SIFT における特徴点候補検出の複素一次系による高速化

梅本　敏孝† 　黄瀬　浩一††

† 大阪府立工業高等専門学校　〒 572-8572　大阪府寝屋川市幸町 26-12
†† 大阪府立大学大学院工学研究科　〒 599-8531　大阪府堺市中区学園町 1-1
E-mail: †umemoto@ipc.osaka-pct.ac.jp, ††kise@cs.osakafu-u.ac.jp

あらまし　SIFT などの局所記述子の最近傍探索によって物体認識を行う場合には，画像から特徴量を抽出するのに必要な処理コストを最小化することが重要である．その特徴量を抽出する処理は，スケールとキーポイント検出，キーポイントのローカライズ，オリエンテーションの算出と特徴量の記述から構成されている．本論文では，これらの処理の中でスケールとキーポイント検出に必要な処理コストを削減する方法を提案する．具体的には，スケールを検索するための画像は，入力画像データとガウス関数の畳み込みによって作られる．そこで，その処理に必要な乗算回数を青島によって提案された複素 1 次系を用いることで削減する．処理時間の計測と ANN を用いた画像検索実験から，提案した方法は認識率でガウス関数を用いた畳み込みを用いた場合と同程度の認識率を得ることができた．さらに，計算時間は 3 分の 1 となった．

キーワード　SIFT，ガウス関数，複素 1 次系，ウェーブレット

Speeding up the Detection of Scale-Space Extrema in SIFT Based on the Complex First Order System

Toshitaka UMEMOTO† and Koichi KISE††

† Osaka Prefectural College of Technology, Department of Industrial Systems Engineering　26-12 Saiwai-cho,Neyagawa,Osaka 572-8572, Japan
†† Dept. of Computer Science and Intelligent Systems, Graduate School of Engineering, Osaka Prefecture University　1-1 Gakuencho, Naka, Sakai, Osaka 599-8531, Japan
E-mail: †umemoto@ipc.osaka-pct.ac.jp, ††kise@cs.osakafu-u.ac.jp

Abstract　For object recognition based on nearest neighbor search of local descriptors such as SIFT, it is important to minimize the cost of extracting image features. The processes of extracting image features are composed of scale-space extrema detection, keypoint localization, orientation assignment and keypoint descriptor.In this paper, we propose a new method of efficient scale-space extrema detection. The scale space of an image is defined as a function, that is produced from the convolution of a variable-scale Gaussian with an input image. Number in which multiplication is executed is reduced by complex first order system proposed by Aoshima. From experimental results with computation time and recognition of image with ANN, we have confirmed that the proposed method is capable of achieving a recognition rate as same as the original method, and 1/3 of the computation time with the original method.

Key words　SIFT,Variable-scale Gaussian,Complex First Order System,Wavelet

1. は じ め に

近年，パソコンの一般家庭への普及の拡大などにより，デジタルデータを扱う機会が増えている．携帯電話やデジカメで写真を撮ったり，動画像においても，デジタルビデオカメラでビデオ撮影をしてそれを活用する場面は多い．

そうしたニーズにより，画像や動画を用いた研究が盛んになっており，画像検索の分野において，画像から特徴量を抽出し，その特徴量を用いて検索をするという考えがある．

特徴量の抽出手法の 1 つとして，SIFT (Scale-Invariant Feature Transform) [1] があり，その SIFT に対して主成分分析 (PCA) を適用した PCA-SIFT も提案されている [2]．これら

の方法を用いると，照明，視点の変化に対し，安定した特徴量を得ることができるため，画像に関連した様々な研究に用いられている．しかし，特徴量の抽出に時間がかかり，動画のリアルタイム処理など，短時間に多くの画像から特徴量を抽出する必要がある場合に対応できないなどの問題点がある．そこで，SIFT 演算の高速化が望まれている．この SIFT 演算において，大きなウェートを占めているものに，DoG 処理によるスケールとキーポイントの検出がある．この処理には，ガウス関数が用いられおり，入力画像との畳み込んだ平滑化画像の差分から構成されている．つまり，ガウス関数と入力画像との畳み込み処理を行う際の乗算回数を削減すれば処理が高速化できると考えられる．

一方，音声信号に代表される時系列信号を解析する場合には，ガウス関数などの窓関数と信号とを畳み込みした信号を高速フーリエ変換するなどの処理が用いられている．このような処理を高速に行う方法として，我々は，青島らによって提案された複素 1 次系 [3] を一般化したウェーブレット変換の高速計算法 [4] を提案した．このウェーブレット変換法の特徴は，櫛形フィルタと呼ばれる遅延演算と共振器を組合すことによって乗算回数の削減していることである．本論文では，この高速ウェーブレット変換法を応用した近似 DoG 処理アルゴリズムを提案し，ガウス関数との畳み込みを用いた DoG 処理に必要な計算時間の比較を行うとともに，検索の精度が高いとされる ANN（Approximate Nearest Neighbor）[5] と呼ばれる多次元空間における近傍検索法を用いた画像検索の精度を比較することで，アルゴリズムの有効性を確かめる．

2. 複素一次系

2.1 一次系とは

複素一次系は，通常の制御で我々が用いている一次系の減衰振動を複素数に拡張したものである．ここでは，連続系の一次系についてまず説明する．連続系における一次系は次に示すような式で表すことができる．

$$\frac{dx(t)}{dt} = fx(t) + gu(t) \tag{1}$$

ここで，$x(t)$ は出力信号であり，$u(t)$ は入力信号である．また，f，g は任意の実定数である．式（1）は次のような一般解をもつ．

$$x(t) = x_0 \exp(ft) + \int_0^t \exp\{f(t-\tau)\} g u(\tau)\, d\tau \tag{2}$$

ここで，$x_0\,(=x(0))$ は $x(t)$ の初期値である．

2.2 離散系における一次系

つぎに前項で説明した連続系における一次系の離散化を考える．$t = nT$ とおくと（n ＝整数，T ＝サンプリング周期），入力 $u(t)$ は $t = nT \sim (n+1)T$ の間で一定値となり，式（2）は $t = nT \to (n+1)T$ で積分すると次のようになる．

$$x(n+1) = \exp(fT)\, x(n) + \frac{g}{f}\{\exp(fT) - 1\} u(n) \tag{3}$$

ここで，$x(n)$ は $x(nT)$ を意味する．

2.3 複素一次系とそれを用いた短時間スペクトル分析法

複素一次系は前項で説明した一次系の 2 つの実定数を複素数に拡張したものとして定義する．つまり，f を $\sigma + j\omega$ に，g を $C_r + jC_i$ にしたものであり，離散系における複素一次系の入出力の関係はつぎのようにおくことができる．

$$\begin{aligned} x(n+1) &= x(n)\exp\{(\sigma+j\omega)T\} \\ &\quad + \frac{C_r + jC_i}{\sigma + j\omega} u(n)\,[\exp\{(\sigma+j\omega)T\} - 1] \end{aligned} \tag{4}$$

この式から，複素一次系のインパルス応答は減衰振動系のインパルス応答で近似することができる．パラメータ ω は，減衰振動系の振動数に，σ は，減衰項に一致する．ここで，パラメータ $C_r = 1$，$C_i = 0$ および $\sigma = 0$ の場合，複素一次系のインパルス応答の実部は cos 波となり，虚部は sin 波となる特徴を持つ．

複素一次系の応答 $x(n)$ は，入力信号 $u(n)$ と式（4）で計算することができる．次に，初期値 $x(-0)$ が 0 であり，時刻 0 つまり $n = 0$ で大きさ 1 の入力が加わり，それ以後の入力が 0，つまり $u(n) = 0$ である場合を考えると，$n = d - 1$ における値は次のようになる．

$$\begin{aligned} x(d-1) &= \frac{C_r + jC_i}{\sigma + j\omega}[\exp\{(\sigma+j\omega)T\} - 1] \\ &\quad \cdot \exp\{(d-1)(\sigma+j\omega)T\} \end{aligned} \tag{5}$$

ここで，$n = d - 1$ において大きさ A の入力が加わったとすると $n = d$ における値は次のようになる．

$$\begin{aligned} x(d) &= \frac{C_r + jC_i}{\sigma + j\omega}[\exp\{(\sigma+j\omega)T\} - 1]\exp\{d(\sigma+j\omega)T\} \\ &\quad + \frac{C_r + jC_i}{\sigma + j\omega} A[\exp\{(\sigma+j\omega)T\} - 1] \end{aligned} \tag{6}$$

ここで，A を次のようにおく．

$$A = -\exp\{d(\sigma+j\omega)T\} \tag{7}$$

この場合，$x(d)$ は 0 となり，入力 $u(n)$ がその後 0 であるならば，$x(n)$ は $n \geqq d$ で恒等的に 0 になる．つまり，図 1 のシステムを考えると，$n = 0$ において $u(n)$ として入力を加えれば $n = d$ 以後，出力 $x(n)$ は 0 になる．$\sigma = 0$ の場合，複素一次系のインパルス応答は実部が cos 波を，虚部が sin 波を乗算し，積分したものになり，短時間フーリエ変換を行ったものと考えることができる．このことから，図 1 で示されるシステムは，短時間スペクトルを求めることができるシステムであり，実際には，次の式で計算することができる．

$$\begin{aligned} x(n+1) &= \exp(j\omega T)\, x(n) + \exp(j\omega T) \\ &\quad \cdot \{u(n) - \exp(j\omega dT)\, u(n-d)\} \end{aligned} \tag{8}$$

3. SIFT および PCA-SIFT とは

3.1 SIFT

画像は膨大なデータを持っているため，画像から特徴的な部分を抽出し，処理を行うことが多い．前節で述べたように，SIFT は，照明や始点の変化に対して安定した特徴量を画像から抽出する手法である．

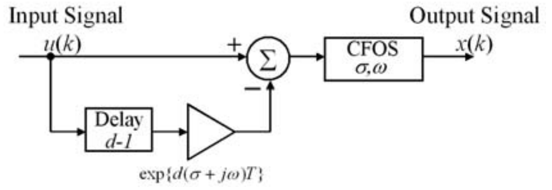

図 1　複素一次系を用いた短時間スペクトル分析法

3.2　PCA-SIFT

Ke らは，SIFT に対して主成分分析 (PCA) を適用する PCA-SIFT を提案している [2]．これにより，SIFT に比べ，安定性と識別性の向上や，特徴ベクトルの次元数の削減が可能となる．PCA-SIFT によって得られる特徴ベクトルは，低次元の実数値ベクトルである．本稿では，Ke らのインプリメントによって得られる，36 次元の整数値ベクトルを用いる．

3.3　スケールスペースを用いた特徴点候補の検出

本論文では，SIFT およびそれを応用した PCA-SIFT の高速化を行うために，スケールスペースを用いた特徴点候補を検出する処理に必要な乗算回数の削減を行う．この節では，スケールスペースを用いた特徴点候補の検出について説明する．

特徴点探索としては，まず，同じ物体において，照明や視点の変化など，異なった条件下でも同様に定まる点や尺度を見つけることが重要である．画像のスケールを変化させた場合においても，点が一定に定まれば，特徴点の候補になる．

画像から特徴点を抽出するにあたり，画像のノイズ除去，すなわち平滑化を行う．分散 σ の値の変化させて得た複数の平滑化画像を順に並べたものはスケールスペースと呼ばれる [6]．画像のスケールを変化させた場合においても定まる点は特徴点の候補となるが，SIFT では，分散 σ の値を変えて得られる複数の画像からでも検出できる点を特徴点候補にする．

具体的には以下の手順による．

（1）　入力画像 $I(x,y)$ に対して，ガウス関数 $G(x,y,\sigma)$ を畳み込み，平滑化画像 $L(x,y,\sigma)$ を得る．

$$L(x,y,\sigma) = G(x,y,\sigma) * I(x,y) \tag{9}$$

ここで "*" は畳み込みの演算を表す．また，ガウス関数 $G(x,y,\sigma)$ は以下のように定義される．

$$G(x,y,\sigma) = \frac{1}{2\pi\sigma^2}e^{-(x^2+y^2)/2\sigma^2} \tag{10}$$

（2）　分散 σ の値を変化させ，1. と同様の処理を行う．これによって，複数の 平滑化画像 $L(x,y,\sigma_i)$ を得る．

（3）　隣り合う平滑化画像 L の差分をとり，DOG (Difference-of-Gaussian) 画像 $D(x,y,\sigma)$ を得る (図 2)．

（4）　得られた DOG 画像から極値点を検出する．

以上によって得られる極値点が，画像における輝度変化の大きな部分であり，特徴点の候補となる．

特徴点候補を効率的に求めるために，ガウス関数の差をとった DOG 関数 (Difference-of-Gaussian Function) を用いている．

$$\begin{aligned} D(x,y,\sigma) &= \{G(x,y,k\sigma) - G(x,y,\sigma)\} * I(x,y) \\ &= LG(x,y,\sigma) - L(x,y,\sigma) \end{aligned} \tag{11}$$

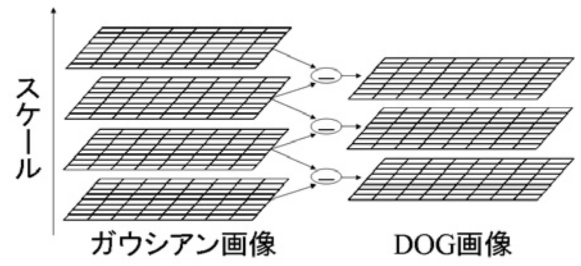

図 2　Difference of Gaussian 画像の生成

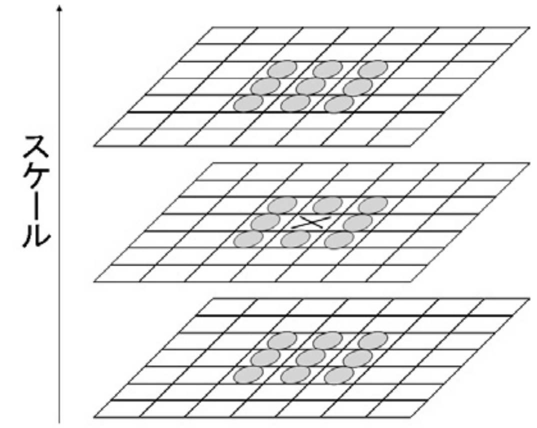

図 3　極値点決定の為の比較の 26 画素

ここで，$D(x,y,\sigma)$ は，DOG 関数を畳み込んだ DOG 画像，k は倍数因子である．DOG 関数を用いることにより，スケールの正規化されたラプラシアン・ガウシアン $\sigma^2\nabla^2 G$ の近似値を求めることができる．

$$G(x,y,k\sigma) - G(x,y,\sigma) \approx (k-1)\sigma^2\nabla^2 G. \tag{12}$$

このようなラプラシアン・ガウシアン $\sigma^2\nabla^2 G$ によって求まる極大値，極小値は，gradient や Hessian，ハリスのコーナー検出などといった他の画像の関数の中で，最も安定することが Mikolajczyk によって報告されている．よって，DOG 関数を畳み込んだ DOG 画像 $D(x,y,\sigma)$ から極値点を求めることは，特徴点の候補として大変有効である．

DOG 画像から極値点を求めるには，図 3 に示すように，現在のスケールの注目画素の 8 近傍，その上下のスケールのそれぞれ近傍 9 点，計 26 点に対して値の比較を行う．注目画素がその 26 画素の中で最大であれば極大とし，最小であれば，極小とする．

4.　スケールスペースを用いた特徴点候補検出の高速化

3.3 節の手順で説明したように入力画像 $I(x,y)$ に対して，ガウス関数 $G(x,y,\sigma)$ との畳み込みを行い，平滑化画像 $L(x,y,\sigma)$ を得る．画像データを $I(x,y)$ を $M\times N$ ピクセル，ガウス関数 $G(x,y,\sigma)$ を $L\times L$ ピクセルとすると，この処理を 2 次元のガウス関数を用いて単純に実行すると $M\times N\times L\times L$ 回の乗算が必要となる．つまり，このフィルタの乗算の回数は，そのフィルタサイズの 2 乗のオーダーに依存して増加する．そこで，その高速化の方法として変数分離による処理の高速化と呼ばれ

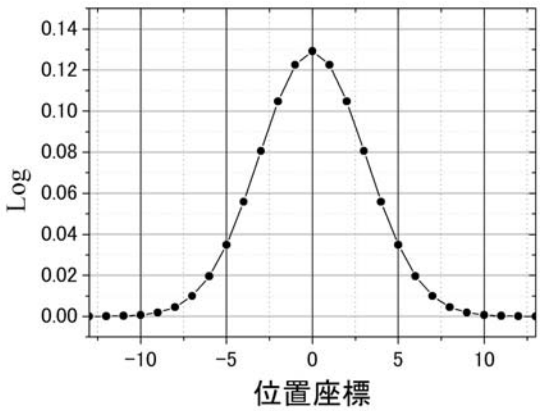

図 4　Gauss 関数 ($\sigma = 3.09$)

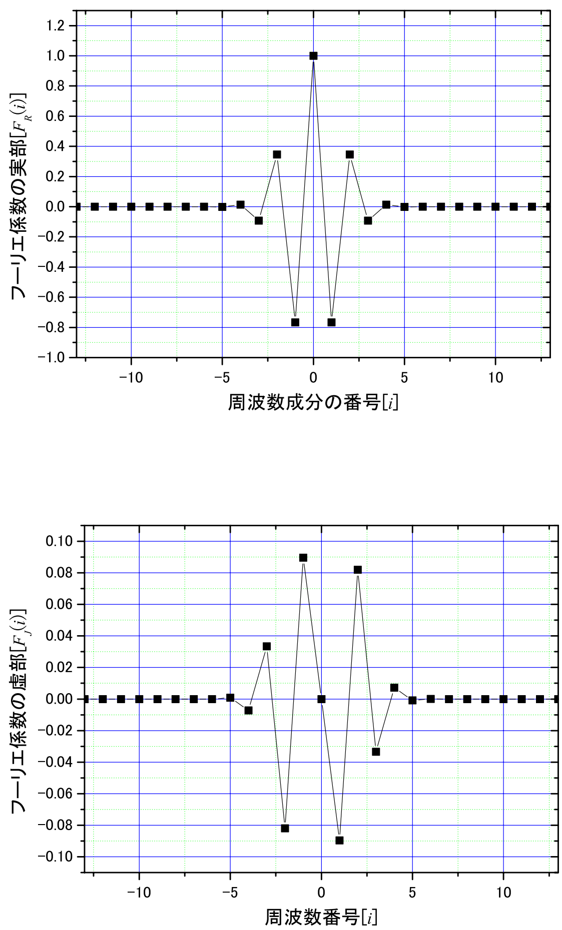

図 5　Gauss 関数 ($\sigma = 3.09$) の離散フーリエ変換

る手法が一般的に用いられる．この方法では，式 (10) を指数関数の特徴を利用して次のように変形する．

$$G(x,y,\sigma) = \frac{1}{2\pi\sigma^2}e^{-x^2/2\sigma^2}\frac{1}{2\pi\sigma^2}e^{-y^2/2\sigma^2} \tag{13}$$

ここで，$G(x,\sigma)$，$G(y,\sigma)$ を次のように置く．

$$G(x,\sigma) = \frac{1}{2\pi\sigma^2}e^{-x^2/2\sigma^2} \tag{14}$$

$$G(y,\sigma) = \frac{1}{2\pi\sigma^2}e^{-y^2/2\sigma^2} \tag{15}$$

これら 2 つの式を用いて，画像をまず列もしくは行ごとに 1 次元のガウス関数を用いて処理を行う．その処理で得られた画像データを逆に行もしくは列ごとに 1 次元のガウス関数を用いて処理する．このように処理することによって乗算の回数は，$2M \times N \times L$ 回となり，処理の高速化が図れる．しかし，この方法でもフィルタの乗算の回数は，フィルタサイズのオーダーに依存して増加するという問題点は残る．

そこで，我々は画像処理の高速化手法として，Viola と Jones らが特徴量を高速に抽出するために用いているインテグラルイメージ [9], [10] と青島によって提案された複素一次系を用いた短時間スペクトル分析法 [3] を応用した方法を提案する．

ここで，例としてフィルタサイズ $L = 27$ ピクセル，$\sigma = 3.09$ とした時の 1 次元のガウス関数を図 4 に示す．このガウス関数を 27 点の離散フーリエ変換するとフーリエ係数は，図 5 に示すような値となる．この図からわかるように図 4 で示されるガウス関数を離散フーリエ変換した場合，フーリエ係数として大きな値をとるのは，3 成分程度である．そこで，本論文では，ガウス関数 $G(x,\sigma)$ を次に示す式で近似することにする．

$$G(x,\sigma) = F(0) + F(1)e^{j2\pi k/L} + F(-1)e^{-j2\pi k/L} \tag{16}$$

ここで，$F(0)$ は $k = 0$ 番目のフーリエ係数，$F(1)$ は $k = 1$ 番目のフーリエ係数，$F(-1)$ は $k = -1$ 番目のフーリエ係数である．この式を用いるとガウス関数 $G(x,\sigma)$ 画像の行方向 $I(x)$(M ピクセル) の畳み込みは次のように近似できる．

$$\begin{aligned} G(x,\sigma) * I(x) \;=\; & \sum_{x=0}^{M}\Bigg[\sum_{k=0}^{L-1}\{F(0) + (F(1)e^{j2\pi k/L} \\ & +F(-1)e^{-j2\pi k/L})\}I(x-k)\Bigg] \end{aligned} \tag{17}$$

ここで，この式の $[\cdot]$ の中の演算を考えと，その 1 項目は次のようにおける．

$$O_1(x,\sigma) = F(0)\cdot\sum_{k=0}^{L-1} I(x-k) \tag{18}$$

この式は，一般的にインテグラルイメージと呼ばれる方法である．インテグラルイメージとは，Viola と Jones らが画像内のある矩形領域内に含まれる画素値の差から特徴量を高速に抽出する方法 [9], [10] として用いたものであり，その利点は乗算の回数を減らすことができることである．次に 2 項目は次のようになる．

$$\begin{aligned} O_2(x,\sigma) \;=\; & \cdot\sum_{k=0}^{L-1}(F(1)e^{j2\pi k/L} + F(-1)e^{-j2\pi k/L})I(x-k) \\ =\; & F(1)\sum_{k=0}^{L-1}e^{j2\pi k/L}I(x-k) \end{aligned}$$

$$+F(-1)\sum_{k=0}^{L-1} e^{-j2\pi k/L} I(x-k) \quad (19)$$

この式を直接計算すると乗算の数が減少しない．そこで，本論文では，青島らによって提案された複素１次系を用いて乗算の回数を減少させる方法を用いることにする．

式 (19) の右辺 2 項目は，次のようになる．

$$O_{2,2}(x,\sigma) = \sum_{k=0}^{L-1} I(x-k) e^{-j2\pi k/L} \quad (20)$$

この式は，入力画像データ $I(x-k)(k=0,1,\cdots,L-1)$ に対する周波数 $f=\frac{1}{L}$ の離散フーリエ変換（Discrete Fourier Transform:DFT）を行っている式と考えることができる．同様に入力画像データ $I(x+1-k)(k=0,1,\cdots,L-1)$ に対する周波数 $f=\frac{1}{L}$ の離散フーリエ変換は次のようになる．

$$O_{2,2}(x+1,\sigma) = \sum_{k=0}^{L-1} I(x+1-k) e^{-j2\pi k/L} \quad (21)$$

式 (21) から式 (20) に $e^{j2\pi 1/L}$ をかけたものを引くと，次のようになる．

$$O_{2,2}(x+1,\sigma) - e^{j2\pi 1/L} O_{2,2}(x,\sigma) = e^{j2\pi 1/L} \cdot \{I(x) - e^{j2\pi L/L} I(x-L)\} \quad (22)$$

ここで，$e^{j2\pi 1/L}O_{2,2}(x,\sigma)$ を右辺に移項し，$e^{j2\pi L/L}=1$ を用いると，式 (22) は次のようになる．

$$O_{2,2}(x+1,\sigma) = e^{j2\pi 1/L} O_{2,2}(x,\sigma) + e^{j2\pi 1/L}\{I(x) - I(x-L)\} \quad (23)$$

この式は，式 (19) の右辺 2 項目を逐次的に計算する式である．また，この式は，式（8）において遅延時間 dT を遅延画素数 L，角周波数 ω を $2\pi/L$ とした式である．つまり，複素一次系を用いた短時間スペクトル分析法を用いれば，式 (19) を逐次的に計算できる．さらに，この方法は，縦方向も同様の考え方で計算することができる．

提案している方法を用いて，両方向に計算していくことで $G(x,y,\sigma)*I(x,y)$ を求める場合に必要な乗算回数を考える．ここで，4 章と同様に，画像データを $I(x,y)$ を $M \times N$ ピクセル，ガウス関数 $G(x,y,\sigma)$ を $L \times L$ ピクセルとする．まず，式 (17) は，式 (18) と式 (19) に分けて考えることができる．式 (18) は，$O_1(x,\sigma)$ を求めるのに 1 回の乗算が必要である．このことから，画像データ $I(x,y)$ 全体で，$2 \times M \times N$ 回の乗算が必要となる．

次に，式 (19) の右辺 1 項目と 2 項目が複素共役の関係にあることを利用する．このことから，式 (19) の右辺 1 項目のみを逐次的に計算し，その複素共役から 2 項目を求めることにする．式 (23) より $O_{2,2}(x,\sigma)$ を求めるために 2 回の複素乗算が必要となるので，4 回の乗算が必要となる．さらに，$O_{2,2}(x,\sigma)$ を用いて式 (19) を求めるために，複素共役の性質を利用すると 1 回の乗算で計算ができる．以上のことから 5 回の乗算で $O_2(x,\sigma)$ が求まる．このことから画像データ $I(x,y)$ 全体で，$10 \times M \times N$ 回の乗算が必要となる．よって，$G(x,\sigma)*I(x)$ を求める場合に必要な乗算回数は，$12 \times M \times N$ 回となり，L が大きい場合でも乗算の計算量が増えないことがわかる．一方，ガウス関数を変数分離による処理の高速化と呼ばれる手法で求める場合には，$2 \times M \times N \times L$ 回の乗算が必要である．つまり，σ が大きくなり，L が大きくなる場合には，乗算の回数が増加することがわかる．このことから，本論文で提案する方法は，乗算の計算量が減少することがわかる．

表 1　計算時間の比較 (msec)

σ	提案法	OpenCV	従来法
3.09	22.9	29.5	82.2
2.4525	22.2	25.2	73.1
1.9466	22.1	22.1	60.5
1.545	21.6	21.6	49.9
1.249	20.7	20.7	38.3

5. 実　　験

5.1 処理時間の比較実験

本節では，画像 1 枚に対し，式 (9) を処理するためにどの程度の処理時間がかかるか比較する．SIFT のインプリメントとして，C#で実装した libsift [11]，C++で実装した SIFT++ [12] などが公開されている．本節では，SIFT++のインプリメントを参考に，式 (9) を処理するためのプログラムを作成した．作成したプログラムは，2 章で提案した複素 1 次系を用いた画像の平滑化処理プログラム，変数分離を用いたガウス関数による画像の平滑化プログラムおよび OpenCV で用意されている cvSmooth() 関数を用いたプログラムの 3 つである．

本実験では，前節で述べた動作にどの程度時間がかかるのかを比較するために，ガウス関数 $G(x,\sigma)$ における σ は，SIFT 処理で用いられる 3.09，2.4525，1.9466，1.545，1.249 の 5 つの値とし，平均サイズ，縦 456 ピクセル，横 640 ピクセル，平均画素数 29.2 万ピクセルの画像 12 枚を用いた．また，実験に用いた計算機は，CPU Intel CoreSolo 1.2GHz であった．比較結果を表 1 に示す．

これらの表からわかるように，提案した複素 1 次系を用いた方法は，4 章で説明した変数分離を用いたガウス関数による画像の平滑化法に比べて高速に計算できるが，OpenCV で用意されている cvSmooth() 関数と比べれば高速に計算できていないことがわかる．これは，OpenCV で用意されている関数は一般的に最適化がなされており，我々ユーザーが作成する関数より高速であるからであろう．

5.2 認識率の比較実験

実験では，近似最近傍探索を用いて認識率の比較を行った．実験には以下に述べる画像データベース，検索質問画像を用いた．局所記述子としては，PCA-SIFT のサイト [13] で提供されるものを用いた．

実験に用いた画像について説明する．用いた画像の数は，ポスターを中心とした 600 枚の画像データである．画像データは，長辺が 640 ピクセル 以下になるように縮小し，グレース

(a) 撮影角度 90°

(b) 撮影角度 75°

(c) 撮影角度 60°

(d) 撮影範囲一部

図 6　検索質問の例

表 2　認識率の比較

ε	10	5	3
提案法	92.4%	95.9%	96.5%
インテグラルイメージ法	63.3%	74.9%	78.6%
従来法	96.3%	97.0%	97.0%

ケールにした．一画像あたり平均特徴ベクトルは，従来法では 760 個，提案法では 752 個の抽出された．また，近似最近傍探索の手法としては，ANN(Approximate Nearest Neighbor) [5] を用いる．この方法は、二分木を用いて近似最近傍探索を高速に行う手法である．

また，検索質問としては，データベースに対応する画像のあるものから作成した．この際，データベースに含まれる画像の中から，200 枚を無作為に選択した．次に，これらを A4 の用紙に印刷し，カメラを用いて撮影した．紙面全体が写る配置で，紙面に対するカメラの光軸の角度 θ を 90°, 75°, 60° に変化させた．また，角度を 90° として紙面の一部分を撮影した．その結果，1 枚の紙面に対して，合計 4 通りの画像を得た．この画像の例を図 5.2 に示す．さらに，撮影した画像を 512 × 341 ピクセル に縮小し，PCA-SIFT により特徴ベクトルを求めた．その結果，画像 1 枚あたり平均，従来法では 336 個，提案法では 342 個の特徴ベクトルが得られた．

許容誤差 ε を変化させた場合の比較を表 2 に示す．なお，比較のため，式 (17) を 1 項目で打ち切った場合（インテグラルイメージ）の精度も示す．

この表および特徴ベクトルの抽出数などからわかるように，インテグラルイメージでは，十分な精度は得られないが，それに式 (19) の成分を加えた提案法は，大幅な近似を行いない場合には，従来法と同等の精度が得られることがわかる．

6. 結　　論

本論文では，PCA-SIFT や SIFT の処理を高速化するためにスケールスペースを用いた特徴点候補検出の高速化する方法を提案するために青島によって提案された複素 1 次系に着目した．

一般的なガウス関数の変数分離を応用した方法での乗算回数が，画像データが（$M \times N$ ピクセル），ガウス関数が $G(x, y, \sigma)$（$L \times L$ ピクセル）とであるとすると，$2M \times N \times L$ 回必要であるのに対して，提案方法では，$1M \times N$ 回である．さらに，処理時間の比較実験においても計算時間が 3 分の 1 程度に短縮することができた．しかし，OpenCV で用意されている cvCVSmooth() 関数を用いた場合と比べると高速化はできなかった．これは，OpenCV で用意されている関数は一般的に最適化がなされており，我々ユーザーが作成する関数より高速であると考えられる．しかし，岩村らによって提案されている GPU を用いた PCA-SIFT の高速化 [14] などの場合には，高速化は可能な方法であると考えられる．

また，提案した方法の特徴点抽出数は，従来法の特徴点抽出数と同程度であり，近似最近傍探索の従来法である ANN を用いて画像検索における認識率の比較実験の結果でも，ANN の許容誤差 ε を 5 程度にすれば，従来法と同程度の認識率が得られることもわかった．

さらに，データベースの数を増やし，画像検索における認識率を確かめる必要はあるが，本論文で提案した複素 1 次系を応用したスケールスペースを用いた特徴点候補検出の高速化法は，PCA-SIFT や SIFT の処理を高速化するのに有効であると考えられる．

文　　献

[1] D.G.Lowe,"Distinctive image features from scale-invariant keypoints", International Journal of Computer Vision, 60,2, pp.91-110 ,2004.

[2] Y. Ke, R. Sukthankar,"Pca-sift:A more distinctive representation for local image descriptors", CVPR2004,Vol.2,pp.506-513,2004.

[3] 青島，" 複素 1 次系とその応用", 計測自動制御学会論文集,Vol.26,No.7,pp.811-817,1990-7.

[4] 梅本, 藤沢, 葭谷," ウェーブレット変換の高速計算法とその応用", 信学論 (A),J79-A,pp.2063-2066,1996-12.

[5] S.Arya, D. M.Mount, R.Silverman, A.Y.Wu,"An optimal algorithm for approximate nearest neighbor searching",Journal of the ACM,Vol.45, No.6,pp.891-923,1998.

[6] 徐剛,"3 次元ビジョン", 共立出版,pp.29-30,1998.

[7] 荒井, 武本, 加藤, 和田," 階層的固有空間による高次元最近傍探索の高速化", 画像の認識･理解シンポジウム (MIRU2006),pp.291-297,2006.

[8] J. Sivic,A. Zisserman"Video Google: a text retrieval approach to object matching in videos",Computer Vision, 2003. Proceedings. Ninth IEEE International Conference on, pp.1470-1477,2003.

[9] Paul Viola, Michael Jones"Rapid Object Detection using a Boosted Cascade of Simple Features",IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Vol.1 pp.511-518,2001.

[10] Paul Viola, Michael Jones"Robust Real-time Object Detection",International Journal on Computer Vision and Image Understanding,Vol.57,Issue 2,pp.137-154, 2004.

[11] http://user.cs.tu-berlin.de/˜nowozin/libsift/

[12] http://vision.ucla.edu/˜vedaldi/code/siftpp/siftpp.html

[13] http://www.cs.cmu.edu/˜yke/pcasift/

[14] M. Iwamura, T.Hondou, K.Noguchi,K.Kise, "An Attempt of CUDA Implementation of PCA-SIFT" , IEICE Technical Report, 107, 281, PRMU2007-117, pp.149-154,2007.