動きぶれや焦点ぼけがある画像からでも抽出できる拡張現実マーカの実現

あらまし 観測画像から物体の位置および姿勢を推定する技術は，拡張現実感研究の基礎を成している．中でも画像マーカは，これを実現する有効な手段の一つである．従来の画像マーカでは，観測画像に動きぶれや焦点ぼけが生じたときにうまく抽出できないことがあった．そこで本研究では，動きぶれや焦点ぼけがある画像からでも抽出できる画像マーカを提案する．従来の画像マーカは，高周波成分であるエッジでパターンを表現するものであったため，動きぶれや焦点ぼけによる高周波抑制によって抽出が困難となっていた．我々は，単一の低周波成分のみでマーカを構成することによって，この問題を解決する．このマーカをモノスペクトルマーカと呼ぶ．モノスペクトルマーカは GPU を用いることで高速に抽出することができる．実験によって，動きぶれや焦点ぼけがある画像からでも，モノスペクトルマーカが精度よく高速に抽出できることを確かめた．

キーワード 拡張現実感，画像マーカ，周波数フィルタリング，GPU，ARToolKit

1. はじめに

カメラで観測された画像から，観測物体やカメラ自身の位置・姿勢を推定する技術は，移動カメラの自己位置推定 [1]，手術シミュレータの実現 [2]，携帯端末アプリケーション開発などのために，活発な議論が行われている．この中で既知パターンを持つ画像マーカは，位置・姿勢推定の実現のために欠かせない技術の一つとなっている．

本研究では，移動するカメラや移動する観測対象の位置・姿勢を推定することを想定し，観測画像に動きぶれや焦点ぼけがある場合にも抽出できる拡張現実マーカの実現を目指す．この技術は，ロボットの自己位置推定やマーカを貼った移動体の検出に効果を発揮する．

従来の画像マーカである AR マーカ [3] や QR コードは，高周波成分であるエッジでパターンを表現する．このため，動きぶれや焦点ぼけがあるときには，観測画像の高周波成分が抑制され，画像マーカをうまく検出することができなかった．

周波数分布に特徴を持たせたマーカ構成の発想が出てこなかったことが，これまでのマーカが動きぶれや焦点ぼけの問題を解決できなかった理由である．本研究では，周波数分布に特徴を持たせたマーカの構成手法を提案することで，高周波抑制の問題を解決する．さらに，マーカの周波数分布の特徴を利用した高速な抽出手法も合わせて提案する．

以降，2. では関連研究を挙げ，本研究との関連について説明する．3. では，動きぶれと焦点ぼけが観測画像に与える影響について述べる．4. では，提案するマーカの設計と抽出の手法について説明する．5. では実験結果を示し，6. で本研究をまとめる．

2. 関連研究

拡張現実感の実現のためには，仮想空間と現実空間の座標系を合わせる必要がある．この作業は幾何学的校正 (geometric calibration) と呼ばれる．以降，幾何学的校正を単に位置合わせ (calibration) と呼ぶことにする．

仮想空間と現実空間の位置合わせのためのマーカとして，ARToolKit のマーカが一般に用いられる [3]．ARToolKit のマーカは，黒の矩形枠の内部に任意のパターンを描き加えて構成される．観測画像中で矩形枠の四隅の頂点を検出し，カメラに対するマーカの位置・姿勢を検出する．内部のパターンは，位置・姿勢を検出した後に既知パターンとの照合を行い，どのマーカが観測されているのかを識別するために使われる．同様に白黒のパターンを用いるものに，ARTag [4] や QR コード，random dot markers [5] などがある．

これらの白黒で構成されるマーカの問題点は，観測画像に生じる動きぶれや焦点ぼけに対処できないことである．これはマーカの位置・姿勢の検出のために，高周波成分であるエッジを利用することに依るものである．観測画像に動きぶれや焦点ぼけが生じると，画像上の高周波成分は抑制され，エッジの位置を正しく検出することができない．

マーカの画像上での見えの違いに着目したマーカには，nested marker [6] や Bokode [7] などがある．Nested marker では，マーカの構造にフラクタル性を導入することで，遠くから見ても近くから見ても同じマーカに見えるようにしている．Bokode は，発光部とレンズを組み合わせて作成されるマーカであり，無限遠点に焦点を設定したカメラで検出ができるようにしたマーカである．これらのマーカにおいても，焦点ぼけや動きぶれによる観測画像の高周波抑制については考慮されていない．

マーカを用いずに物体の位置・姿勢を抽出するために

は，SIFT [8] や SURF [9] などの自然特徴点を用いた位置合わせも行われているが，利用はあらかじめ自然特徴点のパターンを登録しておくことができるものに限られるため，マーカベースの位置合わせとは利用先が異なる．一方，SIFT や SURF は特徴点の見えの大きさに不変な特徴量であり，周波数変化の要因を一部考慮しているという点で，本研究との類似点がある．しかしながら，焦点ぼけや動きぶれによって高周波成分が消失するような状況はやはり想定していない．

ARToolKit のマーカに対して，時間方向の抽出結果の補間によって，その推定精度を上げる研究 [10] や，複数マーカで位置合わせを行うことで精度向上を図る研究 [11] も行われてきた．これらは少数のフレームや少数のマーカだけで正しい抽出結果が得られていないという仮定の下での解決方法であるので，影響が全体に及ぶ動きぶれや焦点ぼけが生じる場合には問題の解決は難しい．

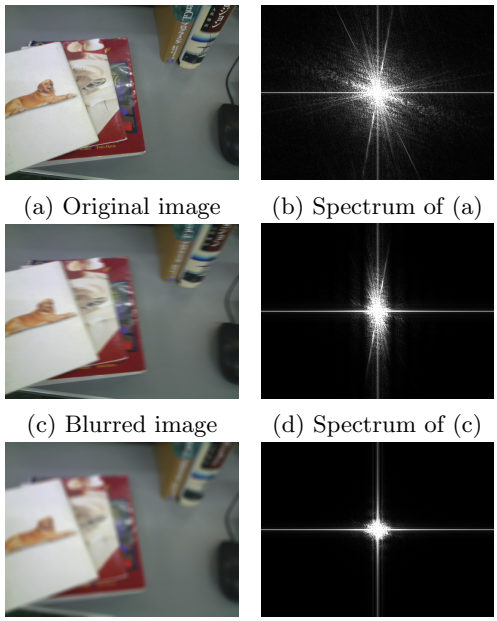
観測画像自体の動きぶれや焦点ぼけを除去する研究や画像の高解像度化を実現する研究も数多くなされている [12]〜[14]．これらの研究では，動きぶれや焦点ぼけがある画像を劣化画像と呼び，PSF(Point Spread Function) によってモデル化することで原画像を復元することを目指している．しかしながら，PSF の推定によって除去できる焦点ぼけや動きぶれには限界がある．我々が想定している移動するカメラが観測するような焦点ぼけや動きぶれに完全には対応できない．一方で，我々が提案する手法は，このような焦点ぼけや動きぶれの除去の利用を妨げるものではない．

逆に ARToolKit のマーカが検出できているという仮定の下で，動きぶれや焦点ぼけの度合いを調べ，仮想物体重畳の際に利用する研究も行われている [15]．この研究では，動きぶれや焦点ぼけを仮想物体の表示の際に利用することで，現実物体と仮想物体の見えの整合性を取ることを目指した．ARToolKit が検出できないような大きな動きぶれや焦点ぼけがある観測画像からでも検出できるようなマーカがあれば，この研究結果を利用することで，従来よりも広い範囲で現実物体と仮想物体の見えの整合性を取ることができるようになる．

3. 動きぶれと焦点ぼけによる高周波成分抑制

動きぶれはカメラの露光中にカメラと撮影対象物体の相対位置が変わることによって生じる．カメラ自体が動く場合や，撮影対象物体が動く場合がこれに当たる．露光時間を短くすることでこの影響を抑えることができるが，十分に明るい環境でない場合には，露光時間を短くすると全体に暗い画像が観測されるという別の問題が生じる．

図 1(a) に示す観測画像があるとき，図 1(b) に示すような周波数スペクトルが得られる．これに対して，横方向の動きぶれが起こると，図 1(c) に示すような画像が得られる．垂直方向のエッジで高周波成分が失われ，図 1(d) に示す周波数スペクトルでは，横軸方向に高周波な成分が失われていることがわかる．



(a) Original image (b) Spectrum of (a)

(c) Blurred image (d) Spectrum of (c)

(e) Defocused image (f) Spectrum of (e)

図 1 動きぶれや焦点ぼけによる周波数スペクトルの変化
Fig. 1 Change of frequency spectrum with Blurring and Defocusing.

一方で，焦点ぼけはカメラの焦点位置とカメラから撮影対象物体までの距離が異なるときに起こる．焦点ぼけの大きさは，焦点距離から撮影対象物体が離れるほどに大きくなる．

図 1(a) に示した観測画像に対して焦点ぼけが起こると，図 1(e) に示すような画像が得られる．全方向のエッジで高周波成分が失われ，図 1(f) に示す周波数スペクトルでは，全方向に高周波な成分が失われていることがわかる．

動きぶれや焦点ぼけが生じると，観測画像の高周波成分が抑制されることになる．観測画像にこれらを生じさせないためには，カメラと対象物体を撮影中に移動させない，露光時間を短くする，焦点距離を適切に調整するなどして観測画像を得ることが重要となるが，これらが難しい状況がある．たとえば，拡張現実感実現のためには，移動環境で撮影を行うことが多く，動きぶれを生じさせないことは難しい．また，撮影対象が複数あり，それらとカメラまでの距離が異なる場合には，両方の撮影対象物体に焦点を合わせることはできない．従来の画像マーカは，高周波成分であるエッジでパターンを表現するために，動きぶれや焦点ぼけのある観測画像上でマーカの位置・姿勢の推定をすることは難しいことがある．

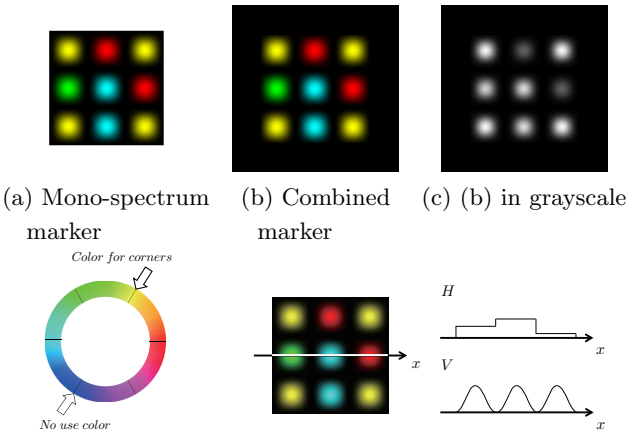
4. モノスペクトルマーカの設計と抽出

4.1 でマーカの設計手法について述べ，4.2 でマーカの抽出手法について述べる．マーカの抽出には，画像上で

のマーカ位置を求める “検出” と，どのマーカが観測されているかを照合する “識別” を含む．

4.1 マーカ設計

我々は，図 2(a) に示すように色が単一の低周波で変化するマーカを設計した．これを**モノスペクトルマーカ**と呼ぶ．単一の低周波成分しか持たないマーカは，動きぶれや焦点ぼけなどによる観測画像の高周波抑制の影響を受けにくい．単一の低周波で構成されることを特徴として観測画像上を探索すれば，マーカ領域を検出することができる．



(a) Mono-spectrum marker　(b) Combined marker　(c) (b) in grayscale

(d) Selected colors　(e) Color in HSV color space

図 2　モノスペクトルマーカの設計
Fig. 2　Design of mono-spectrum marker.

4.1.1 色の決定

モノスペクトルマーカは，HSV 色空間で各位置の色を決定する．

(1) 色相 H の設計

図 2(d) に示すように，ある H の値を，マーカの四隅の色として定義する．本稿では，黄色 ($H = 60$) をこの色とする．H の値を手掛かりのひとつとしてマーカの四隅を検出するので，マーカ以外の領域の色としてあまり現れない色が望ましい．仮に現れたとしても，単一の低周波でないような領域でなければ四隅としては検出されないので，あまり問題とはならない．

また，青周辺の色は明度 V が低く観測されやすく，4.2. で示す手法での検出が難しいので，使用しないことにする．他の色を四隅以外の色として用いる．

色の並びをパターンとしてマーカを識別するので，それぞれの色が区別可能なように，H の値が観測画像からでも識別可能となるように色を選択した．本稿では，実験的に求めた識別可能な色として $H = 60n (n = 0, 2, 3, 5)$ となる 4 色を採用した．

(2) 彩度 S の設計

彩度 S は，それぞれの色で最大となるように設定する．

(3) 輝度 V の設計

輝度 V は，図 2(e) に示すように単一の低周波で変化するように設計する．マーカ検出のときには，V が単一の低周波で変化することを利用することができる．

4.1.2 ARToolKit のマーカとの併用

マーカの位置は，低周波成分で構成される領域の中で，極大値を取る位置によって定義することになる．低周波信号で極大値を取る位置を検出する精度は，高周波成分であるエッジ位置を検出する精度に比べて劣る．たとえば，ARToolKit のマーカでは，高周波成分であるコーナーによってマーカの位置を定義しており，焦点ぼけや動きぶれがないときには，モノスペクトルマーカよりも高い検出精度が期待できる．

そこで，図 2(b) に示すように ARToolKit のマーカと併用することで，焦点ぼけや動きぶれがないときには ARToolKit のマーカとして検出されるようにすることを提案する．ARToolKit であるためのマーカの条件は，(1) 黒の矩形枠を持つことと，(2) 枠内部のパターンがマーカごとに異なることである．ARToolKit の検出および識別は，グレースケールに変換して行われる．図 2(b) に示すマーカは，グレースケールに変換すると図 2(c) に示すように見え，ARToolKit のマーカであるための 2 つの条件を満たす．

ARToolKit では，検出されたマーカの信頼度を内部で計算している．この信頼度が十分に高い場合には，ARToolKit で求められるマーカ位置・姿勢を採用する．この信頼度が低い，あるいは，そもそもマーカとして検出されない場合には，モノスペクトルマーカとして求めたマーカ位置・姿勢を採用する．これによって，焦点ぼけや動きぶれがあってもなくても，精度よくマーカの位置・姿勢を求めることができるようになる．

4.2 マーカ抽出

4.1 で設計されたモノスペクトルマーカは，以下の条件を満たすように設計された．

（1） 四隅の領域の H は指定された値である
（2） マーカごとに格子点上での H の並びが異なる
（3） S は領域内で高い値である
（4） 単一低周波で V が変化する

これらのことを利用して，観測画像からモノスペクトルマーカの検出と識別を行う．

図 3 にモノスペクトルマーカ検出と識別の流れを示す．我々の提案するモノスペクトルマーカは，図中に示す 1. から 4. の手順で抽出する．

1. 直流成分の除去

モノスペクトルマーカの持つ信号を $f(x_o)$，その他の一般物体が持つ信号を $g(x_o)$ と表記する．x_o は画像に投影される前の座標系であることを表す．

観測画像上でのモノスペクトルマーカの領域 M の信号とそれ以外の一般領域 N の信号には，射影による周波

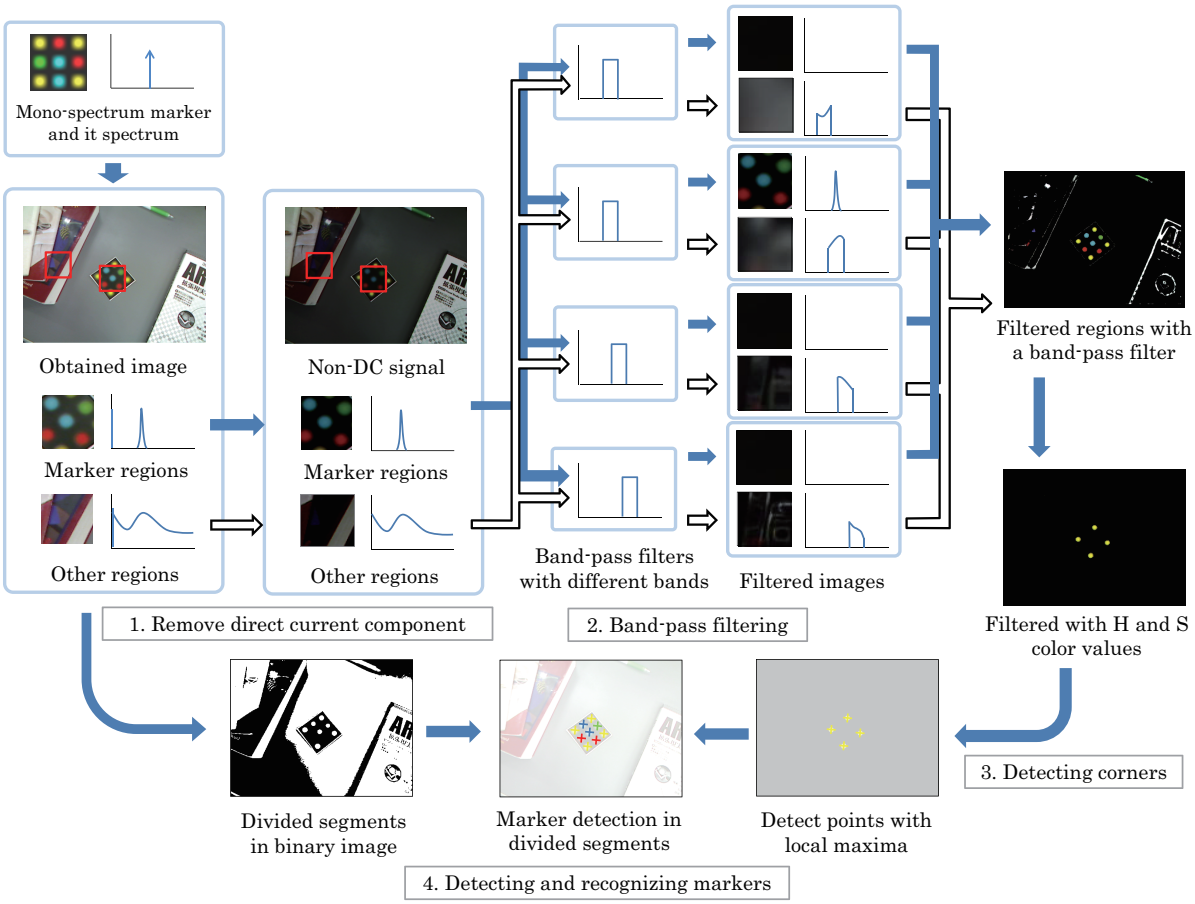


図 3　観測画像からのモノスペクトルマーカの抽出
Fig. 3　Extraction of mono-spectrum marker from observed images.

数スペクトルの変化と，光源環境によって異なる信号のオフセットの付加，および，レンジの変化が起こる．式で書き表すと，

$$M: \quad a\tilde{f}(x_p) + c \tag{1}$$

$$N: \quad a\tilde{g}(x_p) + c \tag{2}$$

となる．$\tilde{f}$，$\tilde{g}$ は周波数スペクトルが射影によって変化したことを表し，a はレンジの変化，c はオフセットを表す．x_p は画像に投影された後の座標系であることを表す．手順 1. では，オフセットである c を除去するために，直流成分の除去を行う．直流成分の除去は，直流以外の周波数成分を通過させるハイパスフィルタ $b_0(x_p)$ を画像に畳み込むことによって計算する．

$$M: \quad (a\tilde{f}(x_p) + c) * b_0(x_p) = a\tilde{f}(x_p) \tag{3}$$

$$N: \quad (a\tilde{g}(x_p) + c) * b_0(x_p) \tag{4}$$

$f(x_o)$ に直流成分は含まれず，周波数スペクトルが多少変化した $\tilde{f}(x_p)$ にも直流成分が含まれないことを期待することができる．よって，$b_0(x_p)$ を畳み込むことで M では $a\tilde{f}(x_p)$ が通過することになる．$g(x_o)$ に直流成分は含まれることがあるので，同様の結果は得られない．

フーリエ変換を用いた周波数領域でのフィルタ処理ではなく，空間領域での畳み込みによるフィルタ処理を採用したのには，以下の 2 つの理由がある．一つ目は，有限サイズのウィンドウでフィルタを畳み込むことで，観測画像上での各小領域での周波数フィルタリングを実現し，フィルタリングの精度を向上させるためである．二つ目は，GPU の実装によって各小領域ごとで並列計算が可能であり，高速な処理が期待できるためである．

2. バンドパスフィルタの適用

次に，観測画像上の各領域に対してバンドパスフィルタを適用する．

モノスペクトルマーカの信号 $f(x_o)$ を構成する周波数成分は既知であるが，射影による影響を受けた $\tilde{f}(x_p)$ の周波数成分は未知である．そこで，通過範囲の異なる複数のバンドパスフィルタ $b_i(i = 1, \cdots, n)$ を適用させる．信号がある周波数に近い成分しか含まないのであれば，いずれかのバンドパスフィルタでは，損失なく信号が通過することが期待できる．観測画像を $s(x_p)$ で表記すると，以下の式で求められる領域をモノスペクトルマーカの領域の候補として検出することができる．

$$\{x_p | \exists i, ||(s * b_0 * b_i)(x_p) - (s * b_0)(x_p)|| < th\} \tag{5}$$

ただし，th は信号の損失を許容する閾値である．

バンドパスフィルタの設計は，周波数領域で通過範囲を決定し，これに逆フーリエ変換を適用することで行える．観測画像上で N 画素で 1 周期となるような信号を通過させるフィルタは，周波数領域で原点を中心に半径 $1/N$ の円周上で 1 を取るスペクトルとして設計できる．

バンドパスフィルタを空間領域で畳み込む処理は，GPU を用いて高速に行うことができる．CUDA の記述法を用いたマーカ候補領域検出の疑似コードを図 4 に示す．

```
Apply b0:
  xp = threadid.x; yp = threadid.y;
  sum0 = 0.0;
  for xw,yw in filterWindow
     sum0 = sum0 + s[xp+xw, yp+yw] * b[0][xw, yw];
  end
  s0[xp,yp] = sum0;

Detect marker candidate region:
  xp = threadid.x; yp = threadid.y;
  markerRegion is null
  for i in 1 to n
     sum = 0.0;
     for xw,yw in filterWindow
        sum = sum + s0[xp+xw, yp+yw] * b[i][xw, yw];
     end
     if abs(sum – s[xp,yp]) < th
        markerRegion = markerRegion + (xp,yp)
     end
  end
```

図 4　マーカ領域候補検出の疑似コード

Fig. 4　Pseudo-code for detectiong marker candidate regions.

3. マーカ四隅の抽出

マーカ四隅の領域の色は，H が指定された値であり，S は大きな値に設定されている．H および S の値で，2. で検出されたマーカ領域を絞り込む．絞り込んだ領域の中で，極大値を取る位置が四隅の領域の中心位置である．適切なウィンドウサイズを設定し，ウィンドウ内部で極大値を取るような座標をこの四隅の中心座標の候補として抽出する．四隅の中心座標候補は，以下の式で求められる．

$$\{x_p | \forall x_q \in window(x_p), s(x_p) > s(x_q)\} \tag{6}$$

観測画像を平滑化しておくことで，画像のノイズを除去し，四隅の中心位置以外で誤って極大値を取ることを防ぐことができる．

極大値を求めるこの処理も，GPU 上で実装することで，画素ごとの並列計算が可能である．

4. マーカの検出と識別

誤って抽出されたマーカの四隅以外の点を除外するために，観測画像を二値化したものについて領域分割を行い，それぞれの分割領域の適当な位置にマーカの四隅の候補点があるかどうかを調べる．この処理は，ARToolKit でも採用されている方法である．分割領域内でマーカ四隅の候補点が 3 点以下しか検出されなかったり，分割領域内で偏った位置にのみマーカ四隅の候補点が抽出される場合には，この分割領域をマーカ領域から除外する．この処理によって，複数マーカが観測されたときにも対応できるようになる．

各マーカ領域について，四隅の観測位置から原画像への平面射影変換を求めることができる．マーカの位置・姿勢はこのカメラ座標系からの平面射影変換で表現できるので，この変換を求めることで重畳表示すべき仮想物体の位置・姿勢がわかる．また，マーカ内部のパターンを照合することもできる．図中のマーカでは，3×3 の領域を設定しているので，3×3 の格子点での H の値を調べることで，マーカのパターンを識別する．どのマーカが観測されているかを識別できれば，どの仮想物体を重畳表示すべきかがわかる．

5. 実　　験

実験では，図 2(b) に示したモノスペクトルマーカと ARToolKit のマーカを併用するものを観測した．実験には Logicool 社製ウェブカメラ Qcam S7500 を用い，解像度 640×480，フレームレート 15fps で撮影を行った．処理端末は，CPU は Intel Core 2 Duo 2.66GHz，GPU は NVidia GeForce GTX295 896MB を搭載したものを用いた．バンドパスフィルタには，実験的に求めた 2 種類のものを採用した．2 種類のバンドパスフィルタは，観測画像上で 7 画素から 15 画素で 1 周期となるもの，15 画素から 31 画素で 1 周期となるものをそれぞれ通過させるものである．

まず，カメラを動かしながら観測画像列 $s(t)(t = 0, \cdots, N)$ を獲得した．このとき，動きを小さくし，かつ，焦点をマーカに合った状態で撮影することで，動きぶれや焦点ぼけがほとんどない観測画像列となるようにした．

次に，以下の手順で撮影した画像列に対して動きぶれと焦点ぼけを人工的に付加した．動きぶれのある観測画像列 $s_B(x_p, t)$ は，$s(x_p, t)$ に近い時刻のフレームの画素値を，時間方向に重みをつけて足し合わせることで計算する．

$$s_B(x_p, t) = \frac{\sum_{u \in \mathcal{N}(t)} s(x_p, u) exp(-d_t(t, u)^2/\sigma_t^2)}{\sum_{u \in \mathcal{N}(t)} exp(-d_t(t, u)^2/\sigma_t^2)} \tag{7}$$

$d_t(\cdot, \cdot)$ はフレーム間の距離であり，$\mathcal{N}(t)$ は，動きぶれで影響を考慮する隣接フレームである．実験では，$d_t(t, u) = ||t - u||, \mathcal{N}(t) = \{t, t-1, \cdots, t-30\}$，$\sigma_t = 1.0$ とした．

焦点ぼけのある観測画像列 $s_D(x_p, t)$ は，$s(x_p, t)$ を画

像内の近傍画素の値に重みをつけて足し合わせることで計算する．

$$s_D(x_p,t)=\frac{\sum_{x_q\in\mathcal{N}(x_p)} s(x_q,t)exp(-d_x(x_p,x_q)^2/\sigma_x^2)}{\sum_{x_q\in\mathcal{N}(x_p)} exp(-d_x(x_p,x_q)^2/\sigma_x^2)} \tag{8}$$

$d_x(\cdot,\cdot)$ は座標間の距離であり，$\mathcal{N}(x_p)$ は，影響を考慮する隣接座標である．実験では，$d_x(x_p,x_q)=||x_p-x_q||, \mathcal{N}(x_p)=\{x_q|d_x(x_p,x_q)\leqq 20\}$，$\sigma_x=6.7$ とした．

それぞれの画像列に対する実験結果を図 5 と図 6 に示す．図 5(a) と図 6(a) は，動きぶれと焦点ぼけをそれぞれ付加した画像の例である．元の観測画像列で ARToolKit によって検出されたマーカ中心位置の座標を正解として，それぞれの生成画像列に対して検出されたマーカ中心位置の座標との誤差を調べた．

図 5(b) は，動きぶれがある画像列に対して，ARToolKit のマーカとして中心位置を検出した結果である．横軸がフレーム番号，縦軸が正解からの誤差である．ただし，マーカを検出することができなかったフレームでの誤差は，正の無限大としてグラフを作成している．いくつかのフレームに対して，マーカを抽出することができなかったことがわかる．検出できたフレームでの誤差の平均と分散は，表 1 に示す通りである．対して，図 5(c) は，提案するモノスペクトルマーカとして位置を検出した結果である．ただしこのとき，ARToolKit によって検出される位置・姿勢は利用していない．すべてのフレームでマーカの位置を検出することができ，ARToolKit のマーカとして検出した位置よりも誤差も小さいことがわかる．動きぶれがあるときには，動きと垂直な方向のエッジが検出しにくくなり，結果として ARToolKit のマーカとして検出した位置は精度が悪くなる．また，原理的には動きぶれの影響を受けないはずのモノスペクトルマーカにも誤差が生じており，4.2 で説明したように ARToolKit と併用することが必要であることも確かめられた．図 5(d) は，モノスペクトルマーカによって検出したマーカ位置に合わせてティーポットを重畳表示した画像である．マーカ位置・姿勢が正しく推定されていることがわかる．

さらに，解像度を 320×240 に縮小して，再度マーカ位置の検出を行った．観測画像列が低解像度であるときには，高周波成分がさらに失われることになり，高周波成分であるエッジ位置の検出精度が下がることが予想された．図 5(e) は ARToolKit のマーカとして位置を検出した結果であり，図 5(f) はモノスペクトルマーカとして位置を検出した結果である．観測画像列が低解像度で与えられる場合にも，高解像度の場合と同じ傾向が見られた．ARToolKit のマーカとして検出したときには，いくつかのフレームでマーカの検出に失敗した．しかし，空間的なぶれの拡がりが小さくなることによって，ARToolKit によってマーカが検出できないフレームは予想に反して減少した．しかしながら，ARToolKit のマーカとして検出した位置よりも，モノスペクトルマーカとして検出した位置の方が依然として精度が高いこともわかる．一方で，モノスペクトルマーカとしての位置検出精度は，高解像度の場合からあまり変化が見られなかった．このことから，観測画像列を低解像度化することによって，モノスペクトルマーカ抽出の高速化が図れるのではないかと期待できる．

表 1　マーカ検出位置の誤差と処理時間 (ms/frame)
(*: マーカが検出されなかったフレームを含まない)

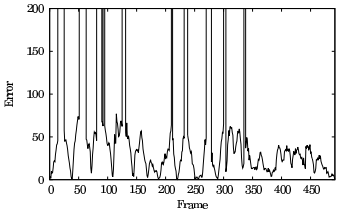
Table 1　Errors and processing time (ms/frame) of marker detection. (*: Not included frames in that markers could not detected.)

	Resolution of images					
	640 × 480			320 × 240		
	Ave.	Var.	Time	Ave.	Var.	Time
ARToolKit marker (Blur) *	27.8	320.0	4.9	22.5	233.0	1.8
ARToolKit marker (Defocus)	42.6	12.4	4.3	45.5	23.1	1.5
Mono-spectrum marker (Blur)	18.5	90.0	90.0	21.4	104.0	23.1
Mono-spectrum marker (Defocus)	18.3	85.2	88.8	22.0	100.9	22.8

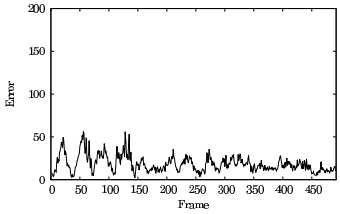
図 6(b) は，焦点ぼけがある画像列に対して，ARToolKit のマーカとして位置を検出した結果である．すべてのフレームに対して，マーカを抽出することができたものの，誤差は高いところで安定した．対して，図 6(c) は，提案するモノスペクトルマーカとして位置を検出した結果である．このとき，動きぶれのときと同様に，ARToolKit によって検出される位置・姿勢は利用していない．すべてのフレームでマーカの位置を検出することができ，ARToolKit のマーカとして検出した位置よりも誤差も小さいことがわかる．焦点ぼけがあるときには，すべての方向のエッジが検出しにくくなり，ARToolKit のマーカは実際よりも遠くに見える効果があった．このため，誤差は全体的に大きくなった．モノスペクトルマーカでは，動きぶれの場合と同様に高い精度で検出できていることが確かめられた．図 6(d) は，モノスペクトルマーカによって検出したマーカ位置に合わせてティーポットを重畳表示した画像である．焦点ぼけがある画像に対しても，マーカ位置・姿勢が正しく推定されていることがわかる．

さらに，解像度を 320×240 に縮小して，再度マーカ位置の検出を行った．図 6(e) は ARToolKit のマーカとして位置を検出した結果であり，図 6(f) はモノスペクトルマーカとして位置を検出した結果である．観測画像列が低解像度で与えられる場合にも，高解像度の場合と同じ傾向が見られた．ARToolKit のマーカとして検出したときにもすべてのフレームでマーカの検出に成功したものの，ARToolKit のマーカとして検出した位置よりもモ

(a) An example of blurred image

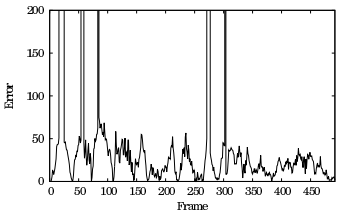


(b) Error in frames by ARToolKit marker

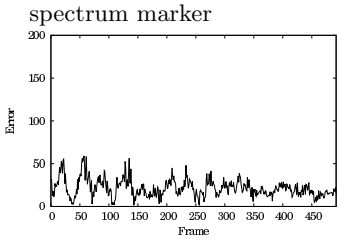


(c) Error in frames by mono-spectrum marker

(d) A virtual object at the position estimated with mono-spectrum marker



(e) Error in frames by ARToolKit marker (Low resolutional images)

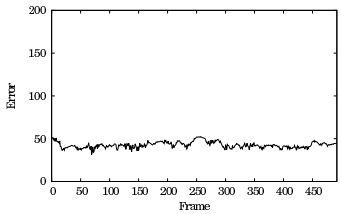


(f) Error in frames by mono-spectrum marker (Low resolutional images)

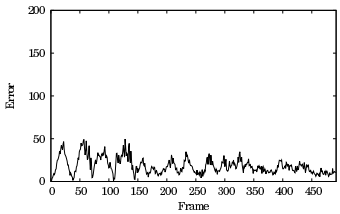
図 5　動きぶれを含む画像列に対するマーカ位置推定結果

Fig. 5　Experimental results of marker detection for blurred images.

(a) An example of defocused image

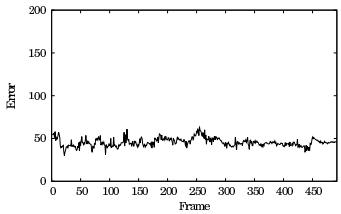


(b) Error in frames by ARToolKit marker

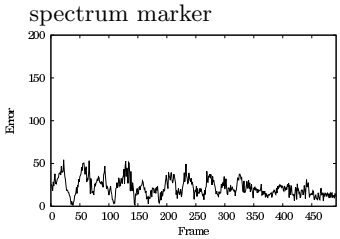


(c) Error in frames by mono-spectrum marker

(d) A virtual object at the position estimated with mono-spectrum marker



(e) Error in frames by ARToolKit marker (Low resolutional images)



(f) Error in frames by mono-spectrum marker (Low resolutional images)

図 6　焦点ぼけを含む画像列に対するマーカ位置推定結果

Fig. 6　Experimental results of marker detection for defocused images.

ノスペクトルマーカとして検出した位置の方が精度が高かった．動きぶれの場合と同様に，モノスペクトルマーカとしての位置検出精度は，高解像度の場合からあまり変化が見られなかった．

処理時間は，表 1 に示す通りである．観測画像を 640 × 480 にした場合には，毎秒 11 フレーム以上で処理が実現し，実時間処理の可能性を示すことができた．この処理速度は，GPU の性能向上や専用チップを利用することによって，さらに向上する可能性がある．また，解像度を 320 × 240 にした場合には毎秒 42 フレームの処理が可能であり，位置検出精度がそれほど落ちていないことから，処理速度向上のために，適切な解像度での処理を行うことが有効であるといえる．

上述の実験では，モノスペクトルマーカ内の各色領域が 1 周期が 7 画素から 31 画素で投影される場合に，マーカを検出することができる．バンドパスフィルタの数を増やすと，より広範な奥行きのマーカを検出することができるようになるが，それだけ処理時間は増加する．そこで，バンドパスフィルタの数を変えたときにどれだけ処理時間が増加するかについても調べた．図 7 にバンドパスフィルタの数と処理時間の関係を示す．解像度が 640×480 の場合には，フィルタが 1 枚増えるごとに 1 フレームごとにおよそ 20ms 増加し，解像度が 320×240 の場合には，フィルタが 1 枚増えるごとに 1 フレームごと

におよそ 6ms 増加することがわかる．マルチ GPU が利用できる環境であれば，GPU の数だけバンドパスフィルタ処理を並列化できるので，処理時間を増やすことなく処理が可能となることが期待できる．

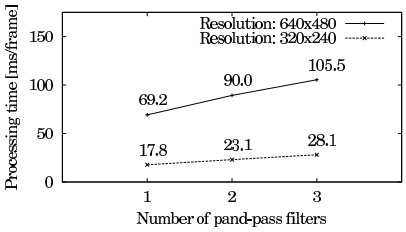


図 7　バンドパスフィルタの数と処理時間
Fig. 7　Number of band-pass filters and processing time.

6. ま と め

本研究では，動きぶれや焦点ぼけがある画像からでも抽出できる拡張現実マーカとして，単一の低周波成分で構成したモノスペクトルマーカを提案した．モノスペクトルマーカの検出には，通過帯域の異なる複数のバンドパスフィルタを用いた．実験結果から，動きぶれや焦点ぼけがある画像からでも高速に抽出できることが確かめられた．

今後の課題として，モノスペクトルマーカ抽出の信頼度の計算を検討することを挙げる．4.1.2 において，ARToolKit のマーカと併用することを提案し，実験においてもモノスペクトルマーカと ARToolKit のマーカの両方の性質を持つマーカを利用した．しかし，どちらの推定結果を利用するかを抽出の信頼度によって選択する手法については，まだ実現していない．どれほどの動きぶれや焦点ぼけによって，どれほど ARToolKit のマーカの推定精度が低下するのか，モノスペクトルマーカの推定結果がどの程度で優位になるのかを調べる必要がある．信頼度が適切に計算できれば，両方のマーカの性質を生かしたマーカの利用が実現できる．動きぶれや焦点ぼけの度合いについては，Okumura ら [15] の手法に倣って，マーカのエッジがどれほどなだらかになるかによって調べることができる．

また，GPU を搭載しない一般端末での処理の高速化も今後の課題として挙げる．本稿では，バンドパスフィルタを画素ごとに並列計算するために，GPU を用いた．ARToolKit に代表される拡張現実マーカの大きな特徴として，使用するメモリが少なく，計算量も小さいことがある．この点において，提案するモノスペクトルマーカは処理に GPU を要求することで，利用できる場面を狭めているといえる．この問題は，携帯端末にも音声処理のために搭載されている DSP(ディジタルシグナルプロセッサ) を利用することで解決できるのではないかと考えている．DSP は高速な周波数フィルタリングなどの機能を提供するものであるので，モノスペクトルマーカの検出に利用できるのではないかと考えている．また一方で，GPU の近年の目覚ましい発展を受けて，携帯電話にも GPU を搭載する動きがあり，これが一般的に普及すれば，これを利用することも考えられる．今後の携帯端末の発展の動向にも注目したい．

文　　献

[1] G. Klein, and D. Murray, “Parallel tracking and mapping for small ar workspaces,” International Symposium on Mixed and Augmented Reality (ISMAR), 2007.

[2] 黒田嘉宏，金守恒志，滝内秀和，田ノ岡征雄，井村誠孝，黒田知宏，大城理，“AR 手術のための直線対応体内レジストレーション,” 日本バーチャルリアリティ学会論文誌,, vol.14, no.4, pp.435–444, 2009.

[3] H. Kato, and M. Billinghurst, “Marker tracking and hmd calibration for a video-based augmented reality conferencing system,” International Workshop on Augmented Reality, pp.85–94, 1999.

[4] M. Fiala, “Artag, a fiducial marker system using digital techniques,” IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp.590–596, 2005.

[5] H. Uchiyama, and H. Saito, “Random dot markers,” IEEE Virtual Reality (IEEE VR), 2011.

[6] K. Tateno, I. Kitahara, and Y. Ohta, “A nested marker for augmented reality,” IEEE Virtual Reality, pp.259–262, 2007.

[7] A. Mohan, G. Woo, S. Hiura, Q. Smithwick, and R. Raskar, “Bokode: Imperceptible visual tags for camera based interaction from a distance,” ACM Transactions on Graphics, vol.28, no.3, 2009, Article 98.

[8] D.G. Lowe, “Distinctive image features from scale-invariant keypoints,” International Journal of Computer Vision, vol.60, no.2, pp.91–110, 2004.

[9] H. Bay, A. Ess, T. Tuytelaars, and L.V. Gool, “Surf: Speeded up robust features,” Computer Vision and Image Understanding, vol.110, no.3, pp.346–359, 2008.

[10] Y. Uematsu, and H. Saito, “Improvement of accuracy for 2d marker-based tracking using particle filter,” International Conference on Artificial Reality and Telexistence (ICAT), pp.183–189, 2007.

[11] Y. Uematsu, T. Teshima, H. Saito, and C. Honghua, “D-calib: Calibration software for multiple cameras system,” International Conference on Image Analysis and Processing (ICIAP), pp.285–290, 2007.

[12] Q. Shan, J. Jia, and A. Agarwala, “High-quality motion deblurring from a single image,” ACM Transactions on Graphics, vol.27, no.3, 2008.

[13] H. Asai, Y. Oyamada, J. Pilet, and H. Saito, “Cepstral analysis based blind deconvolution for motion blur,” IEEE International Conference on Image Processing (ICIP), 2010.

[14] 田中正行，神田崇史，奥富正敏，“残差画像に基づく漸進的ぶれ画像復元,” 電子情報通信学会論文誌 D，vol.J92-D，no.8，pp.1208–1220，2009.

[15] B. Okumura, M. Kanbara, and N. Yokoya, “Augmented reality based on estimation of defocusing and motion blurring from captured images,” International Symposium on Mixed and Augmented Reality (ISMAR), pp.219–225, 2006.