

令和元年度卒業論文

深層学習を用いた特徴点マッチングにおける
回転・拡大縮小不変の実現

宮崎大学 工学部 情報システム工学科

黒河英里子

指導教員 棕木雅之

目次

1. はじめに	1
2. 特徴点マッチング	2
3. 回転・拡大縮小不変	3
4. 深層学習を用いた特徴点マッチングにおける回転・拡大縮小不変の実現	4
4.1 特徴点検出	5
4.2 特徴量記述	6
4.3 マッチング	7
5. 実験	9
5.1 アフィン変換画像を用いた実験	9
5.1.1 実験方法	9
5.1.2 実験結果	12
5.1.3 処理時間	15
5.1.4 深層学習による特徴量記述の有効性	17
5.2 実画像実験	19
5.2.1 実験条件	21
5.2.2 実験結果	21
6. おわりに	23
謝辞	24
参考文献	25

1. はじめに

特徴点マッチングは、画像からの形状復元や物体認識など、コンピュータビジョンの様々な分野で重要な役割を担っている。そのため多くの研究が行われており、SIFT[1]や SURF[2]、KAZE[3]、AKAZE[4]など多くの手法が提案されている。これらの手法は、画像の回転、拡大縮小変化に対して特徴点マッチングの結果が影響されにくい等、優れた特徴がある。しかし、未だマッチング精度には改善の余地がある。特徴点マッチングの精度向上が達成出来れば、特徴点マッチングを使用する形状復元や物体認識などの精度向上にも繋がる。

一方、近年、機械学習の発展手法である深層学習は、コンピュータビジョン分野で大きな成果を挙げている。また、深層学習を用いた特徴点マッチング手法の研究も行われている[5][6]。

梶原[6]の深層学習を用いた特徴点マッチングでは、特徴点検出は、AKAZE を使用している。まず、特徴点の座標と、平滑化画像から推定されたその点の主な方向(オリエンテーション)を AKAZE により求める。次に、得られた特徴点座標の周辺を、オリエンテーション分回転させて切り取る。切り取った画像から学習済みの深層学習を用いて特徴量を抽出し、特徴点マッチングを行う。この手法ではマッチング精度が AKAZE など他手法を上回っているが、拡大縮小変化に対応していなかった。また、特徴点検出、回転不変などが AKAZE に依存しているため、特徴点検出も含めた精度向上が難しい。

そこで、本研究では、この従来研究[6]を改善し、AKAZE 等に依存することなく特徴点検出を行う。また、回転・拡大縮小不変を実現し、特徴点マッチングの精度を向上させることを目指す。

2. 特徴点マッチング

特徴点マッチングとは、同じ物体が写る 2 枚の画像間で、同じ物体上の同じ点を対応付けることである。特徴点マッチングは、特徴点検出、特徴量記述、マッチングの 3 段階で行われる。特徴点検出では、画像中から角や線の交わり等の他と異なる固有の点(特徴点)の座標を検出する。特徴量記述では、検出した特徴点の固有性を、ベクトルやバイナリコード表現した値(特徴量)で算出する。マッチングでは、対応付ける特徴点の特徴量を比較し、類似度が高い特徴点同士を対応付ける。2 画像間で対応付けた点が、同じ物体上の同じ点であることが重要である。AKAZE や SIFT では特徴点検出、特徴量記述を行っている。

AKAZE では、特徴点検出は非線形スケール空間の特徴を検出することで行う。特徴量記述は、パッチに分割されたグリッドを用いて特徴量を算出する。

3. 回転・拡大縮小不変

本研究でいう回転・拡大縮小不変とは、特徴点マッチングを行う 2 枚の画像間で、対象物体が異なる大きさや角度で写っていても、特徴点マッチングに影響しない性質のことである。同じ物体が写っている 2 枚の画像があるとき、必ずしも同じ場所、角度で撮影されていない。そのため同じ物体でも画像によって大きさや角度が異なる。そのような変化が、特徴点マッチングに影響しないようにするために多くの研究がされている。

AKAZE では、回転不変は特徴量記述で、特徴点検出時に求められたオリエンテーションによる向きの正規化を行い、実現している。拡大縮小不変は、特徴点検出時にスケールスペースを構築、検出し、非線形スケール空間から勾配と強度の情報を活用する M-LDB を使用することで実現している。

4. 深層学習を用いた特徴点マッチングにおける回転・拡大縮小不変の実現

本研究では、AKAZE に依存せず、特徴点マッチングを行えるよう従来手法[6]を改善する。図 1 に提案手法の流れを示す。特徴点検出では、画像座標の一定間隔おきの点すべてを特徴点とする。特徴量記述では、学習済みの深層学習モデルを使用し、特徴量を抽出する。この抽出した値を使用し、対応付けを行う。特徴点マッチングでは、画像 1、画像 2 の 2 枚の画像が入力として与えられる。

回転・拡大縮小不変を実現するために、片方の画像(画像 1)の各特徴点について、その点を中心として、数段階に回転、拡大縮小変化させた画像を生成し、それらすべてを用いて、マッチングを行う。他方の画像(画像 2)が画像 1 に対して、回転、拡大縮小されていても、これらを数段階に変化して生成した画像のいずれかに近ければ正しい対応づけが得られると期待できる。これにより回転・拡大縮小不変を実現する。以下、各処理について詳細を述べる。

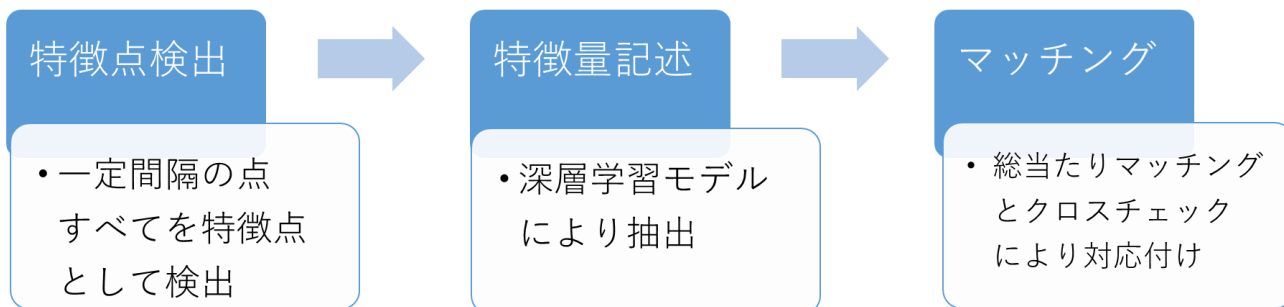


図 1 提案手法の流れ

4.1 特徴点検出

与えられたそれぞれの画像において、画像座標 N 画素おきの点すべてを特徴点とする。

本論文では、 $N=10$ 画素とする。図 2 に $N=10$ 画素で特徴点検出を行った例を示す。

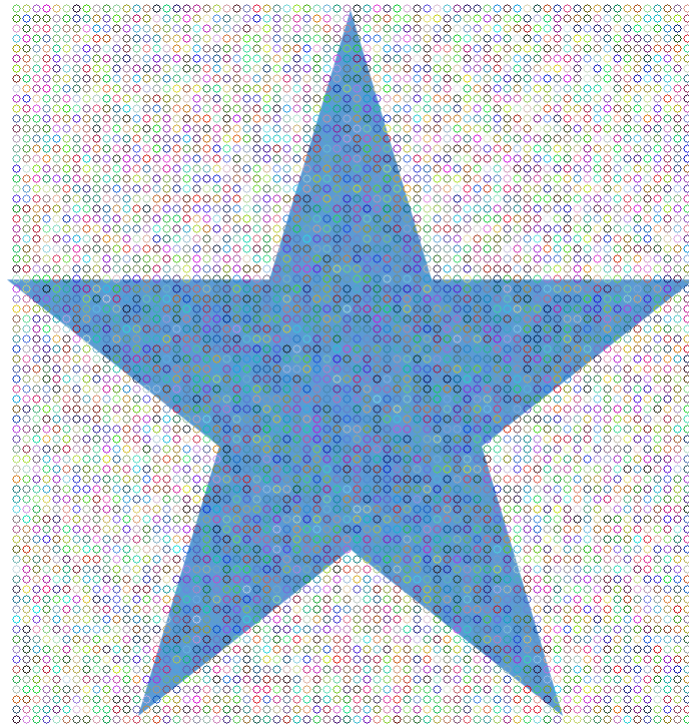


図 2 特徴点検出の例

4.2 特徴量記述

特徴点として検出した点の周辺を、 $K \times K$ 画素の範囲で画像から切り取る。この際、画像の片方(画像 1)は、特徴点を中心として S 通りの拡大縮小変化、 R 通りの回転変化を行い切り取る。そのため特徴点 1 点に対して $S \times R$ 通りの切り取り画像が作成される。これらの切り取った画像から特徴量を算出する。図 3 に切り取り画像の作成例を示す。

特徴量の算出には、AlexNet[7]と呼ばれる構造を持った畳み込みニューラルネットワーク(CNN)に ILSVRC2012 の 1000 カテゴリ、120 万画像のデータを学習させた学習済みの公開モデル(bvlc_reference_caffenet[8])を用いる。このモデルに上記の切り取った画像を与えて得られる fc7 層の出力を特徴量とする。特徴量の次元数は 4096 次元である。

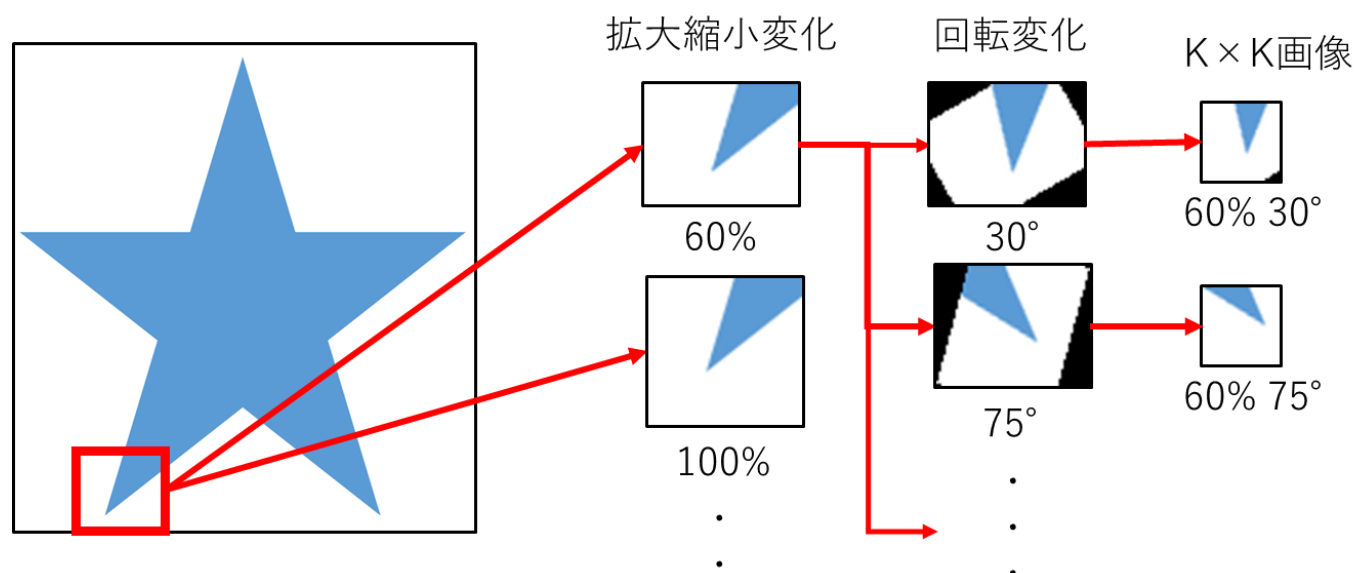


図 3 切り取り画像の作成例

4.3 マッチング

マッチングでは、総当たりマッチングとクロスチェックを組み合わせる。図 4 に簡略化した総当たりマッチングの例を、図 5 にクロスチェックの例を示す。総当たりマッチングは、画像 1 から検出された各特徴点に対して、画像 2 のすべての特徴点と特徴点類似度を計算し、類似度が最も高い特徴点と対応付ける。この際、画像 2 の特徴点が画像 1 の複数の特徴点と対応することがある。クロスチェックでは、画像 1 と画像 2 の立場を入れ替えて同様に総当たりマッチングを行う。その両マッチングで同じ特徴点の組が対応したものののみマッチング結果とする。この 2 つを組み合わせて行うことで総当たりマッチングの問題である 1 つに対して複数個対応付くことがなくなり、より信頼性の高いマッチング結果を得ることが出来る。特徴点を中心として複数通りに回転、拡大縮小変化させた切り取り画像を用いるため、同じ座標で異なる変化を行ったものが複数対応することがある。この場合、最も類似性が高いものを採用する。

特徴点間の類似度はコサイン類似度を使用する。画像 1 の特徴点 i の特徴量を \vec{Q}_i とし、画像 2 の特徴点 j の特徴量を \vec{P}_j と表す。この時、コサイン類似度は式(1)で求まる。この値が 1 に近いほど類似しているとする。

$$\frac{\vec{Q}_i \cdot \vec{P}_j}{|\vec{Q}_i| |\vec{P}_j|} \quad (1)$$

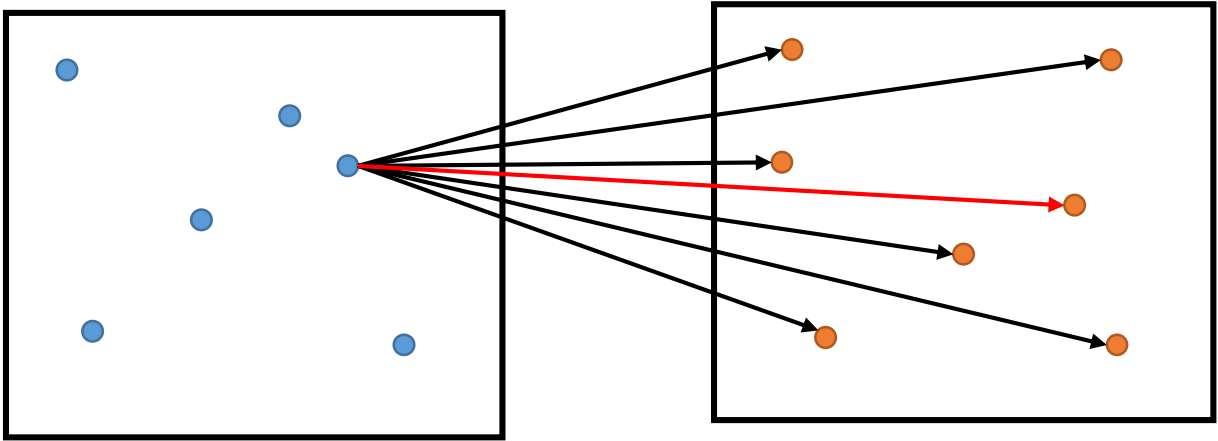


図 4 総当たりマッチング

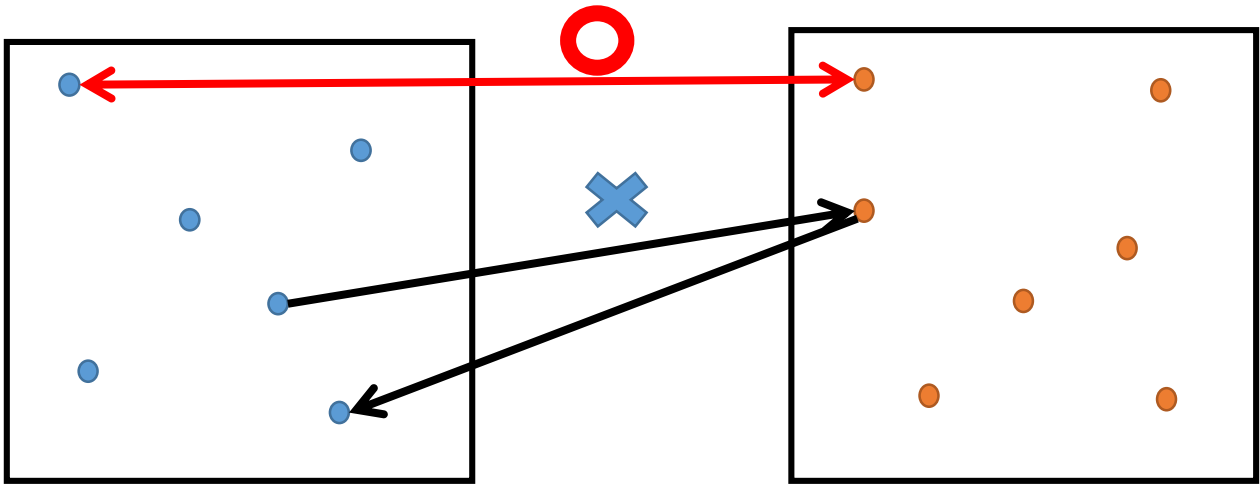


図 5 クロスチェック

5. 実験

5.1 アフィン変換画像を用いた実験

本実験では、回転、拡大縮小変化の度合いに対するマッチング成功率の変化を評価する。

そのために、元画像をアフィン変換により回転、拡大縮小変化させた画像との特徴点マッチングを行う。

5.1.1 実験方法

元画像 1 枚を用意する(画像 1, 図 6)。画像 2 として、元画像を 0° 、 5° 、 15° 、 35° 、 45° 、 90° に回転させ、それぞれ 50%、60%、70%、80%、90%、100%、110%、120%、130%、140%、150%の大きさに変換した 66 通りのアフィン変換で作成した画像を用いる(画像 2, 図 7)。66 通りの画像に提案手法を適用し、特徴点マッチングの結果を得る。

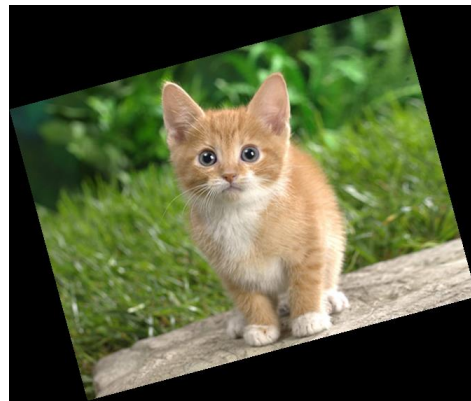
特徴量記述において、特徴点の周辺の切り取り範囲は、 $K = 32$ 画素とする。また、画像 1 の特徴量記述では、60%、80%、100%、120%、140%の $S = 5$ 通りの拡大縮小変化、 0° 、 15° 、 30° 、 45° 、 60° 、 75° 、 90° の $R = 7$ 通りの回転変化を行う。図 8 に画像 1 の同じ特徴点を中心とした切り取り画像の一部を示す。上の段が 0° の回転変化画像を、下の段が 90° の回転変化を示す。左から 60%、80%、100%、120%、140%の拡大縮小変化したものを示している。



図 6 元画像[9](480×360 画素)



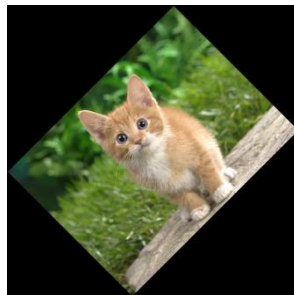
(a) 5° 130%



(b) 15° 120%



(c) 35° 50%



(d) 45° 70%



(e) 90° 80%

図 7 アフィン変換画像



図 8 画像 1 の切り取り画像の一部

マッチング成功率は式(2)で求める。

$$\text{マッチング成功率} = \frac{\text{正しい対応点の数}}{\text{得られた対応点の数}} \quad (2)$$

マッチングに使用する画像 1 の i 番目の特徴点座標 Q_i と画像 2 の j 番目の特徴点座標 P_j が対応したとする。画像 2 はアフィン変換で作成しているため、 P_j に対して逆変換を行うことで、正解座標 P'_j を算出できる。この時、許容座標誤差 T に対して $|Q_i - P'_j| < T$ を満たす対応点を正しい対応点とする。本実験では許容座標誤差 $T = 10$ 画素とする。

5.1.2 実験結果

表 1 に提案手法のマッチング成功率、表 2 に AKAZE のマッチング成功率を示す。提案手法では、画像 1 は特徴点を中心として回転・拡大縮小変化させ、切り取る。画像 2 として、その回転変化に含まれる 15° に回転させた画像を用いた場合のマッチング成功率を図 9 に、画像 1 の回転変化に含まれない 35° に回転させた画像を用いた場合のマッチング成功率を図 10 に示す。また、AKAZE は 0° 、 90° のマッチング成功率が特に高いため、 90° に回転させた画像を用いた場合のマッチング成功率を図 11 に示す。

提案手法のマッチング成功率は全ての場合で 90% を上回る結果となった。AKAZE は 0° 、 90° のマッチング成功率が特に高いこともあり、 0° 、 90° の結果は提案手法が下回ることもあるが、その他は全て AKAZE を上回る結果であった。この結果から提案手法は回転・拡大縮小不変を実現出来た。

表 1 提案手法のマッチング成功率(%)

角度 スケール	0°	5°	15°	35°	45°	90°
50%	98.4	94.6	94.6	92.3	93.2	97.5
60%	97.7	96.9	96.9	96.7	97.2	95.3
70%	93.2	96.4	96.4	96.8	96.8	92.9
80%	98.3	97.7	97.7	97.6	96.3	97.3
90%	95.7	97.0	97.0	97.3	97.1	96.6
100%	100.0	98.1	98.1	98.4	96.2	97.3
110%	96.7	97.3	97.3	96.4	96.4	95.9
120%	97.1	97.3	97.3	97.9	96.9	97.1
130%	97.2	96.9	96.9	97.0	96.7	95.8
140%	97.1	96.2	96.2	96.2	98.7	96.9
150%	94.3	95.0	95.0	93.9	93.5	93.6

表 2 AKAZE のマッチング成功率(%)

角度 スケール	0°	5°	15°	35°	45°	90°
50%	93.3	80.5	54.5	46.9	45.7	95.1
60%	90.4	75.0	57.4	52.7	51.8	88.6
70%	93.3	80.8	67.7	66.8	66.4	92.2
80%	90.8	81.6	69.5	67.0	69.7	91.1
90%	96.4	90.9	84.1	84.4	86.3	96.5
100%	100.0	93.0	94.4	94.2	93.7	100.0
110%	97.0	84.6	90.6	88.9	87.9	96.9
120%	88.3	90.3	79.9	78.8	80.3	87.1
130%	89.9	93.0	89.4	86.9	85.4	89.9
140%	94.2	91.8	89.9	91.0	88.2	93.9
150%	93.8	54.5	87.7	88.4	87.6	91.0

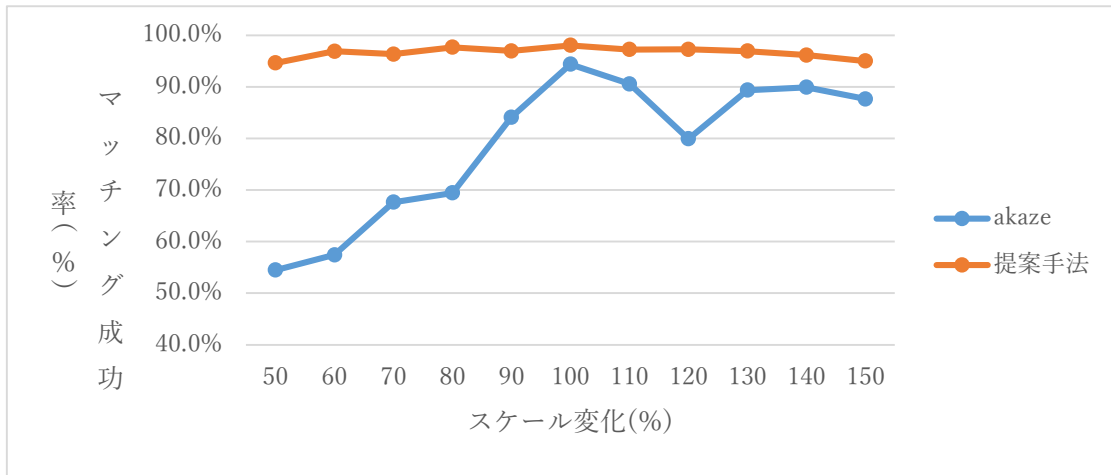


図 9 15° 回転画像のマッチング成功率

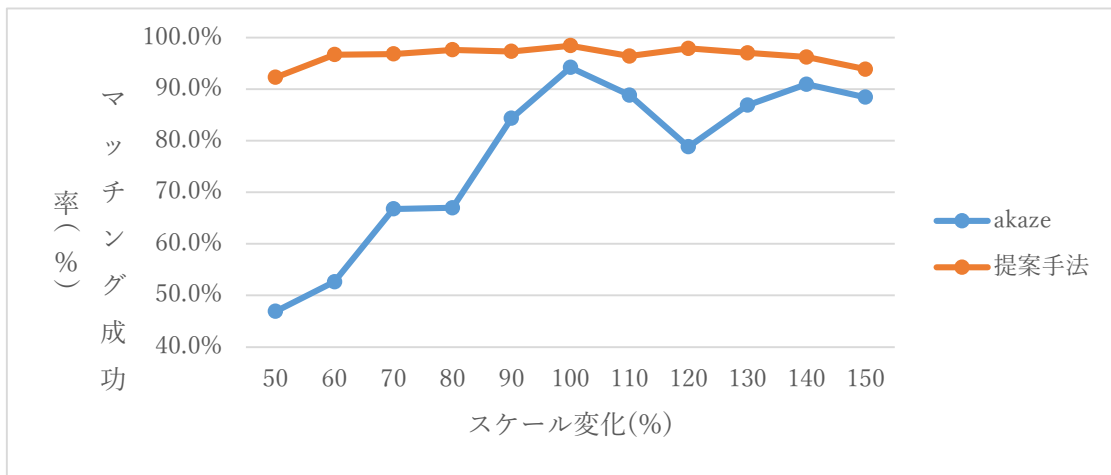


図 10 35° 回転画像のマッチング成功率

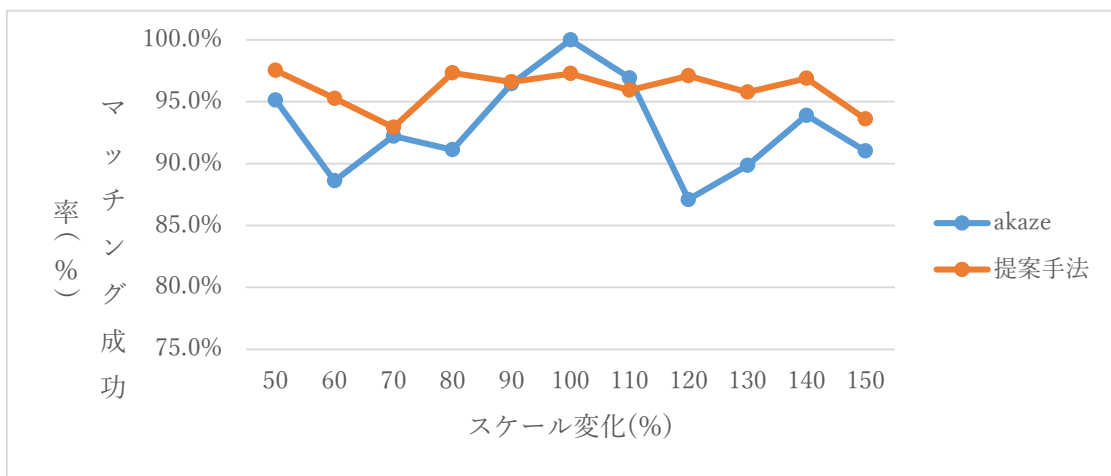


図 11 90° 回転画像のマッチング成功率

5.1.3 処理時間

表 3 に AKAZE と提案手法の処理時間と特徴点、対応点数を示す。また、表 4 に提案手法での処理時間の内訳を示す。

提案手法は多くの特徴点を検出し、その中から適切な対応付けを求める方式であるため、特徴点数が多い。特徴点マッチングの結果として得られた対応点数では、AKAZE は 390 点、提案手法は 1391 点と AKAZE の対応点の 3.5 倍の多くの対応点を検出できた。

一方、処理時間については、AKAZE は 0.579s であるのに対して、提案手法では 10494s を必要としている。比較対象である AKAZE は、KAZE を高速化したものであるため高速に特徴点検出、特徴量記述を行うことができる。提案手法が多くの時間を必要とする大きな理由は、特徴点数が多いためである。特徴量記述では、画像 1 の特徴点 1 つに対して、回転、拡大縮小変化を行い、切り取りを行うため、多くの時間が必要となる。また、マッチングでは、(画像 1 の特徴点数)×(画像 2 の特徴点数)の計算量がかかるため、特徴点数が増えると時間がかかる。

表 3 処理時間,対応点(回転 90° 拡大縮小 150%)

	時間(s)	特徴点数	対応点数
AKAZE	0.579	画像 1:422 画像 2:637	390
提案手法	10494	画像 1:1485×35 画像 2 :3519	1391

表 4 提案手法の処理時間内訳(s)

	特徴量記述	マッチング
画像 1	9633	247
画像 2	614	

5.1.4 深層学習による特徴量記述の有効性

提案手法のうち、深層学習による特徴量記述の部分のみについての有効性を評価するために、実験を行う。

本実験では、特徴点検出と回転、拡大縮小への対処には AKAZE を使用し、特徴量記述のみ深層学習を利用する。

まず、特徴点の座標と、平滑化画像から推定されたその点の主な方向(オリエンテーション)、最適な拡大率(スケール)を AKAZE により求める。次に、得られた特徴点座標の周辺を、オリエンテーション分回転、スケール分拡大縮小させて切り取る。これにより回転、拡大縮小に対処する。切り取った画像から学習済みの深層学習を用いて特徴量を抽出し、特徴点マッチングを行う。この手法を以降、手法 2 と呼ぶ。切り取り範囲は 5.1.1 節と同様 $K = 32$ 画素とする。また、使用する画像は 5.1.1 節と同じ画像である。

手法 2 のマッチング成功率を表 5 に、処理時間と対応点数を表 6 に、手法 2 の処理時間内訳を表 7 に示す。マッチング成功率は、手法 2 がすべて AKAZE を上回る結果となった。この結果から深層学習を用いた特徴量記述が有用であるといえる。また、AKAZE で検出するオリエンテーションとスケールは有用性が高いことがわかる。

一方、手法 2 は、特徴点検出を AKAZE で行っているため、特徴点数は AKAZE と同じである。特徴点マッチング結果として得られた対応点数では、AKAZE の対応点の 0.9 倍ほど少なくなっている。また、処理時間は手法 2 の方が多くの時間を必要とする。深層学習を用いた特徴量記述や総当たりマッチングに時間が必要となるためである。

表 5 手法 2 のマッチング成功率(%)

角度 スケール	0°	5°	15°	35°	45°	90°
50%	98.0	95.2	90.3	88.9	93.2	98.9
60%	98.3	96.8	95.4	95.4	93.0	97.2
70%	98.8	96.8	96.7	96.7	95.3	97.6
80%	97.4	97.6	96.7	96.2	95.8	95.9
90%	99.7	99.1	99.1	99.1	100.0	100.0
100%	100.0	99.5	99.2	99.5	99.7	100.0
110%	99.7	98.9	99.2	99.7	99.4	99.4
120%	97.5	97.3	96.1	97.0	95.7	95.3
130%	97.5	98.5	98.3	98.3	98.5	97.5
140%	99.7	99.2	98.3	99.2	98.3	99.4
150%	98.9	98.1	98.9	99.2	98.9	98.0

表 6 処理時間,対応点(回転 90° 拡大縮小 150%)

	時間(s)	特徴点数	対応点数
AKAZE	0.579	画像 1:422 画像 2:637	390
手法 2	794	画像 1:422 画像 2:637	355

表 7 処理時間内訳(s)

対応画像	特徴量記述	マッチング
画像 1	341	1.6
画像 2	451	

5.2 実画像実験

本実験は実画像での提案手法の有効性を評価する。

5.2.1 実験条件

1つの物体を移動しながら写している動画から、離れた3フレームを抜き出した画像を使用する(図 12)。

使用したデータ[10]は、画像と奥行き情報が対応づいたRGB-Dデータとして提供されている。そこから得られる3次元情報を利用して、画像2の各点について、画像1の正解座標が計算できる。本実験では、正解座標と画像1の特徴点座標との許容座標誤差が $T=30$ 画素以下のものを正しい対応点とする。



(a) フレーム 1



(b) フレーム 2



(c) フレーム 3

図 12 実画像実験の使用画像[10](640x480 画素)

5.2.2 実験結果

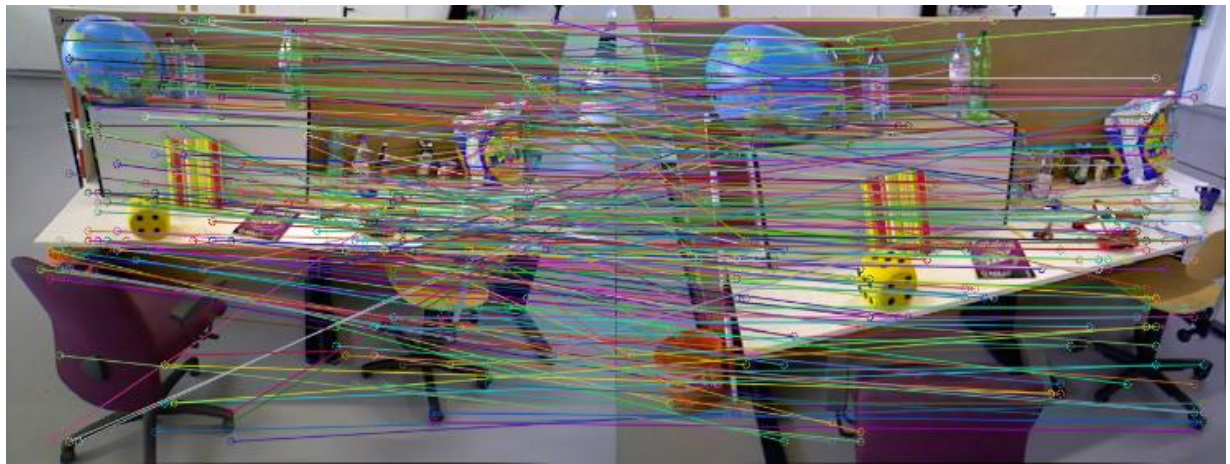
提案手法と AKAZE での実験結果を表 8 に示す。実画像を使用した場合でも、提案手法のマッチング成功率が AKAZE を上回る結果となった。フレーム間が離れていると見え方が大きく変化するので、フレーム 1 とフレーム 3 のマッチングが最も低い成功率と予想していたが、フレーム 2 とフレーム 3 のマッチングが最も低い成功率となった。図 13 は、対応した座標の中で、間違いと判定された対応座標を示したものである。同じ色で繋いでいる点は同士がマッチング結果を表している。間違いと判定されたマッチング結果の特徴は、床やボードなどの特徴が出難い箇所が多い点である。提案手法は、特徴点を絞って検出しているのではなく、座標を 10 刻みで特徴点としているため、他と区別しにくい点も特徴点として検出され、間違った対応付けが得られている。

表 8 実画像でのマッチング成功率(%)

画像 1	画像 2	AKAZE	提案手法
フレーム 1	フレーム 2	69.2	75.5
フレーム 1	フレーム 3	25.4	42.5
フレーム 2	フレーム 3	21.6	25.9



(a) フレーム1と2のマッチング



(b) フレーム1と3のマッチング



(c) フレーム2と3のマッチング

図 13 間違いと判定された対応座標

6. おわりに

本研究では、従来研究[6]を改善し、AKAZE に依存することなく特徴点検出を行った。また、回転・拡大縮小不変を実現し、特徴点マッチングの精度を向上させた。アフィン変換で回転、拡大縮小を人工的に与えた画像を用いた実験では、提案手法はほとんどの場合で、マッチング成功率で AKAZE の値を上回ることが出来た。実画像を使用した実験でも、AKAZE のマッチング率を上回ったことから有効性を示すことができた。また、深層学習での特徴量記述の有用性も示すことができた。

今後の課題として、処理時間を削減することが挙げられる。提案手法の処理時間は、AKAZE に比べて非常に長い。AKAZE では、他と区別できる固有性のある点を特徴点として検出している。しかし、提案手法では、一定間隔の点をすべて特徴点としているため、特徴点数が多くなり、特徴量抽出、マッチング共に時間がかかる。特徴量抽出後に、特徴点間で似た値を削除して、固有性のある特徴点に絞り込むことで、マッチング時の処理時間を短くすることができると思われる。また、提案手法では、回転、拡大縮小変化を行い、多くのパターンで特徴量抽出を行っている。深層学習のモデルを回転や拡大縮小を考慮できるように学習したものを使用することで、特徴量抽出の処理時間も削減できると考えられる。

また、精度改善も必要である。アフィン変換画像を用いた実験では高いマッチング成功率を示すことが出来た。しかし、実画像を使用した実験では AKAZE を上回る結果となったものの、画像によっては 25%程度のマッチング成功率だった。実画像に対してマッチング成功率をさらに向上させる必要がある。

謝辞

本論文の作成にあたり、丁寧で熱心なご指導を頂いた卒業論文指導教員の椋木雅之教授に深く感謝致します。指導教員の椋木雅之教授には、実験を行う際のアドバイスや助言、ご指導をたくさん頂きました。また、椋木研究室の皆様には、日々の雑談から研究に関する助言まで快く受けて頂き、感謝致します。最後に、あたたかい励ましをいつも送り続け、支えてくれた友人、家族に心から感謝致します。

参考文献

- [1] D. G. Lowe, “Distinctive Image Features from Scale-Invariant Keypoints”, IJCV, (2004), pp.91-110.
- [2] H. Bay, T. Tuytelaars, L. V. Gool, “SURF: Speeded Up Robust Features”, CVPR, (2008), pp.346-359.
- [3] P. F. Alcantarilla, A. Bartoli, and A. J. Davison. “KAZE features”, ECCV, (2012), pp.214–227.
- [4] P. F. Alcantarilla, J. Nuevo, A. Bartoli, “Fast Explicit Diffusion for Accelerated Features in Nonlinear Scale Spaces”, BMVC, (2013), pp.13.1-13.11.
- [5] Y. Ono, E. Trulls, P. Fua, K. M. Yi, “LF-Net: Learning Local Features from Images”, (2018), arXiv:1805.09662v2[cs.CV].
- [6] 梶原奏美, “深層学習で抽出した特徴量を用いた特徴点マッチング”, H30 年度宮崎大学工学部情報システム工学科卒業論文, (2019).
- [7] A. Krizhevsky, I. Sutskever, G. E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks”, NIPS, (2012), pp.1091-1105.
- [8] AlexNet 学習済みモデル ,
https://github.com/BVLC/caffe/tree/master/models/bvlc_reference_caffenet
- [9] caffe サンプル画像 ,
<https://kyamagu.github.io/dl-workshop-2016/2-classification>
- [10] 使用した RGB-D 画像 ,
<https://vision.in.tum.de/data/datasets/rgbd-dataset>