 本当の最大位置ではないが、基準方向のためには十分。
 - SIFTでは、基準方向を求めるために、36方向に量子化。
⇒ 平滑化を36回行う必要がある。

- 画像 : Lenna, 標本点 : (256, 256), 平滑化スケール : 60画素
- 画像を1度ずつ回転, 1/16度でヒストグラム最大点を検出
⇒ 検出された最大点の角度と画像回転角の差 : 31.0625 ± 0.0625



4.3 マッチングアルゴリズムの概要

1. テンプレート画像の抽出したい物体領域の中心に**複数のアフィン変換**を施し，**複数スケール**で領域特徴量を求める。
2. 入力画像から，その標本点において複数スケールで特徴量を求める。
3. 両者のベクトルの内積を計算し，それが最大となる標本点位置を抽出位置，スケールを抽出スケールとする。

注意：

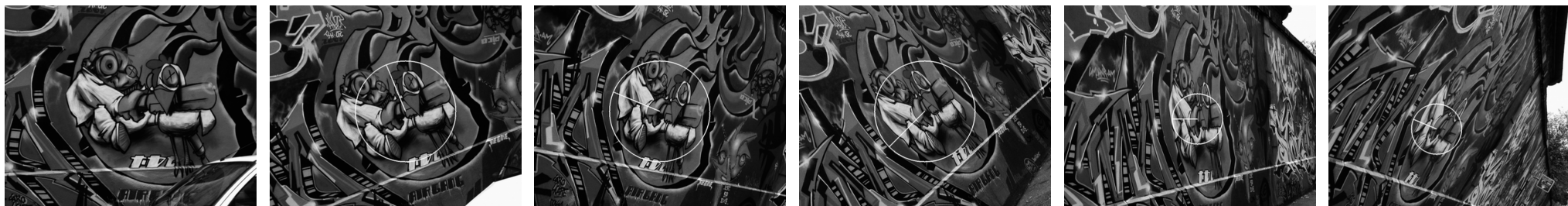
- 入力画像におけるスケールの細かい変化を，テンプレート画像側で吸収するために，テンプレート画像の特徴量を求めるときも複数スケールを用いる。
- 計算機実験では
 - テンプレート画像でのスケール： $S_0, \sqrt{2}S_0$
 - 入力画像でのスケール： $S, 2S, 4S$

5 計算機実験

- 使用した画像

Graffiti : 800×600

Boat : 850×680



(a) テンプレート (b)~(g) 入力画像



(a) テンプレート (b)~(g) 入力画像

比較

- 提案手法 : C program
<https://github.com/heavenstime/fastGMatch>
- ASIFT+RANSAC (OpenCV 3.4.1 + python) パラメータ等に変更なし。
- 計算機 : Intel(R) Core(TM) i5-6400 CPU @ 2.70GHz
- 両者ともテンプレートの特徴量の作成時間は省略した。
- ASIFT+RANSACでも, 正確に領域を抽出できた。

計算時間 (second)

Image \ Method	ASIFT+RAMSAC	Ours
Graffiti	21.881	0.699
Boat	168.366	1.132

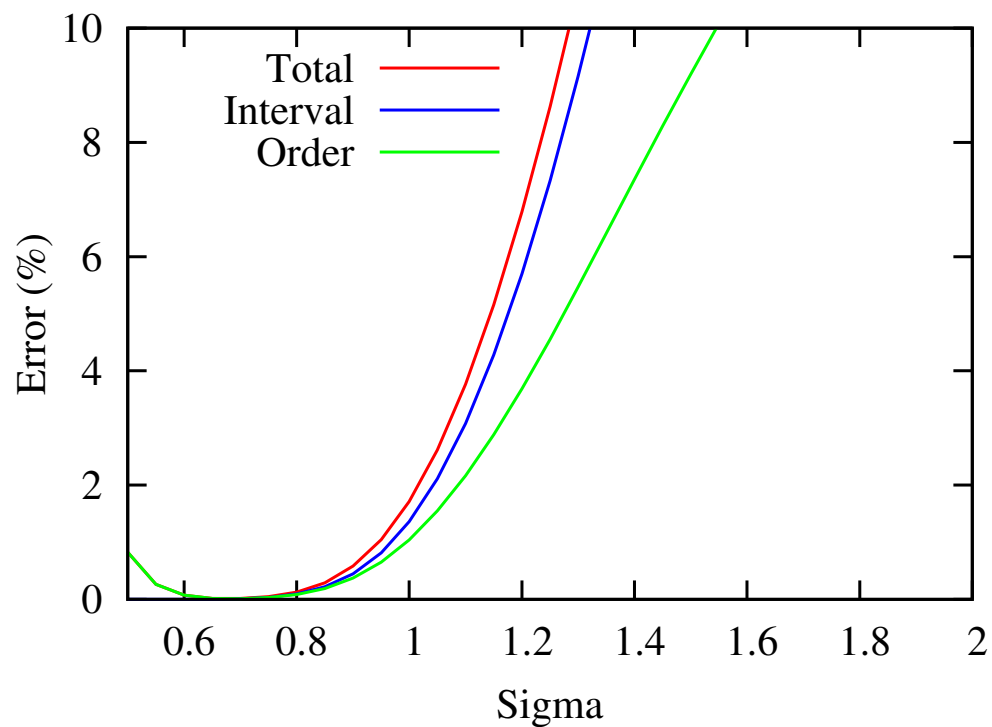
参考 : 平滑化(微分, LoGを含む)のMex関数のソースコード :

<https://github.com/heavenstime/gaussSmooth>

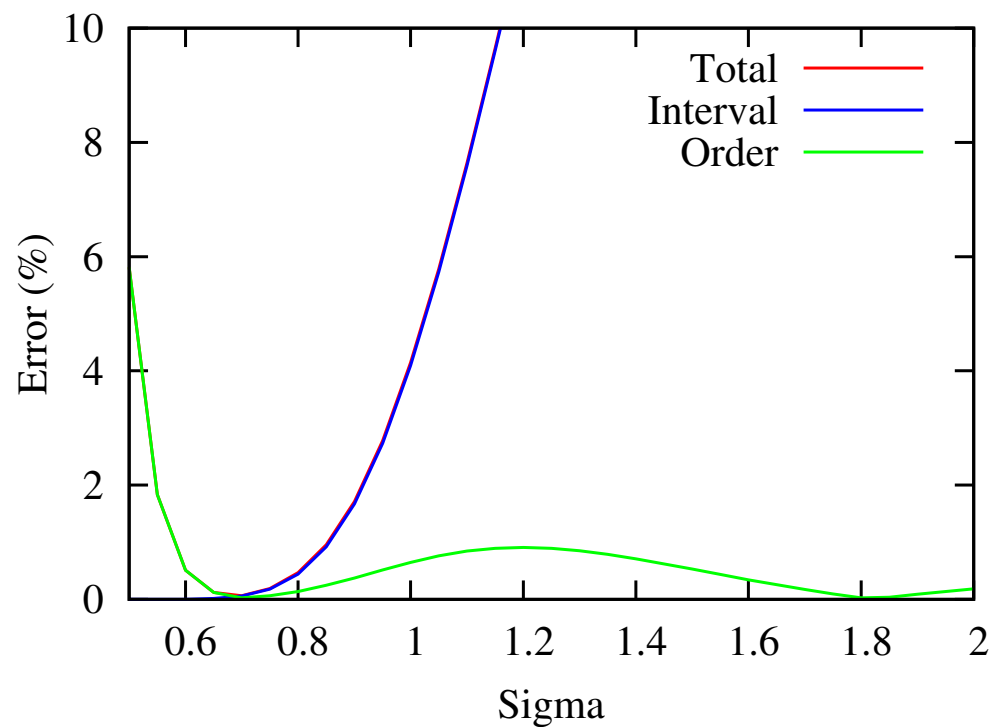
6 まとめ

- スライディング畳み込み積分によるガウス平滑化を用いた領域マッチングの初期探索法を提案した。
- フーリエ記述エッジ方向ヒストグラムにより，GPUに適した単純な計算だけで特徴量が計算できる。
- SIFTの特徴量と異なり，特徴量のエッジ方向抽出時にスケールに応じたガウス平滑化がされている。
- 計算機実験により，十分な精度と，実用的な計算時間で初期探索を行うことができることを示した。
- 今後の課題
 - GPUを利用し，さらに高速化する。
 - 高精度・高速な汎用的特徴点抽出法として展開する。
 - GAT/GPT 相関法の初期探索法として用いる。

$P = 6$ の区間打ち切り, 次数打ち切りによる誤差



$nG[n], K = 256, P = 6$



LoG, $K = 256, P = 6$

単精度浮動小数点演算誤差

- $K = 16$, $P = 1$ の \cos 成分の

$$\frac{\text{区間長 } 10,000 \text{ での最大誤差}}{\text{区間長 } 10,000 \text{ での値の平方根 } 2 \text{ 乗平均}} \times 100$$

