

令和3年度卒業論文
U-Net を用いた AKAZE 特徴点の選別による
マッチング精度の向上

宮崎大学 工学部 情報システム工学科

山元望

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	3
2.1 AKAZE を用いた特徴点マッチング	3
2.2 SuperPoint を用いた特徴点マッチング	4
2.3 梶原[6]の特徴点マッチング	4
2.4 マッチング精度改善へのアプローチ	5
2.5 U-Net の利用	6
3. U-Net を用いた AKAZE 特徴点の選別	7
3.1 学習段階	7
3.2 選別段階	9
4. 実験	11
4.1 元画像・変換画像ともに AKAZE 特徴点を選別する実験	11
4.1.1 実験方法	11
4.1.2 実験結果	13
4.2 元画像のみ AKAZE 特徴点を選別する実験	15
4.2.1 実験方法	15
4.2.2 実験結果	15
5. おわりに	18
謝辞	19
参考文献	20

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、優れた特徴があるが、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がコンピュータビジョン分野で大きな成果を挙げており、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]。その1つに梶原[6]の深層学習を用いた特徴点マッチングがある。梶原[6]の研究では、特徴点検出はAKAZEを使用し、特徴量記述は学習済みの深層学習モデルを用いて行う。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い、処理速度が遅いなど本質的に改善の難しい問題があった。

そこで、本研究では、AKAZEで検出した特徴点(AKAZE特徴点)の中でマッチングに有効な特徴点を選別することで、マッチング精度の向上を目指すという別のアプローチを提案する。学習用画像に回転・拡大縮小変化を加えてマッチングした際に正しくマッチングできた特徴点のみを学習し、AKAZE特徴点の中から条件に当てはまる特徴点だけを取り出すことで、マッチング精度が向上すると考えられる。AKAZE特徴点の中で回転・拡大縮小変化に強いものを選別することで、回転・拡大縮小変化にも対応できる。また、特徴量記述はAKAZEのものをそのまま使うので、処理時間もそれほど増加しない。

AKAZEは特定の対象に対して検出精度が特化しているわけではないが、提案手法により、例えばAKAZEを道路シーンの解析に適用する場合、道路シーンでの特徴点を絞り込み、精度向上を図るといったAKAZEのチューニングが可能となる。

特徴点の学習と選別にはU-Net[7]を用いる。U-Netは全層畳み込みネットワーク(FCN)の1つで、画像の画素ごとに物体を選別することができる。物体の代わりにAKAZE特徴点を学習することで、どのような点を選別すれば良いか判定する選別器を構築する。

以下、第2章では特徴点マッチングの他手法や提案手法のアプローチ、提案手法に使用するネットワー

クについて述べる。第3章では U-Net を用いた AKAZE 特徴点の選別についてのプロセスを示し、第4章で実装した提案手法の実験について述べる。最後に第5章でまとめとする。

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を特徴量として算出する。マッチングでは、対応付ける画像同士の特徴量を比較し、特徴量の距離が近いものを類似度が高いとし、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。

2.1 AKAZE を用いた特徴点マッチング

AKAZE は、特徴点検出と特徴量記述を行う手法の 1 つである。SIFT や SURF の欠点を改善した KAZE をもとにしているため、SIFT や SURF 等の他手法と比べロバスト性や処理速度の面で優れている。

また、AKAZE は回転・拡大縮小不変である。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。特徴点検出では、特徴点と特徴点のスケールを検出する。その際、スケールを正規化することで拡大縮小不変となる。特徴量記述では、勾配の方向や大きさから向きの正規化を行うことで回転不変となる。特徴量はバイナリコードで表されるため、マッチングでは特徴量間のハミング距離を計算し、対応付けを行う。

2.2 SuperPoint を用いた特徴点マッチング

SuperPoint[8]は、特徴点検出と特徴量記述に教師あり学習を使用する手法である。特徴点検出と特徴量記述の両方に同じエンコーダを使用しているため、他手法と比べ処理速度が速く、マッチング精度も高い。特徴点検出は、いろいろな 3 次元図形の合成画像からその図形の角を特徴点として学習することで特徴点を検出する。特徴量記述は、ホモグラフィ変換された画像から得た特徴点画像からそれぞれのホモグラフィを計算し学習することで特徴量記述を行う。

2.3 梶原[6]の特徴点マッチング

梶原[6]の深層学習を用いた特徴点マッチングは、特徴点検出は AKAZE を使用し、特徴量記述は学習済みの深層学習モデルを用いて特徴量を抽出する。マッチング精度は AKAZE など他手法を上回る結果となっている。学習済みの深層学習モデルは、AlexNet[9]と呼ばれる構造をも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を使用している。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、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り、学習済みの深層学習モデルに与えることで特徴量を抽出し、特徴点マッチングを行う。

2.4 マッチング精度改善へのアプローチ

本研究では、深層学習により AKAZE 特徴点の中からマッチングに有効なもののみ選別することで、マッチング精度の向上を目指す。そのために、学習用画像に回転や拡大縮小等の変化を加え、正しくマッチングできた AKAZE 特徴点を学習させる。学習済みの深層学習モデルを選別器とし、選別器に画像を与えることで、AKAZE 特徴点の中から選別器の条件に当てはまるマッチングに有効な特徴点のみを取り出す。深層学習のネットワーク構造は U-Net を用いる。

提案手法により、例えば AKAZE を道路シーンの解析に適用する場合、回転や拡大縮小変化した道路画像との特徴点マッチングの中で正しくマッチングできた特徴点を学習することで、道路シーンに対して検出精度を特化させることができる。また、道路画像の特徴点を選別することで、道路シーンでの特徴点マッチングで精度向上を図るといった AKAZE のチューニングが可能となる。

梶原[6]の研究は、拡大縮小変化に対応していない、処理時間がかかる等本質的に改善が難しい問題があったが、提案手法では、拡大縮小変化した画像とのマッチングで正しくマッチングできた特徴点を学習させることで、AKAZE 特徴点の中から拡大縮小変化に強いものを選別することができ、拡大縮小変化にも対応できる。特徴量は AKAZE 特徴量をそのまま使うので処理時間もそれほどかからない。

提案手法と SuperPoint は学習により特徴点を検出していること、回転や拡大縮小など変化した画像とのマッチングにより学習データを作成していることなど共通点が多いが、本研究は学習データに AKAZE 特徴点を使用しているため、3次元図形の角を学習している SuperPoint に比べ、より細かい特徴点を検出できる。また、SuperPoint は特徴点を学習したものをそのまま特徴点検出器として利用しているのに対し、提案手法は AKAZE 特徴点を学習したものを選別器として利用するという相違点がある。

2.5 U-Net の利用

U-Net は全層畳み込みネットワーク(FCN)の 1 つであり、画像の中で物体がどこにあるかを推定するためのネットワークである。図 1 に U-Net のネットワーク構造を示す。一般的な FCN では、畳み込み処理を加えていくと、ネットワークが「物体が何であるか」についての特徴を抽出していくが、プーリングの影響で「物体がどこにあるか」についての特徴は失われていってしまい、そのまま逆畳み込みを行っても物体の位置情報は十分に復元できない場合がある。しかし、U-Net は畳み込みを行った後、特徴マップを保持しておいて後で逆畳み込みをする際に足し合わせる skip-connection という仕組みがあるため、より精密に物体の領域を出力できる。

以上の利点から、本研究で特徴点がどこにあるかを学習し、与えられた画像に対してどのような特徴点を選別すれば良いかを推定することに対して U-Net は適している。

使用した U-Net は公開モデル(Pytorch-UNet[10])で、Google Colaboratory で PyTorch を用いて環境構築する。

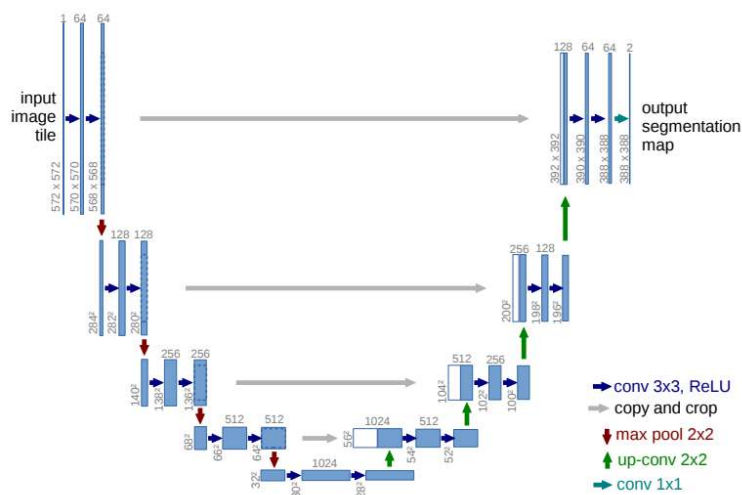


図 1 U-Net のネットワーク構造[7]

3. U-Net を用いた AKAZE 特徴点の選別

提案手法は、学習用画像から抽出した AKAZE 特徴点の中で正しくマッチングが行える特徴点の性質を U-Net に学習させて選別器を構築し、得られた選別器を使って、処理対象の画像から抽出した AKAZE 特徴点を選別する。処理は、選別器を構築するための学習段階と、学習済みの選別器を使った選別段階の大きく 2 つに分けられる。以下、各処理について詳細を述べる。

3.1 学習段階

学習段階では、まず学習用元画像(図 2)を用意する。本研究では、車載カメラから前方の道路シーンを撮影したデータセットである Cityscaps Dataset[11]から 4967 枚の画像を 512×256 画素の大きさに縮小して用いる。次に、学習用元画像に対して、アフィン変換で回転や拡大縮小させた学習用変換画像(図 3)を作成する。この学習用元画像と学習用変換画像に対して、AKAZE を適用して特徴点検出、特徴量記述及びマッチングを行う。

マッチングには、総当たりマッチングとクロスチェックを組み合わせた手法を用いる。マッチングに使用する画像 2 枚をそれぞれ画像 1、画像 2 とする。総当たりマッチングでは、画像 1 から抽出したすべての特徴点について画像 2 中で最も特徴点間の距離が小さ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してしまうことがある。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。双方の結果で特徴点の組が一致するもののみをマッチング結果とすることでこの問題を解消し、信頼性の高いマッチング結果を得る。

学習用変換画像は、適用した変換が既知であるため、画像内の各点は逆変換により元画像での位置を計算できる。これにより、得られたマッチング結果が正しいか判定できる。各学習用元画像において、マッチングが正しく行えた AKAZE 特徴点(選別特徴点)の座標を求める。

次に、求めた選別特徴点を抽出できるように、U-Net を学習する。学習用元画像に対して、同じ大きさの画像を用意する。その画像内の各選別特徴点座標を中心に、 7×7 画素の範囲を塗りつぶした学習用特徴点画像(図 4)を作成する。この学習用元画像と学習用特徴点画像の組が、U-Net への学習データとなる。

この学習データを使って U-Net を学習し、元画像が入力されるとその中で選別特徴点と同様の性質を持つ点の周辺を塗りつぶした画像を出力する選別器を構築する。U-Net の学習は、5 エポック行う。



図 2 学習用元画像の例



図 3 学習用変換画像の例(30° 回転画像)

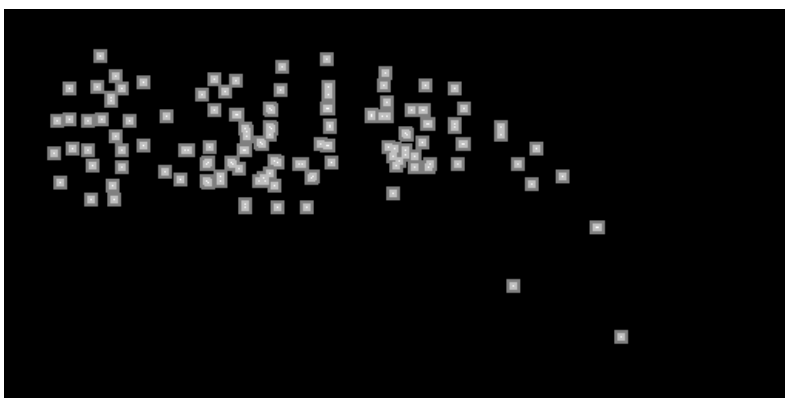


図 4 学習用特徴点画像(図 2、図 3 から作成)

3.2 選別段階

選別段階では、学習させた U-Net を選別器として用いる。処理対象となる元画像(図 5)に対して、AKAZE により特徴点(候補特徴点)を検出する(図 6)。同じ画像を選別器にも与える。選別器からの出力画像に非極大値の抑制を行った上で、画素値が閾値=0.5 以上となる点の座標(選別座標)を抽出する(図 7)。候補特徴点のうち、選別座標と一定距離(本研究では距離 5)以内にあるもののみを選別された特徴点(選別特徴点)とする(図 8)。

特徴点マッチングでは、マッチング対象となる 2 枚の画像それぞれに対して選別処理を適用し、得られた選別特徴点同士でマッチングを行う。マッチングには、総当たりマッチングとクロスチェックを用いる。これにより、AKAZE 特徴点のうち、回転・拡大縮小等の変化があっても正しくマッチングできる点を選別し、マッチング精度を向上させる。



図 5 元画像



図 6 元画像の AKAZE 特徴点

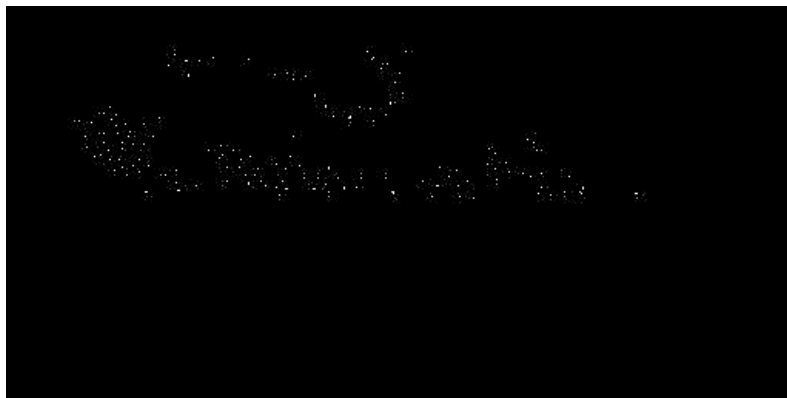


図 7 U-Net の出力画像

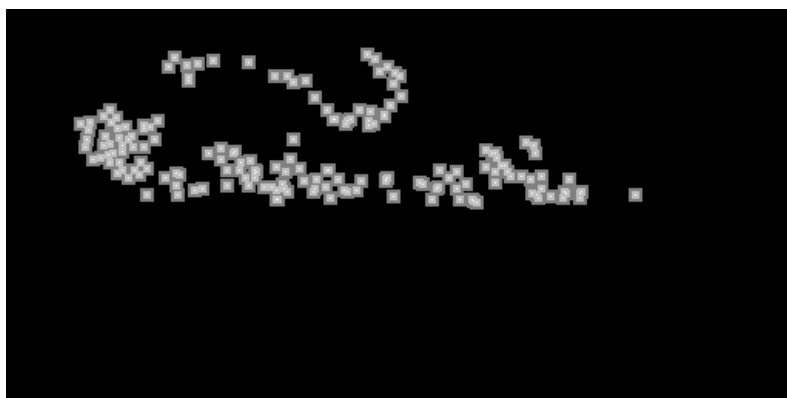


図 8 元画像の選別特徴点

4. 実験

本実験では、提案手法を用いた特徴点マッチングの回転・拡大縮小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、アフィン変換により回転・拡大縮小変換させた画像との特徴点マッチングを行う。

4.1 元画像・変換画像ともに AKAZE 特徴点を選別する実験

テスト用元画像とテスト用変換画像に提案手法を適用し評価実験を行う。

4.1.1 実験方法

テスト用元画像 15 枚を用意する。学習データにも使用した Cityscaps Dataset[11]から学習用元画像に使用していない画像を 15 枚選び、 512×256 画素の大きさに縮小して用いる。次に、テスト用元画像に対して、アフィン変換で回転や拡大縮小変換させたテスト用変換画像を作成する。テスト用元画像を 10° 、 30° 、 45° 、 90° に回転させ、それぞれ 50%、80%、100%、120%、150%の大きさに変換した 20 通りをテスト用変換画像とする。図 8～図 11 にテスト用元画像とテスト用変換画像の例を示す。テスト用元画像とテスト用変換画像ともに提案手法を適用し、特徴点マッチングの結果を得る。

使用する学習済み U-Net モデル(選別器)は 30° 回転、 45° 回転、80%縮小、120%拡大のそれぞれの学習用変換画像に対し正しくマッチングできた AKAZE 特徴点を学習させたもので、実験にはこの 4 通りそれぞれについて 5 エポック学習した 4 つの選別器を用いる。

マッチング成功率は式(1)で求める。

$$\text{マッチング成功率} = \frac{\text{正しい対応点の数}}{\text{得られた対応点の数}} \quad (1)$$

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。



図 8 テスト用元画像



図 9 10° 50%



図 10 30° 80%



図 11 90° 150%

4.1.2 実験結果

表 1 に AKAZE のマッチング成功率を示す。また、表 2～表 5 に学習済みの選別器で選別した特徴点のマッチング成功率を示す。表 2 は 30° 回転、表 3 は 45° 回転、表 4 は 80%縮小、表 5 は 120%拡大を学習した選別器での結果である。AKAZE よりもマッチング成功率が上回ったところを赤字で示す。

提案手法のマッチング成功率は、30° 回転を学習した選別器を使用した場合(表 2)では向上幅は小さいが一定の効果が見られ、20 通りの変換の組み合わせ結果のうち 14 通り(14/20)でマッチング成功率が AKAZE を上回った。45° 回転を学習した選別器を使用した場合(表 3)は、45° 付近で効果が見られ、10/20 でマッチング成功率が AKAZE を上回った。80%縮小を学習した選別器を使用した場合(表 4)は、縮小を学習したにもかかわらず、拡大側で向上した。6/20 でマッチング成功率が AKAZE を上回り、あまり効果は見られなかった。120%拡大を学習した選別器を使用した場合(表 5)は、拡大側の回転の少ない範囲で向上した。5/20 でマッチング成功率が AKAZE を上回り、あまり効果は見られなかった。

本実験では、提案手法は回転に対してはマッチング精度向上の効果が見られたが、拡大縮小に対しては効果が低かった。

表 1 AKAZE のマッチング成功率(%)

角度 スケール	10°	30°	45°	90°
50%	50.24	51.13	53.16	59.55
80%	91.26	88.20	85.37	83.52
100%	98.89	94.51	91.42	88.11
120%	90.47	86.33	82.30	75.76
150%	67.74	67.89	66.23	57.62

表2 30° 学習時のマッチング成功率(%)

角度 スケール	10°	30°	45°	90°
50%	49.42	50.83	49.03	54.59
80%	91.00	90.39	87.36	83.72
100%	99.05	97.13	93.73	88.49
120%	93.88	90.37	88.35	77.20
150%	70.29	70.57	70.36	56.77

表3 45° 学習時のマッチング成功率(%)

角度 スケール	10°	30°	45°	90°
50%	43.47	43.01	41.59	46.35
80%	88.98	86.50	83.47	83.56
100%	98.00	95.25	91.93	89.33
120%	92.11	89.52	87.18	79.88
150%	66.43	71.19	68.48	55.47

表4 80%学習時のマッチング成功率(%)

角度 スケール	10°	30°	45°	90°
50%	46.02	48.19	47.73	55.68
80%	88.65	86.90	82.41	81.99
100%	98.72	94.41	88.10	85.81
120%	92.86	86.86	82.66	73.71
150%	69.54	69.07	68.32	55.92

表5 120%学習時のマッチング成功率(%)

角度 スケール	10°	30°	45°	90°
50%	45.00	47.51	49.14	54.68
80%	88.85	86.53	81.52	77.85
100%	98.68	93.79	88.84	83.19
120%	91.77	87.40	82.75	70.09
150%	70.40	68.23	64.53	51.09

4.2 元画像のみ AKAZE 特徴点を選別する実験

4.1 節では、テスト用元画像とテスト用変換画像どちらも AKAZE 特徴点を選別した。この場合、回転に対してはマッチング精度向上の一定の効果が見られたが、拡大縮小に対しては効果が低かった。U-Net による選別器の学習では、元画像のみを学習データとして使っており、変換画像は使っていなかった。そのため、回転・拡大縮小等の変化を加えた画像に対しては、適切な選別座標を出力できない可能性がある。そこで、追加実験としてテスト用元画像のみに提案手法を適用し、変換画像には選別器を適用しない場合の評価実験を行う。

4.2.1 実験方法

元画像のみに提案手法を適用し、テスト用変換画像は AKAZE 特徴点すべてを用いてマッチングを行う。実験方法や実験に使用するテスト用画像、選別器は 4.1 節と同じである。

4.2.2 実験結果

表 6～表 9 に学習済みの選別器で選別した特徴点のマッチング成功率を示す。表 6 は 30° 回転、表 7 は 45° 回転、表 8 は 80%縮小、表 9 は 120%拡大を学習した選別器の結果である。AKAZE よりもマッチング成功率が上回ったところを赤字で示す。

提案手法のマッチング成功率は、30° 回転を学習した選別器を使用した場合(表 6)では 30° 付近で効果が見られた。20 通りの変換の組み合わせ結果のうち 10 通り(10/20)でマッチング成功率が AKAZE を上回り、4.1 節より結果は低下した。45° 回転を学習した選別器を使用した場合(表 7)は、45° 付近で効果が見られた。12/20 でマッチング成功率が AKAZE を上回り、4.1 節より結果は向上した。80%縮小を学習した選別器を使用した場合(表 8)は、縮小側で効果が見られた。8/20 でマッチング成功率が AKAZE を上回り、4.1 節より結果は向上した。120%拡大を学習した選別器を使用した場合(表 9)は、3/20 でマッチング成功率が AKAZE を上回った。拡大側の回転の少ない範囲で向上したものもあるが、あまり効果は見られず、4.1 節より結果は低下した。

本実験でも回転に対してはマッチング精度向上の一定の効果が見られたが、拡大縮小に対しては効果が低かった。また、使用した4つの選別器のうち2つの選別器で4.1節より結果が向上し、残り2つで低下した。結果が向上したものに関しても向上幅はそれほど大きくなく、わずかな差であった。

以上の結果から、テスト用元画像とテスト用変換画像のどちらにも提案手法を適用する場合と、テスト用元画像のみに提案手法を適用する場合の差はあまりなく、U-Netによる選別器の学習で変換画像を学習データとして使っていない問題はそれほど大きくないことが分かった。また、4.1節と本実験ともに、提案手法は回転に対してはマッチング精度向上の一定の効果が見られたが、拡大縮小に対しては効果が低いことも分かった。選別器の学習時に今回学習した回転角度や拡大縮小変化の大きさとは違う変換画像とのマッチング正解特徴点を学習させ、その選別器で評価実験を行うことで、提案手法の拡大縮小に対しての効果の低さを確認し改善すること、また回転・拡大縮小ともに、よりマッチング精度の向上幅を大きくすることが今後の課題である。

表 6 30° 学習時のマッチング成功率(%)

角度 スケール	10°	30°	45°	90°
50%	46.49	47.53	47.26	55.23
80%	89.91	90.60	89.11	83.91
100%	98.31	96.54	92.66	86.37
120%	89.45	87.72	84.33	72.76
150%	69.35	69.21	66.83	56.02

表 7 45° 学習時のマッチング成功率(%)

角度 スケール	10°	30°	45°	90°
50%	42.58	43.45	43.94	50.72
80%	87.41	86.85	86.19	83.80
100%	98.19	95.79	95.27	88.69
120%	91.00	88.86	87.64	78.38
150%	73.06	74.64	72.38	61.15

表 8 80%学習時のマッチング成功率(%)

角度 スケール	10°	30°	45°	90°
50%	53.43	54.04	56.40	63.48
80%	93.50	91.37	87.68	85.20
100%	97.84	92.13	86.40	82.28
120%	85.30	80.21	75.83	66.55
150%	65.02	64.35	60.51	49.32

表 9 120%学習時のマッチング成功率(%)

角度 スケール	10°	30°	45°	90°
50%	45.35	46.32	47.17	54.32
80%	86.72	86.32	84.08	82.76
100%	98.11	92.46	89.08	84.33
120%	92.36	85.62	79.87	70.72
150%	75.97	70.47	63.79	51.93

5. おわりに

本研究では、学習用元画像と学習用元画像を回転・拡大縮小変化した学習用変換画像で正しくマッチングできた学習用特徴点を U-Net を用いて学習させ、学習済みの U-Net を選別器として用いてマッチングに有効な特徴点を選別することで、AKAZE よりもマッチング精度を向上させることを目指した。

テスト用元画像とテスト用変換画像どちらにも AKAZE 特徴点の選別を行った場合と、テスト用元画像に対してのみ AKAZE 特徴点を選別した場合で評価実験を行った。実験結果は、マッチング精度の向上・低下がまちまちで、AKAZE を上回った場合でもマッチング成功率の向上幅の差もあまりなかった。このことから、U-Net による選別器の学習で、元画像のみを学習データとして使っており、変換画像は使っていないという問題はそれほど大きくないということが分かった。また、提案手法は回転に対してはマッチング精度向上の一定の効果が見られるのに対し、拡大縮小に対しては効果が低いことも分かった。

今後の課題としては、選別器の学習時に今回学習した回転角度や拡大縮小変化の大きさは違う変換画像とのマッチング正解特徴点を学習させ、その選別器で評価実験を行うことで、提案手法の拡大縮小に対しての効果の低さを確認し改善すること、また回転・拡大縮小ともに、よりマッチング精度の向上幅を大きくすることが挙げられる。さらに、学習済みの深層学習モデルを選別器ではなく特徴点検出器として利用できるようにすることも今後の課題である。

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棕木雅之教授に深く感謝致します。指導教員の棕木雅之教授には、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、研究の相談や助言を快く受けて頂いた棕木研究室の皆様感謝します。

参考文献

- [1] David G. Lowe, “Distinctive Image Features from Scale-Invariant Keypoints”, IJCV, (2004), pp.91-110.
- [2] Herbert Bay, Tinne Tuytelaars, Luc Van Gool, “SURF: Speeded Up Robust Features”, CVPR, (2008), pp.346-359.
- [3] Pablo F. Alcantarilla, Adrien Bartoli, and Andrew J. Davison. “KAZE features”, ECCV, (2012), pp.214-227.
- [4] Pablo F. Alcantarilla, Jesús Nuevo, Adrien Bartoli, “Fast Explicit Diffusion for Accelerated Features in Nonlinear Scale Spaces”, BMVC, (2013), pp.13.1-13.11.
- [5] Yuki Ono, Eduard Trulls, Pascal Fua, Kwang Moo 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] Olaf Ronneberger, Philipp Fischer, and Thomas Brox, “U-Net: Convolutional Networks for Biomedical Image Segmentation”, International Conference on Medical image computing and computer-assisted intervention, Springer, (2015), pp.234-241.
- [8] Daniel DeTone, Tomasz Malisiewicz, Andrew Rabinovich, “SuperPoint: Self-Supervised Interest Point Detection and Description”, CVPR, (2018), pp.337-349.
- [9] Alex Krizhevsky, Ilya Sutskever, Geoffrey E. Hinton, “ImageNet Classification with Deep Convolutional Neural Networks”, NIPS, (2012), pp.1091-1105.
- [10] Pytorch-UNet
<https://github.com/milesial/Pytorch-UNet>
- [11] Cityscaps Dataset
<https://www.cityscapes-dataset.com>