



断層映像法の基礎 第28回 マルチグリッドレジストレーション

篠原 広行¹⁾、伊藤 猛¹⁾、軽部 修平¹⁾、小島 慎也¹⁾、橋本 雄幸²⁾

¹⁾ 首都大学東京人間健康科学研究科 放射線科学域

²⁾ 横浜創英短期大学 情報学科

はじめに

第27回では、非剛体レジストレーションに使われているマルチグリッドレジストレーションにおいて、計算コストのために利用される解像度変換について解説した。マルチグリッドに解像度変換を用いてレジストレーションを行うことで計算コストを軽減することができる。今回は解像度変換を行った上でのマルチグリッドレジストレーションについて解説する。マルチグリッドにしてレジストレーションすることで非剛体の位置合わせを行うことができる。

1. マルチグリッドへの変換
2. 解像度を下げた剛体レジストレーション
3. 変換ベクトルの算出と位置合わせ
4. 非剛体レジストレーションの例

1. マルチグリッドへの変換

マルチグリッドへ変換する方法を2次元で考える。最も単純な方法は、半分に区切っていく方法である。第1段階は画像そのままの区切らない段階である。第2段階では縦横ともにそれぞれ2分割し、 2×2 の4分割にする。第3段階ではさらにそれぞれのグリッドを 2×2 の4分割にして、全体では 4×4 の16分割にする。16分割にした第3段階では 256×256 画素の画像では、1つのグリッドが 64×64 画素になる。さらに分割していくことができるが、あまりグリッドを細かくしすぎると、グリッドごとのレジストレーションが困難になるので、ある程度ところで終了する。以上の方法で 256×256 画素の画像を、マルチグリッドへ変換する手順を図1に示す。

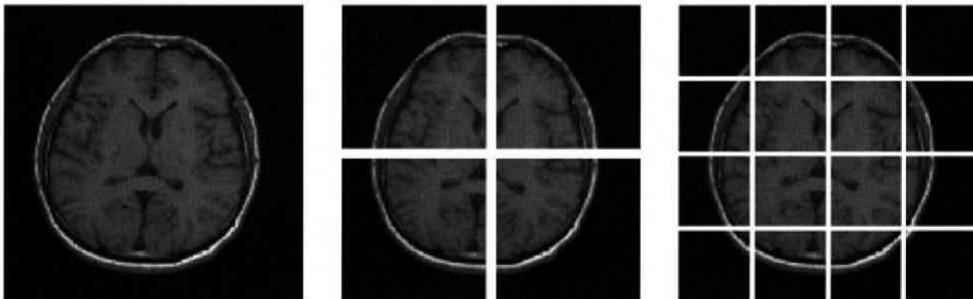


図1. 256×256 画像をマルチグリッドへ変換する手順

(a) 第1段階：原画像 (b) 第2段階： 2×2 の4分割 (c) 第3段階： 4×4 の16分割

(a) | (b) | (c)

別刷請求先：〒116-8551 東京都荒川区東尾久7-2-10
首都大学東京人間健康科学研究科放射線科学域 篠原 広行
TEL：03-3819-1211 FAX：03-3819-1406

マルチグリッドを利用した非剛体の位置合わせを行なうときに、第1段階からそのままの画素で行うのは計算コストの効率が悪いので、対象画像を低解像度に変換して全体の画素数を減らし、グリッドごとに剛体レジストレーションを行うと効率が良くなる。その後、グリッド内の解像度を回復させ、その中をさらにグリッドに分けて剛体レジストレーションを行っていく。低解像度への変換を調節することにより、グリッド内の画素数を変えずに効率よい剛体レジストレーションの計算を行なうことができる。

256×256画素の画像でマルチグリッドを第3段階まで行くとすると、図2に示すように第1段階で第3段階での画素数と同じ64×64画素まで解像度を下げれば、画像全体においても同じ画素数でのレジストレーションを行うことになる。第2段階では、解像度を第1段階の2倍に上げることで全体が128×128画素になり、同じ64×64画素のレジストレーションを4回行えばよいことになる。第3段階では最初の解像度と同じ全体が256×256画素になり、各領域の画素数は64×64画素と同じなので、そのままレジストレーションを16回行うことになる。これで第1段階から第3段階まで同じ画素数同士のレジストレーションを行うことになり、計算コストを軽減させることができる。

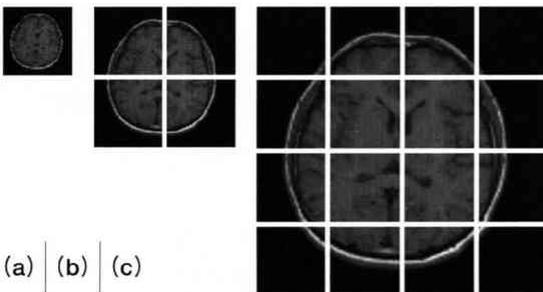


図2. 第1段階から第3段階までの解像度変換
すべての段階で1つのグリッドが64×64画素になっている。
(a) 第1段階：原画像を64×64画素の1/4に解像度変換する
(b) 第2段階：2×2の4分割のグリッドを64×64画素の1/2に解像度変換する
(c) 第3段階：4×4の16分割のグリッドはそのままの解像度

2. 解像度を下げた剛体レジストレーション

マルチグリッドにおける第1段階は画像全体のレジストレーションとなるので、通常の剛体レジストレーションと同じものとなる。ただし、計算コストを考慮して解像度を下げた場合、低解像度のためレジストレーションの精度は悪くなる。非剛体のレジストレーションの場合、他の段階のレジストレーションで画像を一致させていくので、第1段階のレジストレーションは大まかな位置合わせができればよい。よって、第1段階でのレジストレーションの精度が多少悪くなくても問題はない。

第1段階では、平行移動と回転移動のレジストレーションを行う必要がある。シンプレックス法などで同時に位置合わせを行う場合は、平行移動と回転移動の移動単位を合わせる必要がある。レジストレーションの対象となる画像が $n \times n$ 画素の場合、平行移動で1画素を単位とすると、回転移動の単位 $d\theta$ は、

$$d\theta = \frac{\pi}{n} \quad (1)$$

となる。平行移動の1画素と回転移動の $d\theta$ を移動単位としてレジストレーションを行うことになる。

第1段階では、全体の平行移動と回転移動の移動量が算出されるので、その値に応じて元の解像度である原画像を移動させる。原画像を移動する際、回転移動はそのままの値でよいが、平行移動はレジストレーション時と解像度が変わるので、原画像の画素を単位とした平行移動量 (dx_0, dy_0) は、

$$\begin{cases} dx_0 = \frac{n}{m} dx \\ dy_0 = \frac{n}{m} dy \end{cases} \quad (2)$$

となる。ここで、原画像を $n \times n$ 画素とし、解像度を落とした画像を $m \times m$ 画素とする。また、解像度を下げた画像で行ったレジストレーションの結果の x 方向の移動量を dx 、 y 方向の移動量を dy としている。

実際に解像度を下げないで剛体レジストレーションを行った結果と、解像度を下げた剛体レジストレーションを行った結果の例を図3に示す。レジストレーションには相互情報量を利用し、シンプレックス法で平行移動と回転移動を同時に行って最適値を求めた結果である。画像ではそれほど違いが見ら

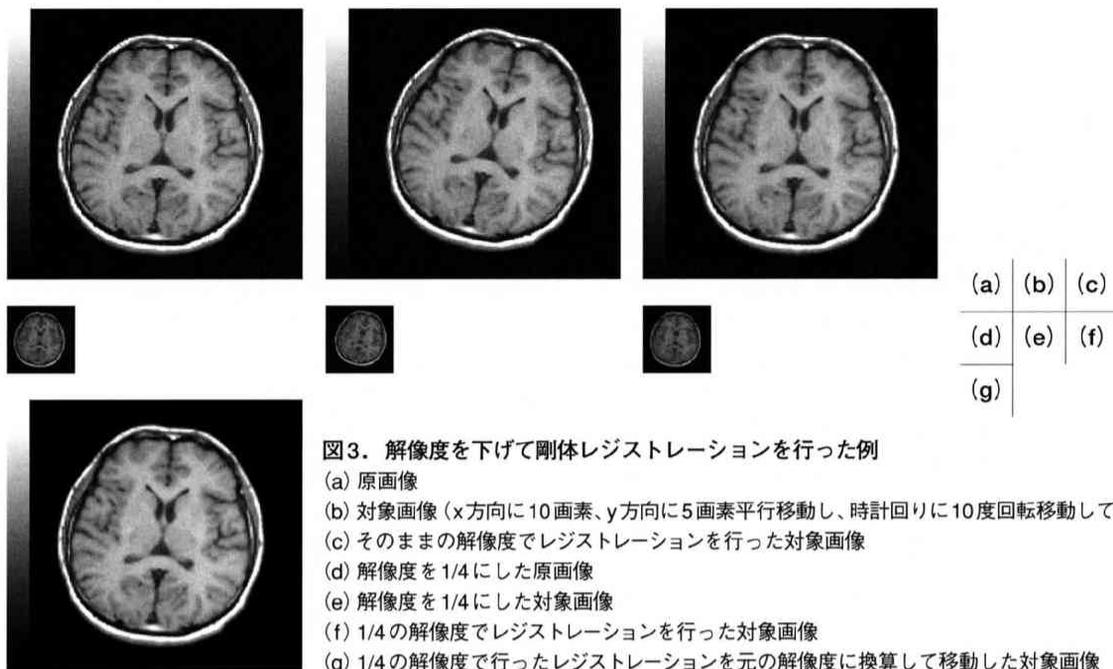


図3. 解像度を下げて剛体レジストレーションを行った例

- (a) 原画像
- (b) 対象画像 (x方向に10画素、y方向に5画素平行移動し、時計回りに10度回転移動している)
- (c) そのままの解像度でレジストレーションを行った対象画像
- (d) 解像度を1/4にした原画像
- (e) 解像度を1/4にした対象画像
- (f) 1/4の解像度でレジストレーションを行った対象画像
- (g) 1/4の解像度で行ったレジストレーションを元の解像度に換算して移動した対象画像

れないが、表1に示したx方向の移動量、y方向の移動量と回転移動量の数値を見ると元の解像度と解像度1/4で元の解像度に換算した値では、若干異なっていることがわかる。数値は今回のシミュレーションの値のみしか表示していないが、ほぼ同じような値でレジストレーションを行えることが伺える。

3. 変換ベクトルの算出と位置合わせ

第2段階からは本来のマルチグリッドでの計算になるので、それぞれのグリッドごとに剛体レジストレーションが行われる。レジストレーションは、簡略化のために平行移動のみで行う。平行移動のみでも、非線形で画像を変換すると結果的に回転移動も含まれてくる。グリッドごとのレジストレーションにおいては、グリッドに分ける際に、画像が切れてしまうところが含まれるので、通常の方法でレジストレーションを行うと、切れた場所の影響が大きくなってしまふ。そこで、レジストレーションを行う2つの画像のうち、片方の画像はグリッドに分けて考え、もう片方の画像はグリッドには分けなくてそのままの画像でレジストレーションを行う。図4に示すように、レジストレーションを行う画像の大きさは異なるが、レジストレーションの際に計算する相互情報量などの評価値は、2つの画像の重なった

表1. 元の解像度と解像度を1/4に下げた場合でのレジストレーションの結果

	x方向の移動 (画素)	y方向の移動 (画素)	回転 (度)
対象画像 (設定値)	10.00	5.00	10.00
元の解像度	10.71	3.19	10.01
解像度1/4	2.64	0.84	9.50
解像度1/4で 元の解像度に 換算	10.56	3.35	9.50

部分のみで行う。こうすることで画像の切れた場所での影響を少なくすることができる。

グリッドごとに平行移動のみで剛体レジストレーションを行うことで、それぞれのグリッドで平行移動ベクトルが算出される。それらの平行移動ベクトルをもとに原画像の変換ベクトルを算出する。

第2段階の2×2の4分割にグリッドを分けた場合、4つの平行移動ベクトルが算出される。それぞれの平行移動ベクトルは各グリッドの中心位置における変換ベクトルと考える。既知の4点の変換ベクトルから他の画素の変換ベクトルを、補間を利用して

算出する。算出の方法はさまざま考えられるが、1つの方法を以下に示す。

図5に示すように、既知の4点の位置をもとに画像を分割して考える。角の4カ所は最も近い既知の変換ベクトルの値と等しくする。画像の辺に隣接する中央外側の4カ所は、その領域と接する2つの既知の変換ベクトルの線形補間で求める。2つの変換ベクトルがそれぞれ (x_0, y_0) と (x_1, y_1) の位置にあるとする。ただし、いずれの場合も x 座標か y 座標のどちらかが等しくなる。 y 座標が等しい2カ所の場合の線形補間は x 座標の値でのみ行えばよいので、それぞれの座標における変換ベクトルを (tx_0, ty_0) と (tx_1, ty_1) とすると、座標 (x, y) における線形補間した変換ベクトル (tx, ty) は、

$$\begin{cases} tx = \frac{x_1 - x}{x_1 - x_0} tx_0 + \frac{x - x_0}{x_1 - x_0} tx_1 \\ ty = \frac{x_1 - x}{x_1 - x_0} ty_0 + \frac{x - x_0}{x_1 - x_0} ty_1 \end{cases} \quad (3)$$

となる。また、 x 座標が等しい残りの2カ所の場合は、

$$\begin{cases} tx = \frac{y_1 - y}{y_1 - y_0} tx_0 + \frac{y - y_0}{y_1 - y_0} tx_1 \\ ty = \frac{y_1 - y}{y_1 - y_0} ty_0 + \frac{y - y_0}{y_1 - y_0} ty_1 \end{cases} \quad (4)$$

となる。最後に4点に囲まれる中央の領域では、その4点から2次元の線形補間を行う。変換ベクトルが存在する4点の座標を (x_0, y_0) 、 (x_1, y_0) 、 (x_0, y_1) 、 (x_1, y_1) とし、それぞれの座標での変換ベクトルを (tx_{00}, ty_{00}) 、 (tx_{10}, ty_{10}) 、 (tx_{01}, ty_{01}) 、 (tx_{11}, ty_{11}) とするとき、座標 (x, y) における線形補間した変換ベクトル (tx, ty) は、

$$\begin{cases} tx = \frac{(x_1 - x) \cdot (y_1 - y)}{(x_1 - x_0) \cdot (y_1 - y_0)} tx_{00} + \frac{(x - x_0) \cdot (y_1 - y)}{(x_1 - x_0) \cdot (y_1 - y_0)} tx_{10} \\ \quad + \frac{(x_1 - x) \cdot (y - y_0)}{(x_1 - x_0) \cdot (y_1 - y_0)} tx_{01} + \frac{(x - x_0) \cdot (y - y_0)}{(x_1 - x_0) \cdot (y_1 - y_0)} tx_{11} \\ ty = \frac{(x_1 - x) \cdot (y_1 - y)}{(x_1 - x_0) \cdot (y_1 - y_0)} ty_{00} + \frac{(x - x_0) \cdot (y_1 - y)}{(x_1 - x_0) \cdot (y_1 - y_0)} ty_{10} \\ \quad + \frac{(x_1 - x) \cdot (y - y_0)}{(x_1 - x_0) \cdot (y_1 - y_0)} ty_{01} + \frac{(x - x_0) \cdot (y - y_0)}{(x_1 - x_0) \cdot (y_1 - y_0)} ty_{11} \end{cases} \quad (5)$$

となる。以上の方法で、第2段階のマルチグリッドレジストレーションから変換ベクトルを求めることができる。

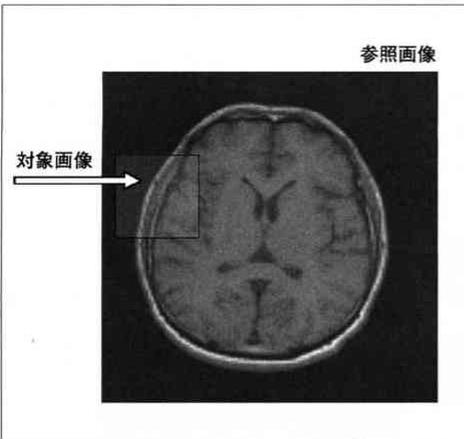


図4. 画像の一部から成る領域のレジストレーション

参照画像はすべての画像を利用し、対象画像は対象領域を自由に移動させて重なった部分だけ相互情報量などの評価値を求める。

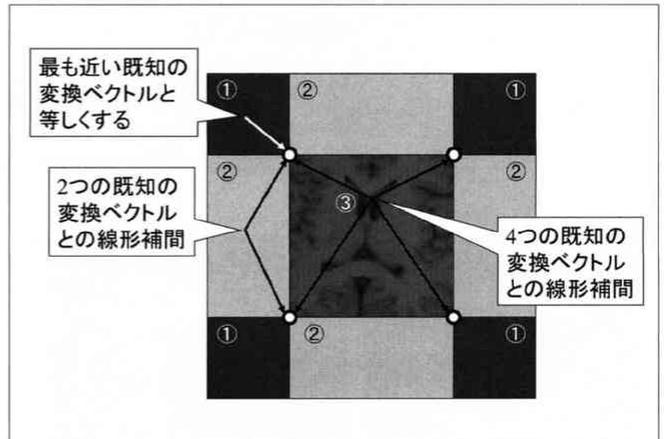


図5. 既知の4点の移動ベクトルをもとに原画像領域全体の移動ベクトルを求める方法

- ① 角の4カ所（濃い灰色）は、最も近い変換ベクトルと等しくする。
- ② 辺中央の4カ所（薄い灰色）は、接する2つの既知の変換ベクトルから1次元の線形補間で求める。
- ③ 残りの中央の部分（画像の見える部分）は、4つの変換ベクトルから2次元の線形補間で求める。

求まった変換ベクトルは、移動後の画像の各画素が原画像のどの位置から移動させればよいかを示している移動ベクトルになっているので、その値をもとに画像を移動させることができる。原画像における移動元の値は、原画像の隣接する値から2次元の補間によって求める。補間の方法で最も単純にできるのは、2次元の線形補間である。2次元の線形補間は(5)式と同じ方法で計算することができる。この移動によって第2段階のレジストレーションが完了する。

第3段階では、4×4の16分割になっており、16個の平行移動ベクトルがそれぞれのグリッドで算出される。全体の変換ベクトルを算出する方法は、第2段階での線形補間と同じ要領で行うことができる。図6に示すように、角の4カ所は最も近い既知の変換ベクトル(算出された移動ベクトル)と等しくする。そのほかの外側の領域は、隣接する2つの既知の変換ベクトルの線形補間で求める。内側

の領域は、隣接する4つの既知の変換ベクトルから2次元の線形補間で求める。

以上の段階を踏んでマルチグリッドレジストレーションが完了する。段階の数は、原画像の大きさとレジストレーションを行うことのできる画素数で決まる。また、グリッドにおけるレジストレーションには、今回紹介したほかにも様々な方法が考えられる。

4. 非剛体レジストレーションの例

以上に示したマルチグリッドレジストレーションの具体的な例を次に示す。

レジストレーションを行う原画像(参照画像)と対象画像を図7に示す。原画像は256×256画素のMRIの画像である。対象画像は原画像を非線形的に歪めて、多少の平行移動と回転移動をしてある。

第1段階では、両方の画像を64×64画素に解像度変換し、その2つの画像で剛体レジストレーションを

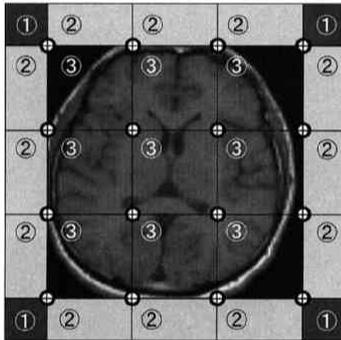


図6. 既知の16点の移動ベクトルをもとに原画像領域全体の移動ベクトルを求める方法

- ① 角の4カ所(濃い灰色)は、最も近い変換ベクトルと等しくする。
- ② 辺に隣接する12カ所(薄い灰色)は、接する2つの既知の変換ベクトルから1次元の線形補間で求める。
- ③ 残りの中央の部分の9カ所(画像が見える部分)は、4つの変換ベクトルから2次元の線形補間で求める。

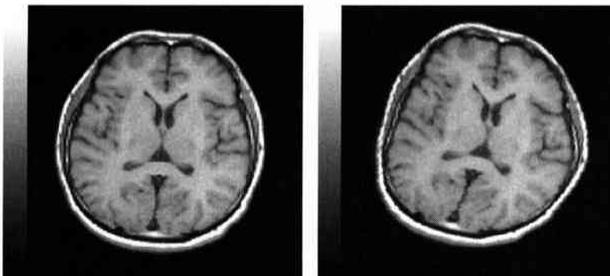


図7. 原画像(参照画像)と対象画像

- (a) 原画像：レジストレーションの際の参照画像となる
- (b) 対象画像：レジストレーションで参照画像と重なるように移動させる

対象画像は原画像を非線形的に歪めて、平行移動と回転移動している。



図8. 第1段階での剛体レジストレーション

- (a) 解像度を1/4にした原画像
- (b) 解像度を1/4にした対象画像
- (c) 1/4の解像度でレジストレーションを行った対象画像
- (d) 1/4の解像度で行ったレジストレーションを元の解像度に換算して移動した対象画像

行う。その結果の画像を図8に示す。非線形的に歪めた元の解像度の対象画像を、その移動量を元にして平行移動と回転移動した画像を図8(d)に示す。

第2段階では、図8(d)に示した画像と参照画像を2×2の4分割にしてそれぞれのグリッドで剛体レジストレーションを行う。レジストレーションを行って、算出された変換ベクトルのx方向の画像とy方向の画像とベクトルの長さである移動量の画像を図9に示す。また、この移動ベクトルから図8(d)に示した対象画像を移動したレジストレーション画像を図10に示す。

第3段階では、図10に示した対象画像と参照画像を4×4の16分割にしてそれぞれのグリッドで剛体レジストレーションを行う。レジストレーションを行って、算出された変換ベクトルのx方向の画像とy方向の画像とベクトルの長さである移動量の画像を図11に示す。また、この移動ベクトルから図10に示した対象画像を移動したレジストレーション画像を図12に示す。

図12に示した画像が今回のマルチグリッドレジストレーションで算出された画像である。ちなみに、第3段階を何回か繰り返すと多少画像が似通って



図9. 第2段階での変換ベクトル
 (a) x方向の変換ベクトル
 (b) y方向の変換ベクトル
 (c) 変換ベクトルの長さ(各画素の移動量)

(a) | (b) | (c)

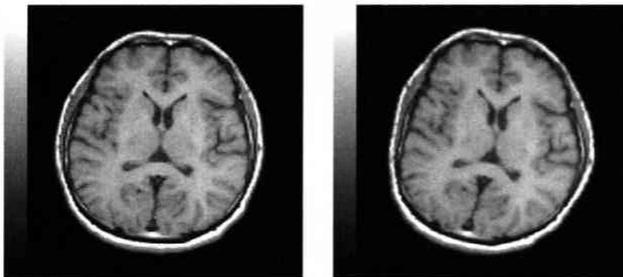


図10. 第2段階でのレジストレーション画像
 (a) 原画像
 (b) 第2段階のレジストレーション画像

(a) | (b)

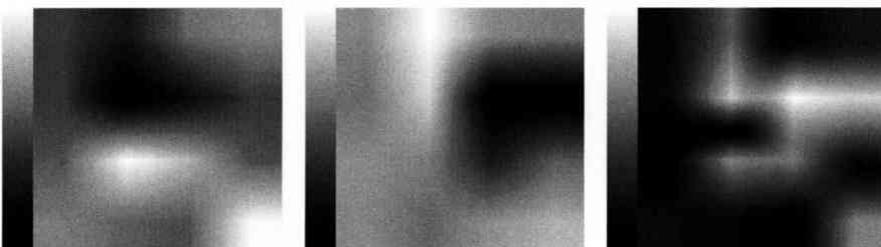


図11. 第3段階での変換ベクトル
 (a) x方向の変換ベクトル
 (b) y方向の変換ベクトル
 (c) 変換ベクトルの長さ(各画素の移動量)

(a) | (b) | (c)

くる。第3段階を2回行ったときの変換ベクトルとレジストレーション画像を図13に、3回行ったときの変換ベクトルとレジストレーション画像を図14に示す。内容をさらに一致させるには、グリッドをもっと細かくする必要がある。いずれにしても画像の濃

度情報だけでは、非剛体レジストレーションを完全に一致させることは困難である。非剛体のレジストレーションでさらに一致させるためには、画像の解剖学的な特徴点など他の様々な情報を加味する必要がある。

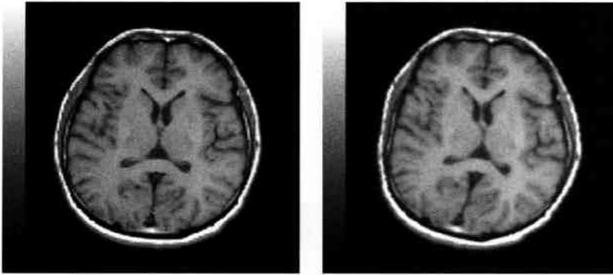


図12. 第3段階でのレジストレーション画像
(a) 原画像 (b) 第3段階のレジストレーション画像

謝辞：本研究で使用したプログラムの開発は平成17年度～平成22年度首都大学東京共同研究費（富士フィルムRIファーマ株式会社）、および平成21年度首都大学東京傾斜的配分研究費によるものである。

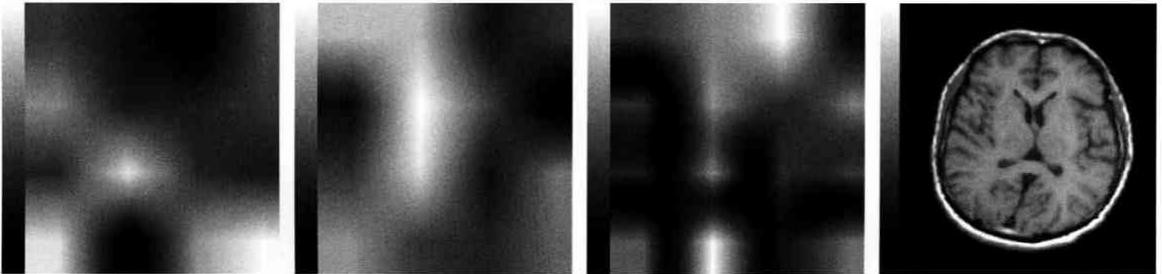


図13. 第3段階を2回行ったときの変換ベクトルとレジストレーション画像
(a) x方向の変換ベクトル (b) y方向の変換ベクトル (c) 変換ベクトルの長さ（各画素の移動量） (d) 第3段階を2回行ったレジストレーション画像

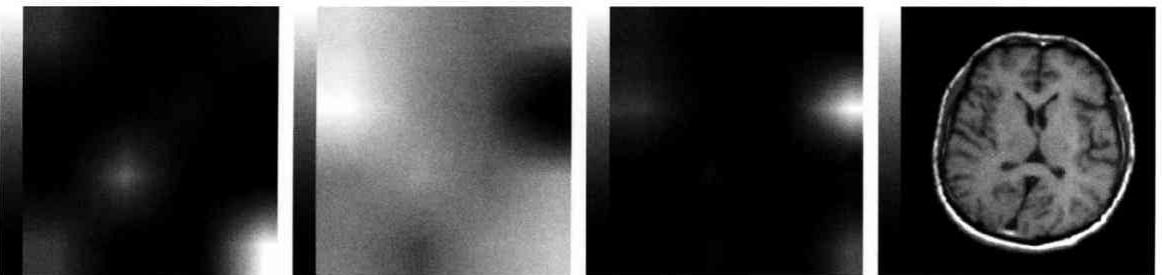


図14. 第3段階を3回行ったときの変換ベクトルとレジストレーション画像
(a) x方向の変換ベクトル (b) y方向の変換ベクトル (c) 変換ベクトルの長さ（各画素の移動量） (d) 第3段階を3回行ったレジストレーション画像