T2R2 東京工業大学リサーチリポジトリ
Tokyo Tech Research Repository

論文 / 著書情報
Article / Book Information

題目(和文)	ロボットを用いた知識転移
Title(English)	Knowledge Transfer on Robots
著者(和文)	木村大毅
Author(English)	Daiki Kimura
出典(和文)	学位:博士（工学）, 学位授与機関:東京工業大学, 報告番号:甲第10006号, 授与年月日:2015年9月25日, 学位の種別:課程博士, 審査員:長谷川　修，新田　克己，渡邊　澄夫，小野　功，石井　秀明
Citation(English)	Degree:, Conferring organization: Tokyo Institute of Technology, Report number:甲第10006号, Conferred date:2015/9/25, Degree Type:Course doctor, Examiner:
学位種別(和文)	博士論文
Type(English)	Doctoral Thesis

Powered by T2R2 (Tokyo Institute Research Repository)

ロボットを用いた知識転移

Knowledge Transfer on Robots

木村 大毅

Daiki KIMURA

大学院総合理工学研究科
知能システム科学専攻

September 2015

© Copyright by Daiki KIMURA 2015
All Rights Reserved

ABSTRACT

近年の少子高齢化対策として，一般家庭へのロボットの進出が期待されている．ところが，人が日常扱っている膨大な種類の物体などを全て学習することは困難と考えられる．更にロボットが普及すると，ロボット自体の多様化も考えられ，それぞれのロボットを個別に学習する必要が生じる．

そこで本研究では，学習した知識を自己や他機に対して転移することで，新しい事柄に関する知識や特性の異なるロボットに対する知識を瞬時に獲得可能な手法を提案する．実験では，実機のロボットも活用して，画像を用いた自己転移の実験と，マルチモーダル情報を活用した自己転移の実験，動作情報を用いた他機転移の実験を実施した．

実験より，転移によって新しい知識を高速かつ少数の学習回数で獲得可能なことを確認した．これにより，ロボットの知的発達機構に必要となる，多様かつ膨大な知識や個体の多様性に対応可能な基盤手法を提案できたと考えている．

ABSTRACT

This study proposed knowledge transfer methods for intelligent robots. This thesis is organized in five chapters as follows:

Chapter 1: Introduction

In the near future, there will be a shortage of labor due to the aging population. To compensate for this shortage of personnel, developing autonomous mental development robots is seen as a solution to this problem. However, it's impossible for a robot to learn all human knowledge such as object categories and place information. Also, when robots distribute among the population, these have to be individually taught knowledge based on its unique specifications. It's also impossible to teach all kinds of robots in such future. Therefore in this thesis, we propose novel methods to transfer to itself for acquiring a large variety knowledge within a short period of time, and also transfer learned knowledge to other robots.

Chapter 2: Self-transfer using visual information

Firstly, we focused on transferring knowledge to itself. We proposed a novel attribute-based fast online learning and transfer method by using a new proposed statistical recognition method on self-organizing and incremental neural network, called STAR-SOINN. This neural network is extended from SOINN to do statistical supervised learning; it can also memorize multiple concepts and classes with only one network. We conducted experiments using the proposed attribute-based transfer method and competitive methods to estimate attributes of an unknown animal by visual information. For example, the method has to estimate "has tail", "cannot fly", and "walks slowly" attributes from a photo of a 'pig' without knowledge of pigs; it needs to transfer knowledge acquired from previously seen animals. According to these experimental results, the recognition rate of the proposed method is equivalent to the previous online method. Also, there are numerous advantages of the proposed method compared to other methods: the number of required neural networks is minute; processing times for learning and testing are shorter; it has the ability to use noisy Internet data; it can define real-valued attributes.

Chapter 3: Self-transfer using multimodal information

During the next phase, we focused on using attribute learning for real robots. If a robot is expected to perform in the real world, the robot needs to recognize many unknown

objects by using some multimodal sensors such as a microphone, a laser sensor, and a touch sensor. Therefore we proposed an attribute-based learning and transfer method for multimodal sensors. This method uses STAR-SOINN, and has an integration method for attribute values from some modalities. By using this method, the robot can guess attributes of "paper-made" and "soft" before grasping a paper cup even if it's the robot's first time to touch the cup. Then the robot will grasp this paper cup softly. Next we compared proposed method to previous methods by two experiments to check the object recognition accuracy and to try estimating attributes of unknown objects. For the recognition task, the proposed method is more accurate and the processing time is quicker than a method that naively integrates the modalities, and also a previous method. In the estimation of attributes task, proposed method could estimate almost all attributes for unknown objects.

Chapter 4: Transferring between robots using motion knowledge

Finally we focused on a method to transfer knowledge to another robot with different specifications. If there are two robots with different arm lengths, the end-effectors of these robots with same joint angles are different. Therefore, it is difficult to share knowledge with other robots without an adaptation for specifications. We proposed a method that can adapt the motion information by using an end-effector position errors that are obtained during trials by a little bit motion information. This method is improved from an update algorithm

of the Self-Organizing Map (SOM). To test this method, we conducted experiments for quantitatively evaluation using robots in a simulator and a real robot. For instance, if one robot was taught about how to reach for a ball, the robot can share this motion skill to another robot, then the other robot can also reach. According to these experiments, a real robot can adapt and use transferred motion knowledge from the robot in the simulator.

Chapter 5: Conclusion

In this paper, we focused on a knowledge transfer method for intelligent robots. The first and second methods are based on SOINN, which was originally adapted from SOM. The third method is based on the update algorithm of SOM. Therefore, these methods use a similar algorithm and have the same network structure. In other words, these methods can be combined into one method that is a method to transfer knowledge to itself and also to other robots. This means if the robot learns an object's vital information, it can use this knowledge while it learns another object and recognize the unknown object. Also this robot can distribute knowledge to other robots. We believe these methods will be become an important baseline technology for real world robots in the future.

Contents

ABSTRACT **iii**

1 序論 **1**

1.1 研究の背景 1

1.2 知識の転移 3

1.2.1 自己転移 5

1.2.2 他機転移 8

1.3 本研究の目的 10

1.4 本論文の構成 13

2 画像情報を用いた知識転移 **15**

2.1 背景 15

2.2 提案手法 18

2.2.1 STAR-SOINN 19

2.2.2 提案手法のアルゴリズム 28

2.3 実験と評価 32
2.3.1 実験概要 32
2.3.2 従来手法 35
2.3.3 評価方法と結果 37
2.4 考察 39
2.4.1 提案手法の評価 39
2.4.2 インターネットの活用 42
2.4.3 属性の追加と削除 45
2.4.4 曖昧性のある属性 47
2.4.5 今後の課題 48
2.5 まとめ 48

3 マルチモーダル情報を用いた知識転移 50

3.1 背景 50
3.2 提案手法 55
3.2.1 学習フェーズ 55
3.2.2 認識フェーズ 58
3.2.3 辞書を用いた物体の特定 60
3.3 実験と評価 60

3.3.1 有効性の検証 61
3.3.2 未知物体の属性の推定 70
3.4 考察 72
3.4.1 ロボットの基盤技術となる可能性 72
3.4.2 インターネットの活用 73
3.4.3 今度の課題 73
3.5 まとめ 74

4 動作情報を用いた知識転移 75

4.1 背景 75
4.2 提案手法 79
4.2.1 仮定 79
4.2.2 経験ベースの逆運動学 80
4.2.3 問題設定 82
4.2.4 提案手法 83
4.3 実験 86
4.3.1 2 軸のロボット上での実験 86
4.3.2 実機のロボット上での実験 93
4.4 考察 98

4.4.1 SOM 変形法と誤差伝播法の違い 98

4.4.2 転移先のロボットから取得するサンプルの変動 101

4.4.3 提案手法の可能性 105

4.4.4 今後の課題 106

4.5 まとめ 106

5 結論 **107**

5.1 自己転移と他機転移 107

5.2 本論文のまとめ 109

REFERENCES **110**

PUBLICATIONS **122**

ACKNOWLEDGMENTS **127**

List of Figures

1.1 転移学習の概念図 4

1.2 学習時の自己転移の概念図 6

1.3 識別時の自己転移の概念図 7

1.4 他機転移の概念図 9

1.5 本論文で提案するロボットを用いた知識転移 12

2.1 SOINN の学習フローチャート 20

2.2 STAR-SOINN の学習フローチャート 23

2.3 画像情報を用いた知識転移に関する提案手法の概要図 29

2.4 画像情報を用いた知識転移に関する実験の概要図 32

2.5 本実験で用いた動物画像の一例 33

2.6 本実験で用いた動物と属性の関連度 34

2.7 AT-SOINN の概要図 36

2.8 学習画像数の割合と認識率の関係 39

2.9 学習時間による比較 40
2.10 識別時間による比較 40
2.11 インターネット上の画像を用いた結果 43
2.12 ラベル拡散による効果の例 46

3.1 マルチモーダル情報を用いた知識転移の概要 53
3.2 マルチモーダル情報を用いた知識転移に関する提案手法の概要図 . . . 56
3.3 使用したロボットや，センサ，物体の外観 63
3.4 使用した画像情報の一例 64
3.5 本実験で用いた物体と属性の関連度 65
3.6 学習時間の増加に関する比較 68
3.7 未知物体に関する推定結果 71

4.1 動作情報を用いた知識転移の概念図 77
4.2 それぞれのロボットに対するエンドエフェクタの可動領域 87
4.3 関節値に制限ありの場合の駆動誤差 90
4.4 関節値に制限なしの場合の駆動誤差 91
4.5 使用したシミュレータ上のロボット 93
4.6 使用した実機のロボット 94
4.7 ロボットの関節の場所と回転方向 95

4.8　実機のロボットを用いた駆動誤差 97

4.9　SOM 変形法を用いた転移後の可動領域 99

4.10 誤差伝播法を用いた転移後の可動領域 100

4.11 転移先のサンプル数を変動させた時の誤差（ロボット A への転移） . . 102

4.12 転移先のサンプル数を変動させた時の誤差（ロボット B への転移） . . 103

4.13 転移先のサンプル数を変動させた時の誤差（ロボット C への転移） . . 104

5.1　本論文の手法を用いた自己転移と他機転移の統合 108

List of Tables

2.1 未学習動物に対する認識率の比較 38

2.2 作成された SOINN の数と保有ノード数の比較 38

2.3 画像の自己転移に関する各手法の比較 49

3.1 マルチモダール情報の自己転移に関する他手法との比較 52

3.2 本章での記号の意味 57

3.3 本実験で使用する物体と属性の一覧 62

3.4 各手法及び各モダリティによる認識率の比較 67

3.5 提案手法の計算時間 69

4.1 2 軸のロボットに対する腕の長さ 86

4.2 それぞれの手法による駆動誤差 92

4.3 6 軸のロボットに対する腕の長さ 96

4.4 実機のロボットを用いたそれぞれの手法による駆動誤差 98

Chapter

1

序論

1.1 研究の背景

現在，日本の少子高齢化の進行は深刻であり，平成 23 年 10 月に総務省統計局から発行された平成 22 年度の国勢調査 [1] の中でも，少子高齢化は顕著に現れている．そのため，将来的に国内の労働力人口の減少が懸念されており，それにより労働力不足や作業負荷の増大が発生すると考えられている．また，経済産業省の「産業構造ビジョン [2]」では，今後の日本は自動車業依存から脱却し，新しい戦略五分野の強化に切り替えていくとしている．そのなかでも先端分野として期待が高まっている「ロボット」は，今後の重要な日本の市場と言え，基幹となる新産業を創出せねばならな

いとされている．そこで，前述した今後の問題として懸念されている労働力人口の減少を補い，更には新産業となり得るひとつに「人の生活環境下で人のために自律的に活動する知能ロボットの開発」が重要であると考えられる．

近年，様々な種類のロボットが様々な研究機関や企業で開発が進められている．例えば，一般家庭の部屋の掃除を目的として開発されたロボットであるルンバ（Roomba）が挙げられる．ルンバは，人工知能の研究者であるロドニー・ブルックスらが創設した iRobot 社の製品であり，包摂アーキテクチャ（Subsumption Architecture [3]）を用いることにより自動で掃除を行うことができる．他にも Honda 社の ASIMO は，世界的に著名であり，運動性能については世界でも群を抜いて優れている [4]．また，大阪大学や産業総合研究所が開発した人の姿に酷似したロボット HRP シリーズ [5] や，SoftBank 社と Aldebaran Robotics 社が発表した家庭用の人型ロボット Pepper [6] なども人の生活環境下で活動しうるロボットとして挙げられる．更に近年では，無人小型飛行機（Drone）の開発も盛んに行われている [7].

ところが，ハードウェアとしてのロボットは開発されている一方で，ハードウェアを知的に動作させる知的発達機構の開発は発展途上である．そのため，様々な研究機関によってロボットの知能に関する研究が実施されており，強化学習 (Reinforcement Learning) を用いたロボットの動作の学習に関する研究 [8] や，ニューラルネットワーク（Neural Network）を用いた研究 [9] が挙げられる．更に 2009 年には，IEEE より Autonomous mental development という雑誌が出版されている．この雑誌では，ロボッ

トを知的に動作させるための技術について議論が行われている．また，ロボットの動作以外にも視覚を用いた自己位置推定の研究として，災害時の支援物資の搬送を目的としたドローンを用いた研究 [10–12] なども盛んに行われている．

さて，このようなロボットの社会への進出を想定した時，多種多様な知識を，様々な特性を持つロボットに対して学習させる必要が存在する．“多種多様な知識” とは，ロボットが人の生活環境下で稼働するために必要となる知識が多様であることを意味している．‘多種’ とは，人が普段理解している知識の根源が，視覚や聴覚，触覚などの様々な感覚から生成されているものであることを意味しており，‘多様’ とは，人の生活環境が刻々と変化することや，その環境の状態が多様に存在していることを意味している．“様々な特性を持つロボット” とは，今後社会に多くのロボットが進出した時に，現在のパソコンや携帯電話のように，センサの種類や腕の長さなどが異なる様々なロボットの出現を意味している．本論文では，これらの多様性が持つ知識を効率的に学習するために,「知識の転移」を取り入れた知的発達機構を構築する．

1.2　知識の転移

知識の転移とは，新しい事柄を学習や認識する際に，他の事柄を学習する時に獲得した知識を参考にすることを意味している．この時，特に学習時に転移を活用したものを「転移学習」[13] と呼ばれ，少ない学習回数により学習を完了させることが可

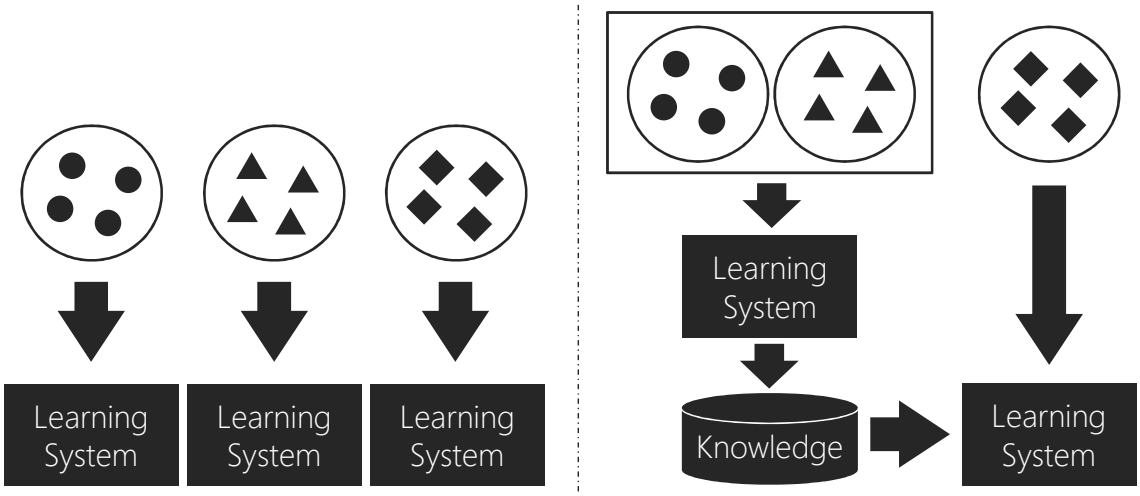

Figure 1.1: 転移学習の概念図
左：一般的な学習，右：転移学習

能な学習様式である．ここで転移学習の概念を Figure 1.1 に示す．認識時に転移を活用した場合，未学習な事柄に対しても，既知の事柄に関する知識を用いて可能な限り認識を行う．このような認識手法は，学習するべき知識に多様性がある人の生活環境下などで特に有効である．本論文では，転移学習と，転移を持ちいた認識手法を含めて，“知識の転移” と呼ぶこととする．

さて，ロボット上での知識の転移を想定した場合，２種類の転移が存在していると考えられる．それらは，単一のロボット内での転移と，ロボット間での転移である．前者は，様々な事柄で共有可能な知識，例えば物事の共有概念に属される知識を，新しい知識に活用することで実現される．後者は，新しい機体上での学習時に，既に学習したロボットの特性との差異を活用することで実現される．なお，本論文では，前

者の転移を “自己転移”，後者の転移を “他機転移” と呼ぶ.

1.2.1 自己転移

自己転移とは，新しい事柄の学習や識別時に，他の事柄で学習した共有可能な概念を活用することを意味している．学習時に活用することで，少ない学習回数により新たな事柄の獲得が可能となる．本論文では，このような転移を，“学習時の自己転移” と呼ぶ．一方で，識別時に活用することで，未学習の事柄であっても，共有概念について推定することができる．このような特性は，全ての知識を獲得困難な実世界の問題に対して特に有用である．その一例として，未学習の動物を学習済みの動物が持つ「属性」知識の転移を用いて推定する研究 [14] が挙げられる．このような認識手法を，‘属性の転移による識別’ と呼ばれ，近年様々な研究 [15–17] で用いられている．本論文では，このような転移を，“識別時の自己転移” と呼ぶ.

それでは，学習時，および認識時の自己転移に関して，その詳細を記述する.

まず，学習時に自己転移を用いた時の概念図を Figure 1.2 に示す．図の上は既に学習している知識を，下は新たな知識を学習するときの様子を表している．ここで上から下への矢印は，知識の転移を表している．この図では，下の新たな知識を学習する前に，既に三角と丸という形を学習している．この状態で，新たな知識となる四角を学習する時に転移を活用している．すなわち，四角形の角と三角形の角が似ていることを学習器が判断できれば，四角形と三角形に対して角の個数の違いのみを学

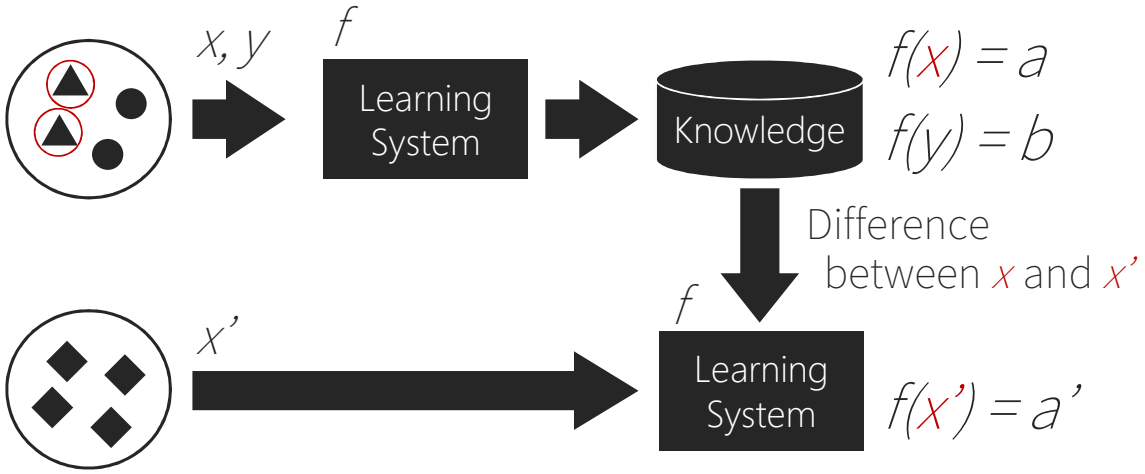

Figure 1.2: 学習時の自己転移の概念図

習するだけで早い段階で学習が完了すると考えられる[1]．これを概念的にまとめると Figure 1.2 及び以下のようになる．f は“学習器が学習した関数”，$f(x) = a$ は“x という入力を与えた時に a を出力した”を表している．この時，x を三角形，y を丸として，学習によって“三角形は角が 3 つある”（$f(x) = a$），“丸は角を含まない”（$f(y) = b$）という知識を獲得しているとする．この状態から，新たに四角形 x' を学習することと考えた時，既に学習している三角形 x と四角形 x' の類似性より，$f(x) = a$ の知識を活用することで，素早く“四角形は角が 4 つある”（$f(x') = a'$）という知識を獲得できる．このような知識を再利用しながら学習することを“学習時の自己転移”と定義する．

次に，識別時に自己転移を用いた時の概念図を Figure 1.3 に示す．図の上は既に学習

[1]正確には四角形と三角形は角の角度も異なるが，ここではそれを共に「角」と定義したとする

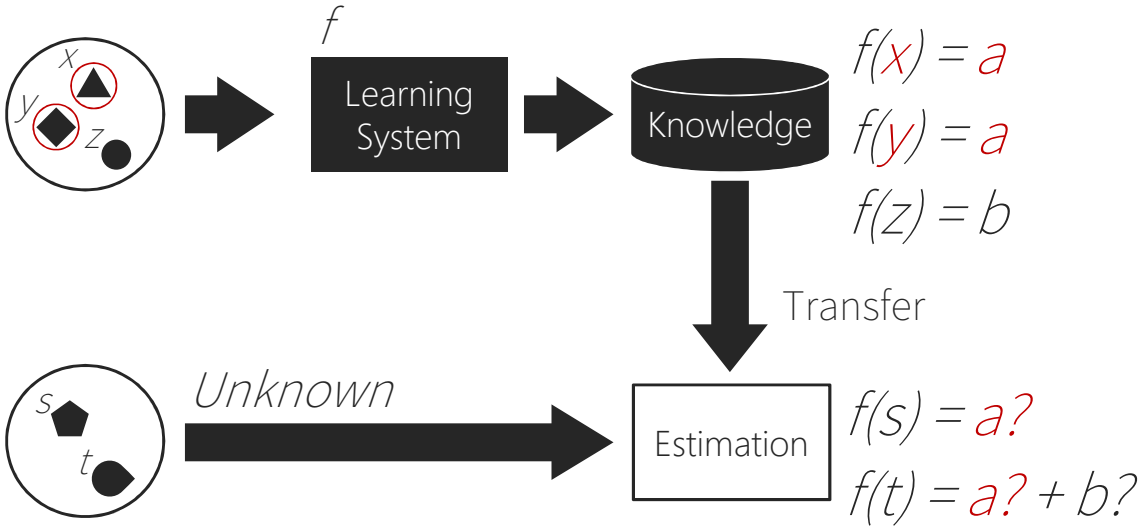

Figure 1.3: 識別時の自己転移の概念図

している知識を，下は新たな知識を学習する時の様子を表している．ここで，Figure 1.2 と同様，上から下への矢印は知識の転移を表している．この図では，上の四角形や三角形，丸を既に学習しており，下の五角形や，曲線と角を持ち合わせた図形を新たに学習するときの様子を示している．すなわち，新たな五角形という図形を認識する時に，三角形や四角形で獲得した「角」という知識を活用し，更に曲線と角を持ち合わせた図形では「角」と「曲線」という知識を活用する．このような特性を用いることで，未学習の事柄に対しても可能な限り識別する．これを概念的にまとめると Figure 1.3 及び以下のようになる．f は“学習器が学習した関数”，$f(x) = a$ は“x という入力を与えた時に a を出力した”を表している．この時，x を三角形，y を四角形，z を丸として，学習によって“三角形には角がある”（$f(x) = a$），“四角形には角があ

る”（$f(y) = a$），“丸には曲線がある”（$f(z) = b$）という知識を獲得しているとする．この状態から，学習していない五角形 s が与えられた時，三角形や四角形を用いて学習した「角」が含まれていると判断している（$f(s) = a?$）．また，学習していない曲線と角を持ち合わせた図形 t が与えられた時,「角」及び円を用いて学習した「曲線」が含まれていると判断している（$f(t) = a? + b?$）．このように，未学習の知識に対しても，共有概念を用いることで，ある程度推定することが可能である．

1.2.2　他機転移

他機転移とは，異なる特性を持つロボット間での知識の転移を意味している．他機転移を実現することで，転移元のロボットでの学習知識を，新しいロボット上でも活用可能となる．例えば，腕の長さが異なるロボットが複数台存在する時，一つのロボットで腕の動作情報を十分に学習させることで，他のロボットでも同じような動作を実行可能なことを意味している．ところが，特性の異なる転移先のロボット上で，転移元の知識をそのまま活用することは難しいと考えられ，転移先のロボットでも，転移元の知識を活用するための多少の学習は必要と考えられる．そのため，他機転移においては，学習時の転移が重要となる．また，他機転移においては，転移を行う２つのロボットに類似性が存在していることを前提としている．それでは以下に，学習時の他機転移の詳細を記述する．

学習時の他機転移を用いた時の概念図を Figure 1.4 に示す．この図では，２台の特

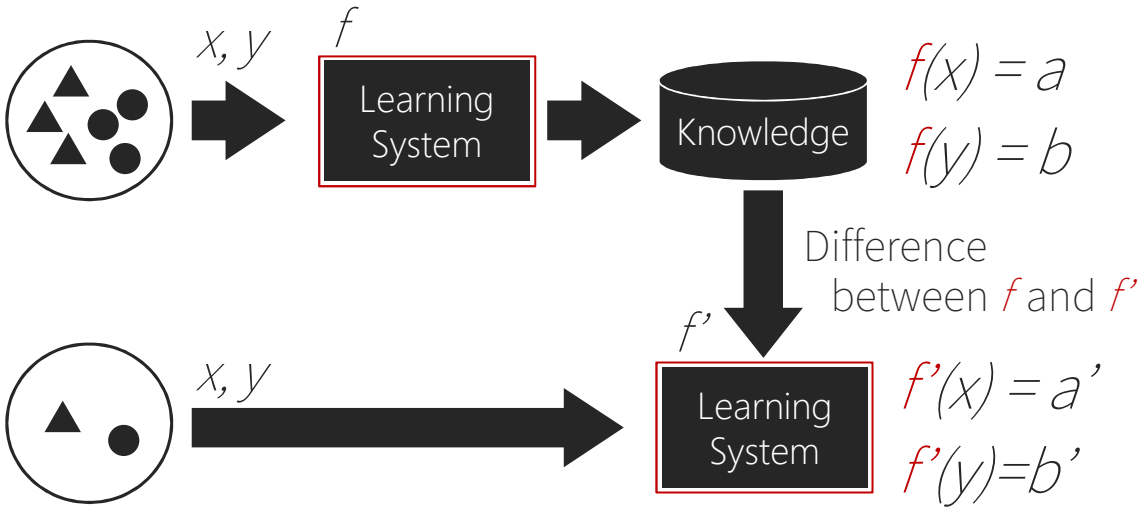

Figure 1.4: 他機転移の概念図

性の異なるロボットを作成したと考え，上が転移元のロボット上での学習，下が転移先のロボット上での学習を表している．既に三角と丸という知識を転移元のロボットで十分に学習しているとして，特性の異なる転移先のロボット上で同じ三角と丸という知識を学習する時の様子を表している．これを概念的にまとめると Figure 1.4 及び以下のようになる．f は“ロボットの特性を考慮した上での学習した関数”を，$f(x) = a$ は“x という入力を与えた時に a を実行した”を表している．そして異なる特性を持つ新たなロボットの学習すべき関数を f' とすると，転移先のロボット上で学習すべき関数は $f'(x)$ や $f'(y)$ となる．そこで，転移先のロボット上で x について学習する時，転移元で獲得した知識 $f(x)$ と，単純に新たなロボット上で実行した時の出力 $f'(x)$ の差分を計算して，f を補正することで転移先のロボット上での関数 f' を作成する．す

なわち，転移元での知識を転移先のロボット上で試し，その違い（誤差）を計測して，転移元の知識を転移先のロボット上で活用できるように補正する．この時，関数 f と f' は似ているほど転移効果が期待でき，つまりは特性が似ているロボット同士の方が転移の効果を期待できる．

1.3　本研究の目的

本研究の目的は，知識転移を活用した効率的な知識の獲得を目標とする知的発達手法の構築である．これらは，ロボットが社会へ幅広く進出した時に “多種多様な知識” を “様々な特性を持つロボット” に対して学習させる必要が存在し，それを解決するための基盤技術になり得ると考えている．そこで，本研究では更に，提案する手法の定量的な有効性の確認に留まらず，実際にロボット対して手法を実装して有用性の検証を行う．

そして，知識転移には自己転移と他機転移の 2 種類があり，どちらの転移も実世界で稼働するロボットにとって必要な要素であると考えられる．自己転移を活用することにより，単一ロボット内における未学習の知識に関する効率的な学習および認識が可能となる．他機転移を活用することにより，あるロボット上で学習した知識を，特性の異なる新たなロボットの知識に転移することができ，学習回数の削減が可能となる．

ここで，これらの知識転移に関する具体的な応用例について考える．まず自己転移を考えた時，転移に必要となる共有概念として物事の「属性」というものが良いと考えられる．この属性とは，物体を対象とすれば「四角い」「黒い」「固い」などが当てはまる．この属性という共有概念の転移を活用して，実世界の物体や動物などを認識する実験を試みる．更にこの時，ロボットに対象としているので，画像や音声，触覚などの様々なモダリティを活用できる仕組みを検討する必要がある．

次に，他機転移について考える．他機転移において重要となるのは，ロボットの特性により大きな変動がある知識を対象とすることである．上で記述した自己転移の例のような，画像や音声センサの場合，機体によって多少の誤差はあるものの，特性により大きな違いは存在していない．その一方で，ロボットの腕や足などの動作について考えてみる．これらは，ロボットの特性（例えば，腕の長さ）によって大きく変動してしまう．そこで，他機転移においては，ロボットの動作を対象として実験を行う．すなわち，腕の長さなどのロボットの特性を変更したロボットをシミュレータ内や実機で用意をして，それらの間で動作情報の転移を実現する．

ここで，本論文で提案する手法を，Figure 1.5 に示す．図中の左側のロボットが転移元のロボットを表しており，単一のロボット内での知識転移（自己転移）と，右側の特性の異なるロボットへの転移（他機転移）を実現する．この時，転移する情報は，ロボットに搭載されたカメラ，マイク，触覚センサから得られたセンサ値や，ロボットの腕に関する動作情報である．次に，それぞれ転移に関して実現する手法の詳細を

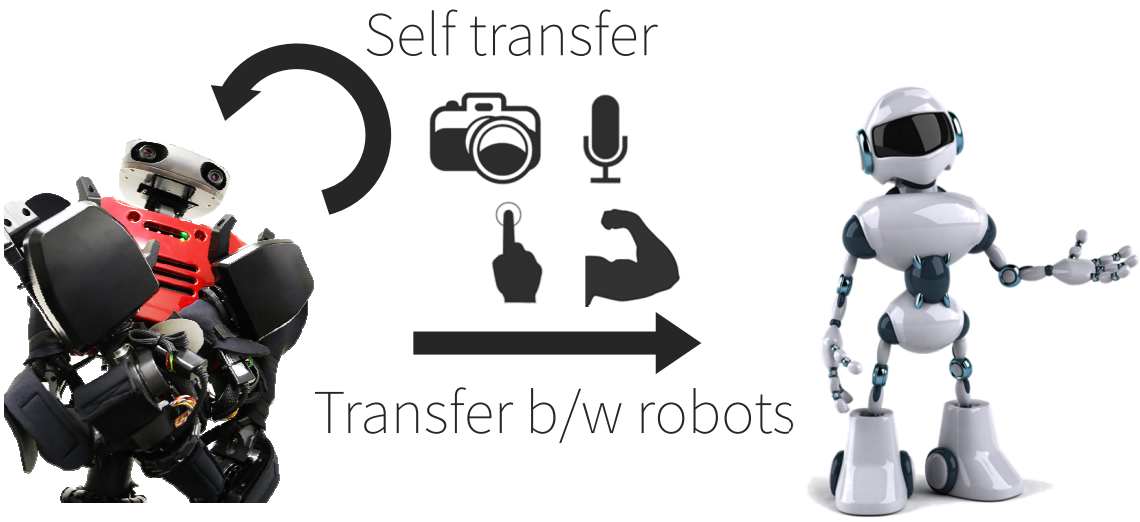

Figure 1.5: 本論文で提案するロボットを用いた知識転移

以下に記す.

1). 自己転移

i). 画像情報のみを用いた手法

ii). 様々なセンサ情報を統合的に活用した手法

2). 他機転移

i). ロボットの動作情報を用いた手法

1.4　本論文の構成

本論文は，5 章から構成されている．各章の主な内容を以下に示す．

第 1 章

序論として本論文の背景と目的について述べる．

第 2 章

画像情報のみを用いて自己転移に関する基盤的な手法を提案する．その後，従来手法などと比較評価実験を実施して，手法の有効性を検証する．また，インターネット画像などの活用に関して考察を行う．

第 3 章

ロボットが持つ様々なマルチモーダルセンサから取得される情報を統合的に活用する自己転移の手法を提案する．その後，従来手法などと比較評価実験を実施する．そして，ロボットに対して実装し，手法の有用性について議論する．

第 4 章

特性の異なるロボット間での動作情報の他機転移の手法を提案する．実際に特性の異なるロボットをシミュレータ内などに容易をして，手法の有効性について検証する．その後，実際のロボットとシミュレータ上のロボットを活用して転移手法の有用性について議論する．

第 5 章

本論文の結論について述べる．特に，本論文での提案した手法を用いることで，ロボットを用いた知識転移として実現可能な範囲を明確にする．

Chapter

2

画像情報を用いた知識転移

2.1 背景

人の生活環境下で稼働するロボットの構築を考えた時，人が日常使用している膨大な種類の物体を認識する必要がある．また環境は刻々と変化するために，追加的に知識を獲得する知的発達機構も必要となる．そこで，ノイズは多く存在するが多様な大量の情報を瞬時に取得可能なインターネットからも，新たな知識を追加的に獲得する機構が必要であると考えている．

現在，物体認識の分野では画像を用いた研究が主流である．特に，特定の物体や顔などに対して個別の学習器を作成して，それらを認識するものが主流である [18–21].

ところが，人が日常に判別している物体は少なくとも 30,000 種類あると言われており [22]，すべての物体に対して個別の学習器を作成することは現実的に不可能である．そのため近年，属性ベースの学習手法が注目されている [14]．属性ベースの学習手法とは，物体などがもつ属性（大まかなカテゴリ）を基準にして学習する手法であり，属性の組み合わせにより物体を表現する．そのため，少ない属性の学習で多くの物体を表現することが可能となる．また，学習した属性を用いることで，実世界に多く存在する未学習物体も，属性を推測することができる．更に，求められた属性と，属性と物体の種類を列挙した辞書を用いることで，物体の種類も推定することができる．

ここで，具体的に従来研究について記述する．現在，属性ベースの学習手法は多く存在し [14–17]，その代表例として Lampert らの研究が挙げられる [14]．彼らは，属性ベースの学習に関する 2 つの学習方式を提案した．更に彼らは，実際の動物の画像とその属性を利用して評価実験を作成し，そのデータセットと SVM [23] を用いて，それぞれの学習方式を評価した．なお用いられた属性の例として,「肉食動物である」や「飛ぶことができる」,「凶暴である」が挙げられる．ところが，彼らの手法はバッチ学習であり，追加的に知識を増やすことが出来ない．実環境で稼働するロボット上での運用を想定した時，予め学習物体を全て与えることは困難である．環境に応じ，その場で追加学習できる機構が必要である．そのため，オンライン追加学習が，ロボットのための知的発達機構に対して必須となる．

そこで，Kawewong らは，オンライン追加学習が可能な属性ベースの学習手法を

提案した [17]．そして，従来手法 [14] と比較実験を行って評価している．ところが，彼らの手法は属性の増加に伴い学習や認識の処理時間などが増大する問題が存在する．これは，学習器の数が属性の数と比例しているためである．その理由は，そもそも学習器 SOINN が教師なしクラスタリングのために設計されたものであり，教師情報を管理するアルゴリズムが存在しなく，更に複数のクラスタを持つ概念を学習する時，１つの学習器で１つの概念しか学習できないためである．

そこで本稿では，まず，自己増殖型ニューラルネットワーク [24] に対して，教師情報に対する統計情報の算出アルゴリズムを追加し，教師情報の統計的な推定機構を取り入れる．これにより，教師あり学習に対応することができ，更に１つの学習器で複数の教師情報に対する学習が可能になる．単一の学習器で学習可能なことは，新たな教師情報を学習する時に，既に獲得している他の教師情報に関する知識を活用可能なことを意味しており，学習時の転移を実現している．そして，改良した自己増殖型ニューラルネットワークを活用して，学習器の数が少ない属性ベースの自己転移手法を提案する．学習器が少ないため，計算回数が少なくなり，高速な属性知識の学習と転移を実現できると考えられる．そして，[14] や [17] と同様の実験で本手法の評価を行う．考察では，ノイズは多いが大量の学習データを保持しているインターネットからも知識を獲得できることを確認し，本手法が持つ有用性についても考察する．

なお現在，オンライン学習可能な SVM が提案されている [25]．ところが，この学習器を用いた場合にも，学習器の数が属性の数に比例しており，属性の増加に伴い学

習や認識の処理時間が増大する．本稿では，ベースラインとして SVM を用いた手法の結果を参照するために，オンライン手法に比べて安定性があるバッチ学習 [14] との比較を行う．

以降，本章は以下のように構成されている．まず，2.2 節では改良した自己増殖型ニューラルネットワークと，属性ベースの自己転移手法の詳細について述べる．次に，2.3 節では提案手法の評価について述べ，2.4 節では提案手法に関する考察と今後の課題について述べる．最後に，2.5 節では本章のまとめについて述べる．

2.2 提案手法

ここではまず，自己増殖型ニューラルネットワーク (SOINN) に対して，教師情報の統計的な推定機構を取り入れ，一つの学習器で複数の知識を学習可能とする手法 “STAtistical Recognition-SOINN”（以降，STAR-SOINN）を提案する．その後，提案した自己増殖型ニューラルネットワークを活用した，オンライン学習可能で非常に軽量な属性情報の学習と転移手法を提案する．

なお，この時，学習器として SOINN を選んだ理由は，実世界のデータに対して非常に有用な次の 3 つの特性を持つためである．

- 学習に対する事前情報が不要である特性
- オンライン追加学習が可能である特性

- 実世界のノイズに対応する強いノイズ耐性

また本章では，以下の単語を次のように定義し用いている．

クラス 物体や動物の種類（例：リンゴ）

属　性 物体や動物の大まかなカテゴリ（例：球体）

ラベル 学習器が持つ学習カテゴリに関する情報

2.2.1 STAR-SOINN

SOINN [24] に対して統計を取り入れ，一つのネットワークで複数概念を学習できるように改良する．まず，従来の SOINN について記述する．

2.2.1.1 SOINN

SOINN とは，SOM [26, 27] と GNG [28] に着想を得て構築された学習手法である．具体的には，パターン情報の入力に合わせてニューロン（ノード）が増殖や消滅を行い，仮想的なノード同士の繋がり（エッジ）を用いて構造化されるネットワークモデルである．現在 SOINN には複数の構造（アルゴリズム）が存在し，多くの論文で紹介されている [24, 29–32]．その中でも Adjusted SOINN [24] は，他の SOINN と比べて，パラメータ数が少なく，計算時間が短いなどの優位点があり，SOINN を用いた応用研究で使用されている [33–39]．そのため本稿では，Adjusted SOINN を改良する．

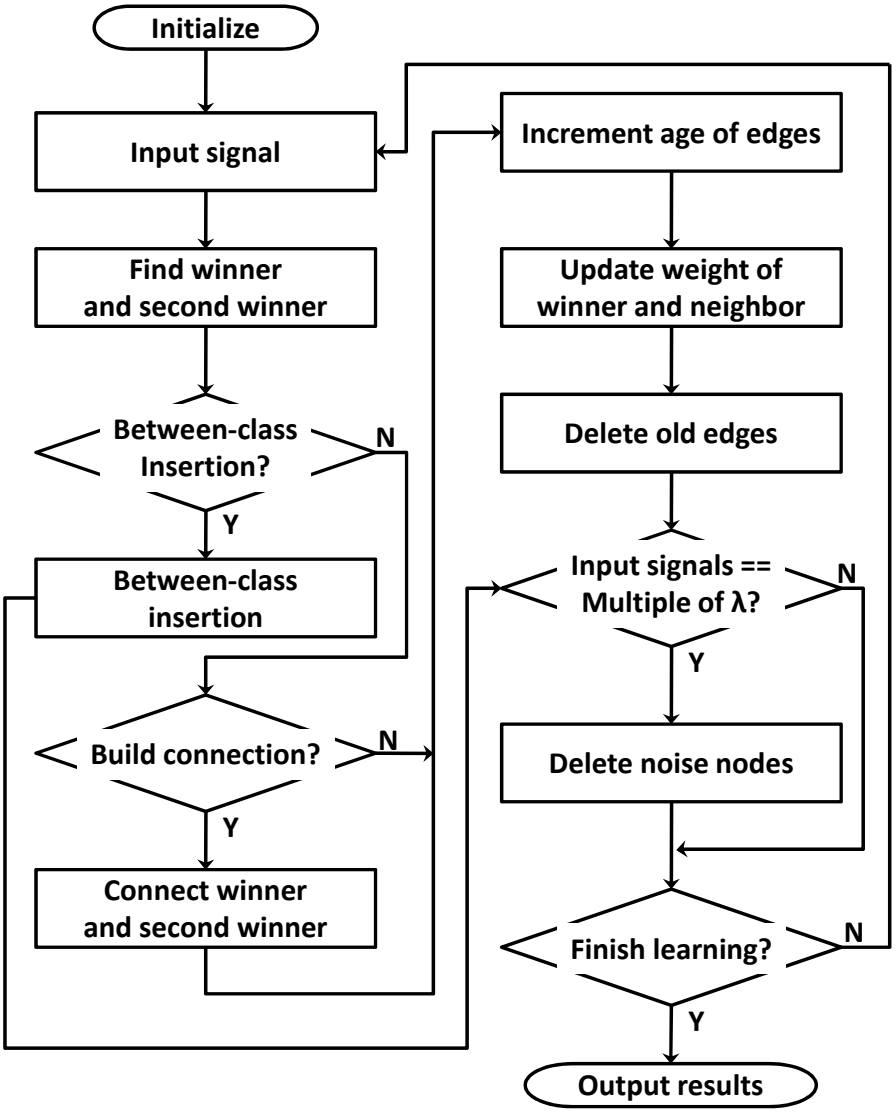

Figure 2.1: SOINN の学習フローチャート

参考のために，Adjusted SOINN の学習アルゴリズムのフローチャートを Figure 2.1 に示す.

2.2.1.2　STAR-SOINN の概要

STAR-SOINN は，従来の SOINN に対して，大きく 2 つの改良を行なっている．

1 つは，ノードの挿入や削除，及びノードの重みベクトルを調節する処理の改良である．これは，従来の SOINN のネットワークがデータの入力順序に依存する問題（入力に対する不安定性）を軽減する目的で行なっている．従来の SOINN では，入力されたデータ数がある定数の倍数になった時に，ノイズの削除処理を行っていた．ところが，削除処理のサイクルの中で，早い段階で入力されたデータと，遅い段階で入力されたデータで，ノイズの削除処理を行うまでのステップ数が異なっていた．このような差異は，入力順序による学習結果の不安定化の原因の一つであると考えることができる．そこで，全てのノードに対して年齢を導入し，ノイズの削除処理までのステップ数が全ノードで同じになるように改良する．また，ノイズに対する判定条件の明確化や，重みベクトルの更新処理のヒューリスティックな部分，例えば第 2 勝者ノードの更新時に 1/100 を乗じるなどの廃止などを行なっている．

もう 1 つは，ノードにクラスの統計情報を追加した改良である．SOINN は，ネットワークモデルを用いたパターン情報の記録という観点では様々な優位点を保持している．しかし，クラスに関する教師ありデータの学習に関しては，未だ議論がされていない．そのため従来は，クラスごとにネットワーク（SOINN）を作成し，認識データと，それぞれのネットワーク内のノードとの距離を測定することで，クラスの推

定を行なっていた．具体的には，認識データと各ネットワークに対して固定数の近傍ノードを探し，それらノードと認識データとの距離の総和が最も短いネットワークのクラスを，推定されたクラスと設定していた．ところが，この手法では，クラスと同数の SOINN が必要となり，学習後のデータの量や学習時間がクラス数と比例する問題点があった．更には，他のクラスでの学習結果を活用できない問題点が存在していた．そこで本稿では，クラスに関する教師信号を「ラベル」としてノードに付加する機構と，ネットワークの変化に合わせてラベル情報の変更を行う機構を追加する．これにより，1 つのネットワークで全てのクラスに関する情報を学習できるため，ネットワーク数の削減ができ，更には学習時に類似性の高い他のクラスの学習結果を活用することができる．

2.2.1.3　アルゴリズム

STAR-SOINN の学習アルゴリズムのフローチャートを Figure 2.2 に示す．認識アルゴリズムは，使用目的などにより異なるため，ここでは言及しない．

学習は，大きく 2 つの処理に分けることができる．ノードやエッジなどのネットワーク構築に関する更新処理と，ノイズの除去を行うノードの削除処理である．これらは，パターン情報とクラスに関する教師信号の組が入力される度に逐次実行される．例えば，パターン情報として画像の特徴ベクトル，教師信号として学習画像に対応する物体の種類や属性などが当てはまる．

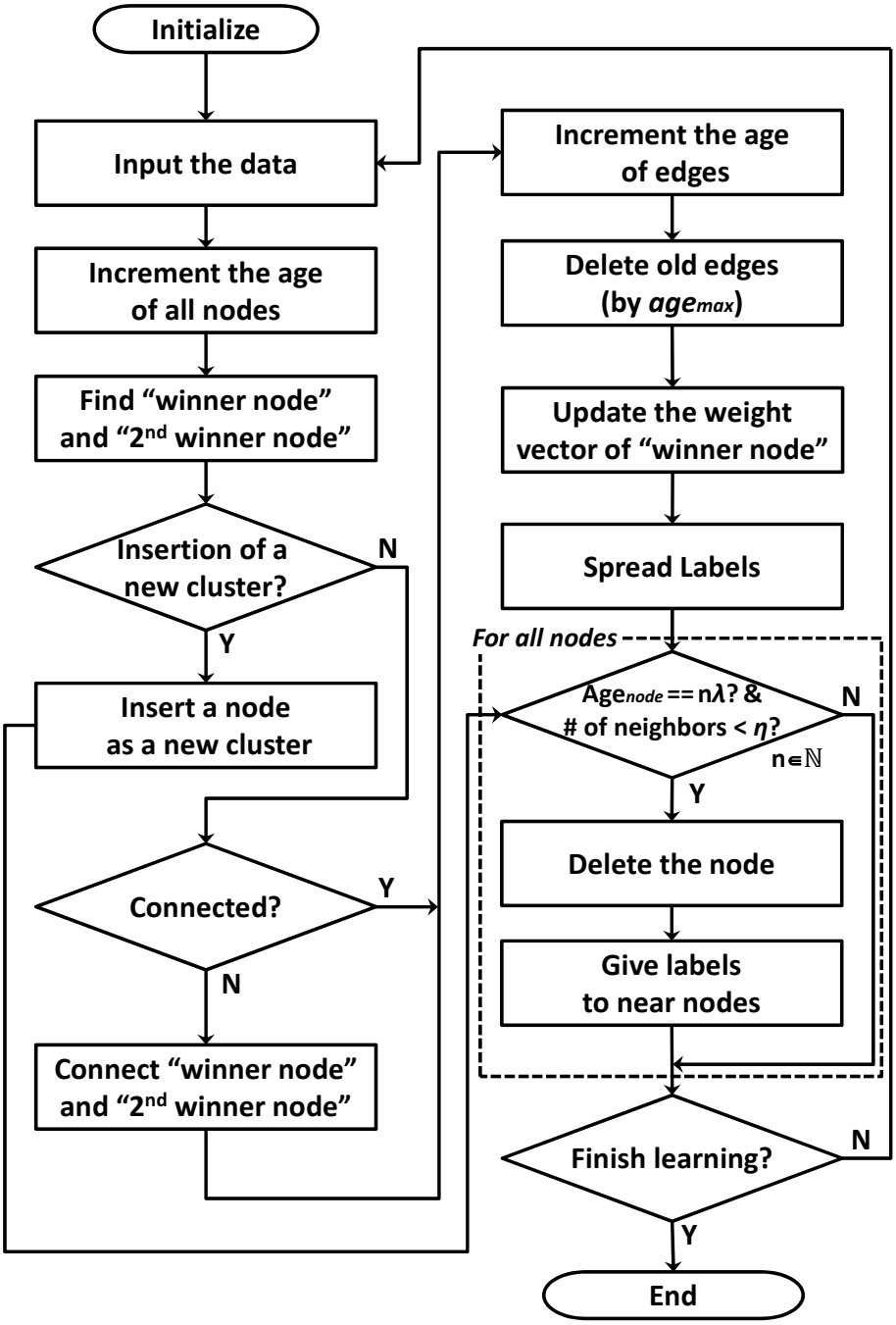

Figure 2.2: STAR-SOINN の学習フローチャート

(1) ネットワークの更新

学習データ（パターン情報と，クラスの教師信号）が入力されると，全ノードの年齢をインクリメント (+1) する．ノードの年齢とは，ノードが存在していた期間を表す．次に，入力パターンと全ノードの重みベクトルとの距離を算出する．それを基に，入力パターンに最も近い重みベクトルを持つノード（第 1 勝者ノード：$winner$=“w”）と，2 番目に近い重みベクトルを持つノード（第 2 勝者ノード：$2^{nd}winner$=“sw”）を決定する．Equation (2.2.1) を用いて，これら 2 つのノードに対する類似度閾値 T_w, T_{sw} を算出する．N_i はノード i の全隣接ノード，W_i はノード i の重みベクトル，$\mathcal{A}$ は SOINN がもつ全てのノードである．なお「隣接ノード」とは，エッジで接続されているノードを表す．

$$T_i = \begin{cases} \max\limits_{k \in \mathcal{N}_i} \|W_i - W_k\| & (\mathcal{N}_i \neq \phi) \\ \\ \min\limits_{k \in \mathcal{A} \setminus \{i\}} \|W_i - W_k\| & (\mathcal{N}_i = \phi) \end{cases} \tag{2.2.1}$$

これらの類似度閾値から，入力情報が属するクラスタを決定する．入力パターンと第 1 勝者ノードの重みベクトルとの距離が，第 1 勝者ノードの類似度閾値よりも大きい，もしくは，第 2 勝者との距離が，第 2 勝者の類似度閾値よりも大きい場合には，入力情報は新しいクラスタであると判定する．その場合には，入力パターンを重みベクトル，クラスの教師信号をラベル情報とする新しいノー

ドを挿入する．ラベル情報とは，各々のクラスとの関連度合いを保持している．例えば，“クラス α とは 75%の関連があり，クラス β とは関連がない (0%)” などが記録されている．

一方で上記以外の場合には，入力情報と，第 1 勝者ノード，第 2 勝者ノードは同一クラスタに属すると判定し，第 1 勝者と第 2 勝者の間に年齢が零のエッジを作成する．既にエッジが存在する場合には，そのエッジの年齢を零にする．エッジは，2 つのノードが同一クラスタであることを表し，エッジの年齢は，それらのノードが他のクラスタと判定された回数を表す．次に，第 1 勝者の隣接ノードの中で，第 2 勝者以外と接続しているエッジの年齢をインクリメント (+1) する．この時，年齢が，事前に定義するパラメータ age_{max} を超える場合，そのエッジを削除する．次に，第 1 勝者ノードの重みベクトル W_w を Equation (2.2.2) に従い更新する．M_w は第 1 勝者ノードが過去に第 1 勝者になった回数，V は入力パターンを表す．最後に，入力されたクラスの教師信号を基に，周辺ノードのラベル情報 L_i を更新する．これは，第 1 勝者ノードとその隣接ノード，及び，第 2 勝者ノードとその隣接ノードに対して行われ，前者は Equation (2.2.3) に，後者は Equation (2.2.4) に従う．$\mathcal{C}$ はクラスの集合，$\mathcal{N}_w$（$\mathcal{N}_{sw}$）は第 1 勝者（第 2 勝者）ノードの隣接ノードの集合，L_i^m はノード i のクラス m のラベル重み，S^m はクラス m の入力された教師信号，U_i はノード i のラベルが過去に更新された回数である．

$$\Delta W_w = \frac{1}{M_w + 1}\,(V - W_w) \tag{2.2.2}$$

$$\Delta L_i^m = \frac{1}{U_i + 1}\,(S^m - L_i^m) \qquad (\forall i \in \{winner\} \cup \mathcal{N}_w, \forall m \in \mathcal{C}) \tag{2.2.3}$$

$$\Delta L_i^m = \frac{1}{2(U_i + 1)}\,(S^m - L_i^m) \qquad (\forall i \in \{2^{nd}\ winner\} \cup \mathcal{N}_{sw}, \forall m \in \mathcal{C}) \tag{2.2.4}$$

(2) ノイズの除去

ネットワークの更新後，ノイズの可能性が高いノードの削除を行う．本稿での“ノイズ”とは，出現頻度が低いデータのことを指す．先の更新法より，ある領域に対して多くの入力が行われると，ノードやエッジが次々と作成され，隣接ノードの数が増える．そのため，隣接ノードの数が少ないことは，その領域に入力が少ないことを示している．この特性を用いて「ノードの年齢がパラメータ λ の倍数」と「隣接ノードの数がパラメータ η より小さい」の 2 つの条件式が共に真であるノードをノイズとみなし削除する．λ は判定を行う間隔を，η は

ノイズの定義を調節する．

ノードの削除時には，そのノードが持つラベル情報を周辺ノードに伝播させる．これは，上記条件では入力回数が少ない重要な分布も削除してしまう可能性があり，削除による影響を軽減する目的で行う．伝播は，削除するノードの重みベクトルと最も近い重みベクトルを持つノード fn のラベル L_{fn}，及び 2 番目に近いノード sn のラベル L_{sn} に対して行う．前者は Equation (2.2.5)，後者は Equation (2.2.6) の更新式に従う．$\mathcal{K}_d$ は削除するノード d が持つラベルの集合，L^m_{fn}（L^m_{sn}，L^m_d）はノード fn（sn，d）のクラス m のラベル重み，T_{fn}（T_{sn}）はノード fn（sn）の類似度閾値，U_{fn}（U_{sn}）は L_{fn}（L_{sn}）が過去に更新された回数を表す．また，D_{fn}，D_{sn} は，Equation (2.2.7)，Equation (2.2.8) で定め，W_{fn}（W_{sn}，W_d）はノード fn（sn，d）の重みベクトルを表す．

$$\Delta L^m_{fn} = \begin{cases} \dfrac{L^m_d - L^m_{fn}}{U_{fn}+1} & (D_{fn} \leq T_{fn}) \\ \\ \dfrac{T_{fn}\left(L^m_d - L^m_{fn}\right)}{D_{fn}(U_{fn}+1)} & (D_{fn} > T_{fn}) \end{cases} \quad (\forall m \in \mathcal{K}_d) \quad (2.2.5)$$

$$\Delta L_{sn}^{m} = \begin{cases} \dfrac{L_d^m - L_{sn}^m}{2(U_{sn}+1)} & (D_{sn} \leq T_{sn}) \\ \\ \dfrac{T_{sn}\left(L_d^m - L_{sn}^m\right)}{2D_{sn}(U_{sn}+1)} & (D_{sn} > T_{sn}) \end{cases} \quad (\forall m \in \mathcal{K}_d) \tag{2.2.6}$$

$$D_{fn} = \|W_{fn} - W_d\| \tag{2.2.7}$$

$$D_{sn} = \|W_{sn} - W_d\| \tag{2.2.8}$$

2.2.2 提案手法のアルゴリズム

本節では，前節の STAR-SOINN を活用した画像情報の自己転移手法 (Attribute Transfer by STAR-SOINN : AT-STAR-SOINN) について述べる．まず，概要図を Figure 2.3 に示す．本手法は，原理的にはロボットなどで取得できる様々な情報（画像や音声，圧力情報など）を用いた学習に応用可能である．ところが，理解の補助のため，Figure 2.3 および以降の説明は，後述する実験に合わせて画像を用いた例で記述する．本手法には “学習フェーズ” と “推定フェーズ” があり，前者では入力されたデータを用いて属性ベースの学習を，後者では学習属性を転移し，クラスの推定を行う．

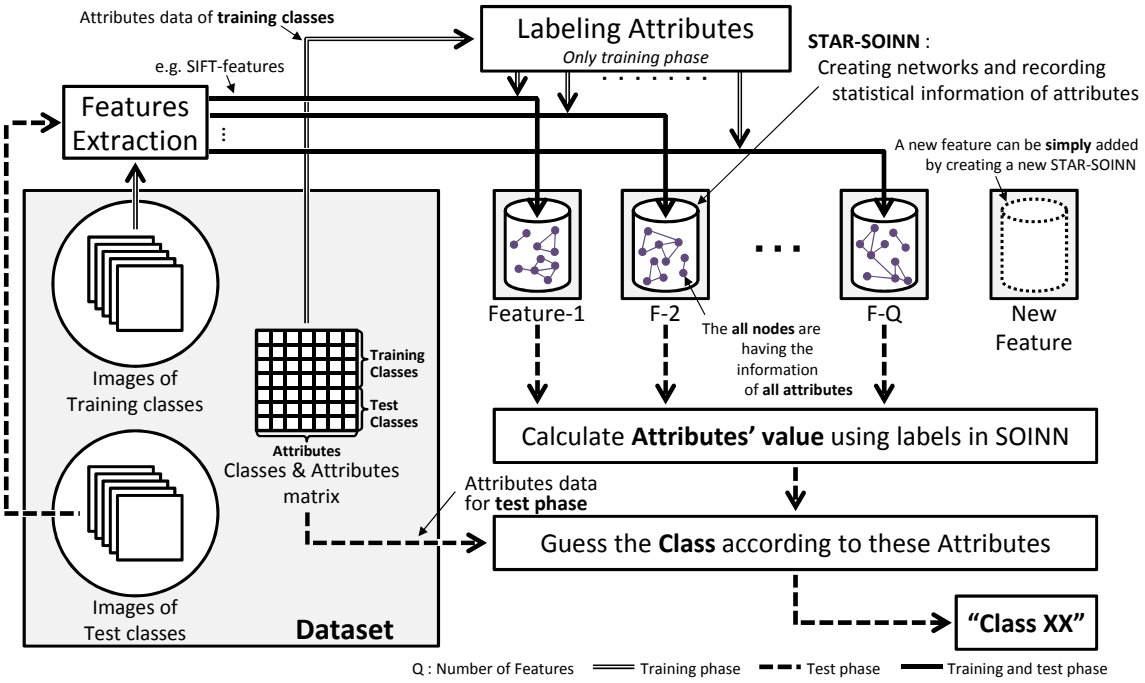

Figure 2.3: 画像情報を用いた知識転移に関する提案手法の概要図

Figure 2.3 では，学習フェーズのステップを実線と二重線で示し，認識フェーズのステップを実線と点線で表している．

2.2.2.1 学習フェーズ

始めに，学習画像から予め設定された特徴を抽出し，その特徴に学習する画像のクラスが持つ属性値を付加する．この時，“クラスと属性の関連度の表”（Figure 2.3 の “Classes & Attributes matrix”）を利用し属性値を参照する．これは，全てのクラス（動物や物体など）が持っている属性を列挙したものであり，人が事前に定義して本

手法に与える．この時，関連度は，“関連がある” と “関連がない” の二値や，実数値（確率のように 0 から 1 の間）で定義される．次に，抽出された特徴ベクトルをパターン情報，属性値をクラスの教師信号として，特徴毎に生成された STAR-SOINN へ入力し，学習させる．また，事前に定義するパラメータ ρ 番目より後のクラスを学習する際には，STAR-SOINN への入力を行う前に後述する認識処理を実施して，各々の属性の関連度を算出する．これは，認識時に必要となる属性判定の閾値決定に用いており，利用方法と計算式は後述する．

2.2.2.2 推定フェーズ

まず，認識画像から学習フェーズと同じ特徴を抽出する．抽出された特徴と，その特徴に対応する STAR-SOINN を用いて，それぞれの属性の関連度を求める．次に，“クラスと属性の関連度の表” を用いて，認識画像のクラス g を推定する．この一連のクラス g に関する推定は，Equation (2.2.9) に従う．$\mathcal{Z}$ はクラスの集合，$\mathcal{M}$ は属性の集合，$\mathcal{Q}$ は特徴の集合，V_q は入力画像の特徴 q のパターン，a_z^m はクラス z と属性 m の関連度を表す．関連度は，$0 \leq a_z^m \leq 1$ で，1 に近いほど関連度が高いことを表す．$R_q^{m^+}$ および $R_q^{m^-}$ は，学習クラス中の属性 m と関連性が高いクラスの属性値の平均および低いクラスの属性値の平均であり，Equation (2.2.10) および Equation (2.2.11) で決定される．$\mathcal{Z}_t$ は学習するクラスの集合，$\mathcal{Z}_\rho$ は学習クラスの 1 番目から ρ 番目までのクラスの集合を表す．Equation (2.2.9) や Equation (2.2.10)，Equation (2.2.11) の中で

用いられている $P^m(V_q)$ の定義を Equation (2.2.12) に示す．これは，特徴パターンの k 個の近傍重みベクトルを求め（k-近傍法：k-nearest neighbor [40]），それらのノードが持つラベルの平均である．平均を求める際には，ノードが持つ閾値で正規化された距離を用いて重み付き平均を用いている．なお，$\mathcal{N}_k(V_q)$ は V_q の k 個の近傍ノードの集合，$L^m_{q,i}$（$W_{q,i}$，$T_{q,i}$）は特徴 q の STAR-SOINN 内にあるノード i の属性 m のラベル重み（重みベクトル，類似度閾値）を表す．また，$\max(a,b)$ は a と b を比較した時の大きな値を表す．

$$g = \underset{z\in\mathcal{Z}}{\operatorname{argmax}} \left[\sum_{m\in\mathcal{M}} \left(\sum_{q\in\mathcal{Q}} \left[P^m(V_q) - \frac{1}{2}\left(R_q^{m^+} + R_q^{m^-} \right) \right] \cdot (2a_z^m - 1) \right) \right] \tag{2.2.9}$$

$$R_q^{m^+} = \frac{1}{\sum_q \sum_z \sum_m a_z^m} \sum_{q\in\mathcal{Q}} \sum_{z\in\mathcal{Z}_t\setminus\mathcal{Z}_\rho} \sum_{m\in\mathcal{M}} \left[P^m(V_q) \cdot a_z^m \right] \tag{2.2.10}$$

$$R_q^{m^-} = \frac{1}{\sum_q \sum_z \sum_m (1 - a_z^m)} \sum_{q\in\mathcal{Q}} \sum_{z\in\mathcal{Z}_t\setminus\mathcal{Z}_\rho} \sum_{m\in\mathcal{M}} \left[P^m(V_q) \cdot (1 - a_z^m) \right] \tag{2.2.11}$$

$$P^m(V_q) = \frac{1}{k} \sum_{i\in\mathcal{N}_k(V_q)} \left[0.5 + \frac{L^m_{q,i} - 0.5}{\max\left(1\,,\ \frac{\|V_q - W_{q,i}\|}{T_{q,i}} \right)} \right] \tag{2.2.12}$$

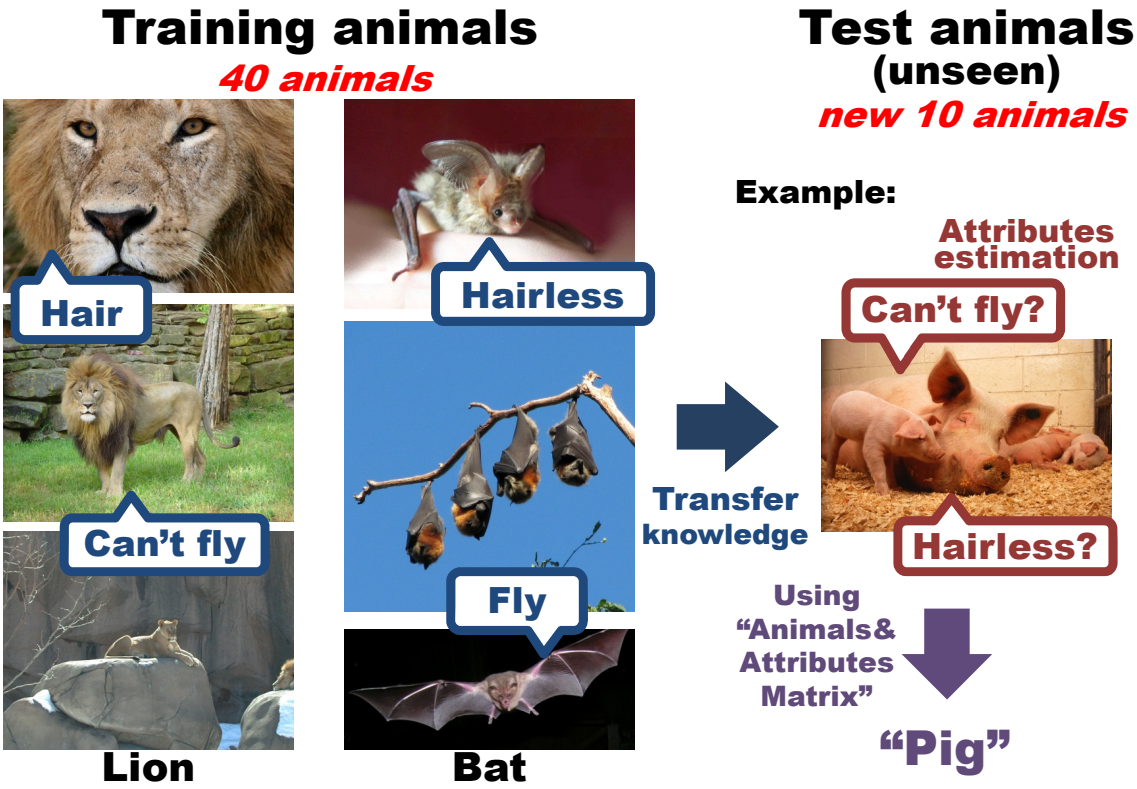

Figure 2.4: 画像情報を用いた知識転移に関する実験の概要図
既習動物の知識を転用して，未学習動物の属性を算出し，その種類を推定

2.3 実験と評価

2.3.1 実験概要

本実験の概要図を Figure 2.4 に示す．本実験で用いるデータセットは，Lampert らが作成した "Animals with attributes - A dataset for Attribute Based Classification"[1] であ

[1] http://attributes.kyb.tuebingen.mpg.de/

Figure 2.5: 本実験で用いた動物画像の一例

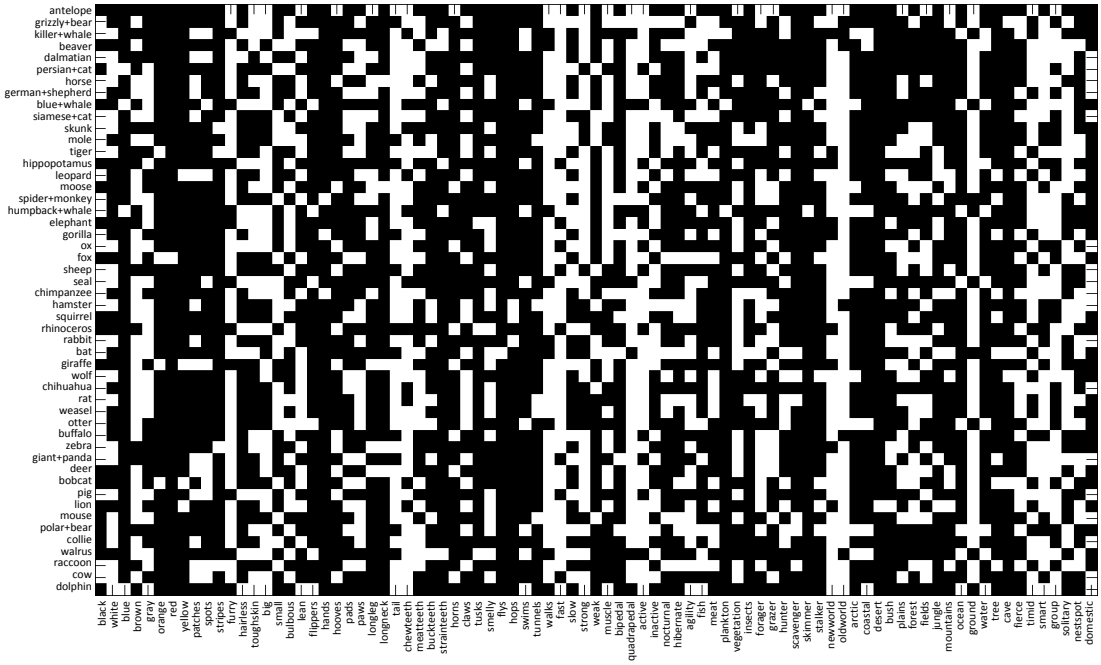

Figure 2.6: 本実験で用いた動物と属性の関連度
「白」は “関連あり”,「黒」は “関連なし”

る．このデータセットには，50 種類の動物の画像（Figure 2.5）と，各々の動物と 85 種類の属性との関連度を記した一覧表（Figure 2.6）が含まれている．なお，本実験で用いられる関連度は，“関連がある” と “関連がない” の二値のみである．50 種類の動物の内，40 種類を学習用，10 種類を認識用として定められており，それぞれの動物に対し平均 600 枚程度の画像が用意されている．なお本実験では，6 種類の特徴を使用する．その特徴とは，SIFT [41] と，SURF [42]，pHOG [43]，rgSIFT [44]，LSS [45]，CQ [14] 特徴量に対する，それぞれのヒストグラム（*Bag-of-visual-words* [46]）である．

2.3.2　従来手法

従来手法として属性情報を用いた学習手法を２つ紹介する．オフライン SVM を用いた手法 [14] と，Adjusted SOINN [24] を用いたオンライン学習手法 [17] である．以降，前者を “DAP および IAP”，後者を “AT-SOINN(Attribute Transfer SOINN)” と呼ぶ．

2.3.2.1　DAP および IAP

DAP(Direct attribute prediction) と IAP(Indirect attribute prediction) は，Lampert らが提案した学習方式である [14]．これらは学習する内容が異なり，DAP では属性と画像特徴の関連性を，IAP では学習動物と画像特徴の関連性を学習する．そのため DAP の認識処理は，特徴から属性の関連度を算出し，その組み合わせから動物を推定する．一方 IAP は，特徴から学習用動物との関連度を算出し，その組み合わせから属性値を推定して，更にその属性値から動物を推定する．Lampert らは，学習器として SVM を利用して，DAP と IAP の学習方式の違いによる認識率の変化などを確認している．

2.3.2.2　AT-SOINN

AT-SOINN は，SOINN [24] を用いた DAP 方式の手法であり，本稿では比較のために詳細を記述する．まず，AT-SOINN の概要図を Figure 2.7 に記す．次に，学習フェーズと認識フェーズについて述べる．

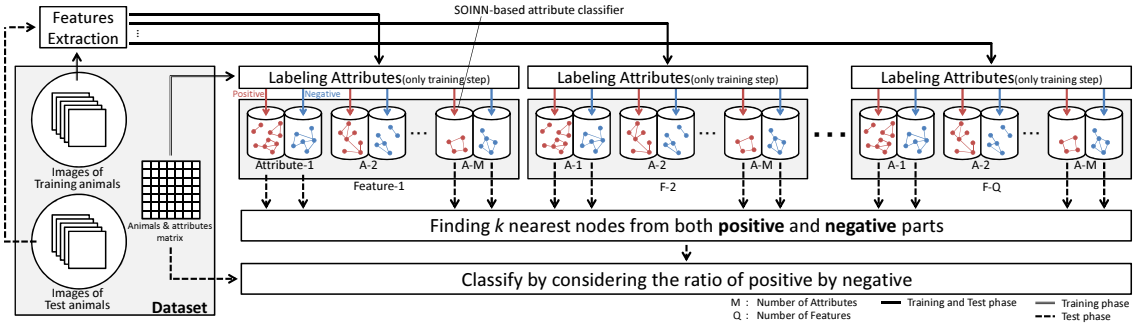

Figure 2.7: AT-SOINN の概要図

$$g = \underset{z \in \mathcal{Z}}{\operatorname{argmax}} \left(\sum_{q \in \mathcal{Q}} \left(\sum_{m \in \mathcal{M}} \left(\frac{d^p_{q,m} \left[a^m_z = pos? \right] + d^n_{q,m} \left[a^m_z = neg? \right]}{d^p_{q,m} + d^n_{q,m}} \right) \right) \right) \tag{2.3.1}$$

$$d^p_{q,m} = \frac{1}{k} \sum_{t=1}^{k} \| V_q - W^p_{q,m,t}(V_q) \| \tag{2.3.2}$$

$$d^n_{q,m} = \frac{1}{k} \sum_{t=1}^{k} \| V_q - W^n_{q,m,t}(V_q) \| \tag{2.3.3}$$

学習フェーズでは，画像から特徴を抽出して，動物の属性に従って SOINN を用いて学習する．具体的には，全ての属性に対して，関連性がある属性については，その属性の pos-SOINN へ，関連性がない属性については，neg-SOINN へ特徴を入力する．そのため AT-SOINN では，属性の数 $\times$ 特徴の数 $\times 2$ 個のネットワークが必要である．

認識フェーズでは，抽出された特徴パターンと，全ての属性の pos-SOINN と neg-SOINN に対して，重みベクトルとの距離を求め，小さな値を k 個選択して，その総和を算出する．その後，動物と属性の関連度を用いて距離を足しあわせ，動物を推定する．一連の判定式を Equation (2.3.1) に示す．g は推定された動物，$\mathcal{Z}$ は動物の集合，$\mathcal{M}$ は属性の集合，$\mathcal{Q}$ は特徴の種類の集合を表す．a_z^m は動物 z と属性 m の関連性を表し,「関連がある」場合には $a_z^m = pos$,「関連がない」場合には $a_z^m = neg$ とする．$[P]$ は，条件 P が真であるときは “1”，偽であるときは “0” とする．$d_{q,m}^p$，$d_{q,m}^n$ は，それぞれ Equation (2.3.2)，Equation (2.3.3) より求める．V_q は入力画像の特徴 q のパターン，$W_{q,m,t}^p(V_q)$（$W_{q,m,t}^n(V_q)$）は特徴が q で属性が m の pos-SOINN（neg-SOINN）の中で V_q と t 番目に近い重みベクトルを表す.

2.3.3 評価方法と結果

提案手法 (AT-STAR-SOINN) を，従来手法の DAP および IAP [14] と，AT-SOINN [17] との比較で評価する．提案手法のパラメータは，$\lambda = 600$，$age_{max} = 100$，$\eta = 1$，$k = 13$，$\rho = 25$ である．これらは，動物の種類を少なくした予備実験の結果に基づき決定した．なお実験した候補は，$\lambda = 200, 400, 600, 800, 1000$，$age_{max} = 50, 100, 150$，$\eta = 1, 2$，$k = 9, 11, 13, 15$，$\rho = 10, 25$ である．また，提案手法はオンライン学習であるため，学習順序により結果が異なる．そのため，以降の結果は，順序をランダムに変更し，10 回実行した平均値を記載する．

Table 2.1: 未学習動物に対する認識率の比較

Method		Accuracy(avg)
Online	**AT-STAR-SOINN**	25.72%
	AT-SOINN [17]	26.96%
Batch	DAP(SVM) [14]	**40.51**%
	IAP(SVM) [14]	27.83%

Table 2.2: 作成された SOINN の数と保有ノード数の比較

Method	# of SOINNs	# of Nodes
AT-STAR-SOINN	**6**	**1,952**
AT-SOINN [17]	1020	347,082

各々の認識率を Table 2.1 に記し，学習及び識別に所要する時間を Figure 2.9，Figure 2.10 に示す．なお，ここでの処理時間は，特徴抽出時間など全ての手法において同様に必要となる処理の時間を含めていない．次に，提案手法 (AT-STAR-SOINN) と AT-SOINN については，学習後の手法が保持するネットワークの数と，それらが保有する全ノード数を Table 2.2 に記す．また，学習画像の枚数を変化させた時の未知動物に対する認識率の推移を Figure 2.8 に示す．横軸は全ての画像を学習した場合を 100%とした学習画像の割合であり，本図は学習量が少ない時（50%以下）の認識率の変化を示している．点は実際の認識率を，実線はこれらの結果から対数近似を行っ

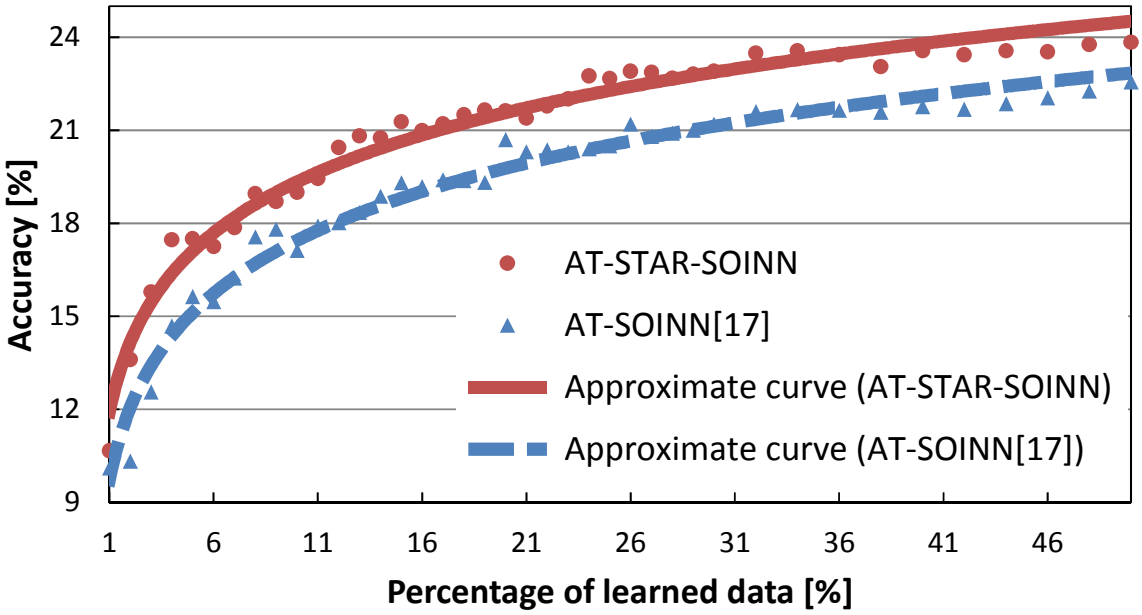

Figure 2.8: 学習画像数の割合と認識率の関係

た結果を示す．なお，実験は同一環境（CPU：2.93GHz，メモリ：8GB）で実施した．

2.4 考察

2.4.1 提案手法の評価

2.3.3 より，提案手法の認識率は，バッチで学習した DAP には劣るが（バッチ学習では実環境で稼働するロボットへの実装が不可能），従来の SOINN を用いた AT-SOINN と同等であった．そのような認識性能を保ちながら，学習時間を DAP に比べ約 13,034

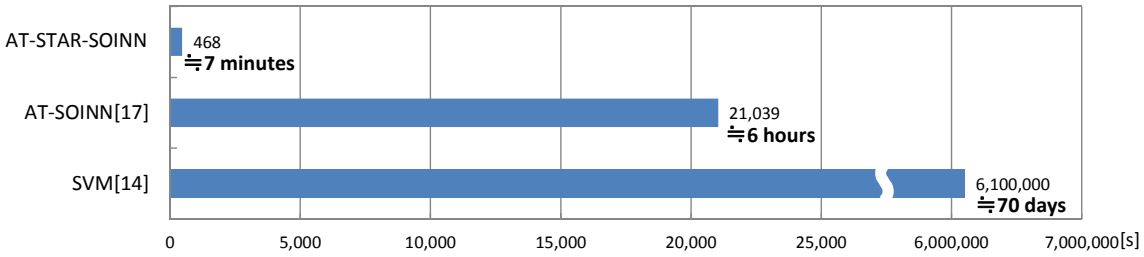

Figure 2.9: 学習時間による比較
24,295 枚の画像の学習にかかる時間（特徴抽出時間等を除く）
提案手法では 1 秒間に 52 枚の画像を学習可能

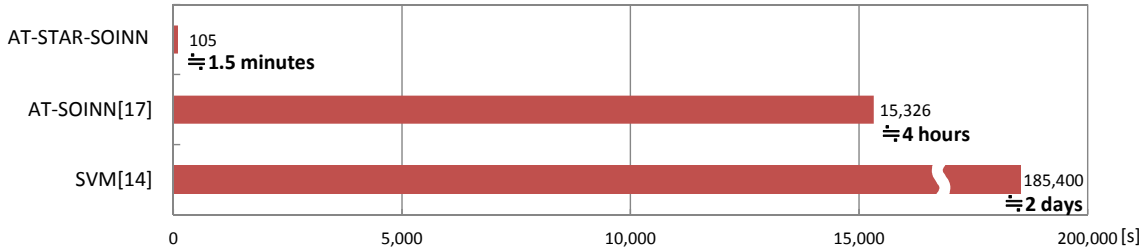

Figure 2.10: 識別時間による比較
6,180 枚の画像の識別にかかる時間（特徴抽出時間等を除く）
提案手法では 1 枚の画像を 17 ミリ秒で認識可能

倍速く，AT-SOINN に比べ約 45 倍速くすることができた．更に，認識時間を DAP に比べ約 $1,765$ 倍速く，AT-SOINN に比べ約 146 倍速くすることができた．また，ノード数を AT-SOINN の 0.56% に抑えた．つまり，提案手法は，オンライン学習である AT-SOINN と同等の認識率を保ちつつ，非常に高速で軽量な学習および認識を実現している．

従来手法 [17] と比べて，学習時間と認識時間を削減できた理由は，学習器が持つネットワーク数と大きく関係している．ネットワークの数が少ない場合，学習時と認識時共に距離計算を行う回数が少なくなる．従来手法 [14] は属性ごとにネットワークを作成し学習していたが，提案手法はネットワークに全ての属性情報を保管するラベルを導入して，ネットワーク数が属性の数に比例しない構造を持っている．そのため，ネットワーク数が Table 2.2 のように減り，高速化を実施できた．

更に，DAP [14] と AT-SOINN [17] では属性ごとに学習器を作成して学習しているが，提案手法では 1 つの学習器（ネットワーク）で全ての属性を学習している．そのため，似通った属性（正確にはパターン的に近い属性群）が存在する時，それぞれの属性を学習する際に，他の属性の学習結果を活用しながらの学習が可能である．つまり，ある属性を学習するときに，似通った属性が学習済みであれば，早く属性の学習を完了することができる．このように提案手法は，従来手法に比べて，学習効率を高めた学習が可能である．実際に，Figure 2.8 を確認すると，提案手法は早い段階で認識率が向上している．[14] については，バッチ学習であり追加学習性は無いため計測

していない．この特性は，少ない学習画像による学習を実現し，実世界のような膨大な数の属性や物体の学習が必要となる環境において，非常に有用である．

前述したような時間と情報量の削減や，学習効率の向上は，ヒューマノイド・ロボットへの実装を考慮した実世界知能に相応しい知的な情報処理を実現する基盤技術の必要不可欠な特徴である．そのため，本提案手法がこれらの知的発達機構の基盤技術の一つになりうると考えられる．

その一方で，認識率に関しては，バッチ学習手法である SVM を用いた手法 [14] と比べて低下している．この原因として考えられるのは，SOINN が持つオンライン学習性による安定性の低下が関係していると考えている．また，オンライン追加学習で動作している本提案手法は，認識精度にばらつきが存在している．これら点については，今後改良及び検討すべき課題であると考えている．

2.4.2 インターネットの活用

現在，学習データは，全て人が選別し与える必要がある．ところが，学習すべき物体数が膨大である実世界での運用を考慮した時，その選別は非常に困難となる．一方で，近年普及したインターネットを用いることで，多くの画像を瞬時に入手できる．ところが，不特定多数の人々によって提供されたものであるため，ノイズと思われる画像も多い．そこで，本提案手法の学習器 SOINN が持つノイズ耐性を用いてインターネット上のノイズを削除することで，人が学習データを準備せずに学習できる

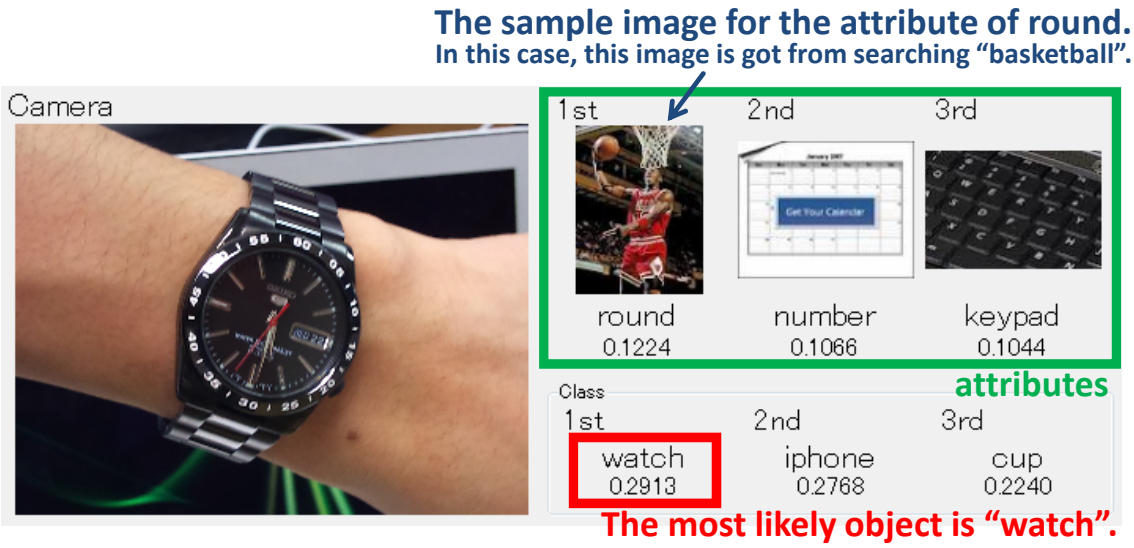

Figure 2.11: インターネット上の画像を用いた結果
未学習の腕時計をネット上の画像を用いて推定した結果
左：認識画像，右：推定結果（上が属性，下が物体）

ことを検証する．また，インターネット上のデータは数が膨大であり，本手法の高速性と軽量性も必要となる．

実際に，インターネットの画像を用いて学習するシステムを構築した．まず事前に，学習物体と認識物体のそれぞれの属性は与えられている．学習フェーズは，検索エンジンで学習物体の名前を検索しダウンロードしてオンライン学習する．認識フェーズは，カメラの前に置かれているものの属性を推定し，辞書を用いて物体を推定する．システムの概要は，Figure 2.3 と同様である．物体の認識結果と，学習や認識に所要する時間で評価を行う．

具体的な実験の内容と，その結果を以下に記す．学習物体を 25 種類，認識物体を 8 種類，属性を 18 種類定義した．次に，学習物体に対して実際に検索エンジン [2] を用いて 100 枚のサムネイル画像（低画質画像）を取得した．その画像から，画像の全体に対する pHOG 特徴量 [43] と，物体の輪郭のみの pHOG，色ヒストグラムを抽出して，それらを本提案手法を用いてオンライン学習させた．本実験で使用した特徴は，認識率よりも速度を重視して選択している．そのため，認識精度が不十分であれば特徴を再度選択することにより，より良好な結果が得られると考えられる．この時に所要した学習時間（ダウンロードと特徴抽出を含まない）は，4 秒ほどであった．次に，認識物体をカメラの前におき，そこから上記特徴を抽出した後，本提案手法の認識処理を行った．例として，認識物体である “腕時計” を見せた時のシステムの結果を Figure 2.11 に示す．図中の左図が入力画像であり，右上が推定された属性の中で関連度が高い上位 3 つの属性，右下が推定された属性から導かれた物体の尤もらしい上位 3 つである．この時，特徴抽出も含めた，全ての処理の所要時間は 0.3 秒未満であった．認識結果としては，8 種類の全ての認識物体に対して同様の認識処理を行なってみたところ，9 割以上の確率で正確に推定することができた．

本システムは，“CVPR 2014” [47]，及び “画像の認識・理解シンポジウム (MIRU 2012)” [48]，“画像センシングシンポジウム (SSII 2012)” [49] などの会議，複数回の講演や展示にて実演デモを実施した．そこで来場者の身に着けている物を瞬時に認識で

[2] Google Images (http://www.google.com/imghp)

きることを披露し，その安定性を確認した．認識率に加えて，それぞれ時期が異なっていたため，検索結果の変動に対する頑健性も確認した．このシステムの実現により，人が物体と属性に関する関係性（キーワードと，その物体に関する概念）を与えるだけで，自動的にシステムが知識を獲得して，それを用いて目の前のものを認識する基盤技術が確立できたと言える．また，物体と属性に関する関係性を自動で獲得する方法も現在検討されている [16]．このような手法と組み合わせることで，完全に “キーワード” のみでシステムが知識を獲得できる技術が実現できると考えている．

2.4.3 属性の追加と削除

提案手法は，属性の追加が容易に可能である．従来手法の AT-SOINN や DAP などで属性を追加させる場合には，一から新しく学習器を作成する必要がある．一方で，提案手法では SOINN が保有しているラベルの追加で対応する．また，類似した学習データに関する既存のネットワークが存在すれば，ラベルの拡散により瞬時に対応できる．これは，新たに学習する属性とパターン的に似通った属性が，既に学習済みである時に効果的である．ある知識を受けると，ネットワーク上で繋がっているノードに対してもラベル情報を拡散する．つまり，新たに学習する属性の知識を与えられた時，類似の属性により構築されたネットワークを用いてラベルを拡散することで，少ない学習回数で早く学習が完了する．

Figure 2.4.3 に具体的な例を示している．上図のような属性を学習すると仮定し，

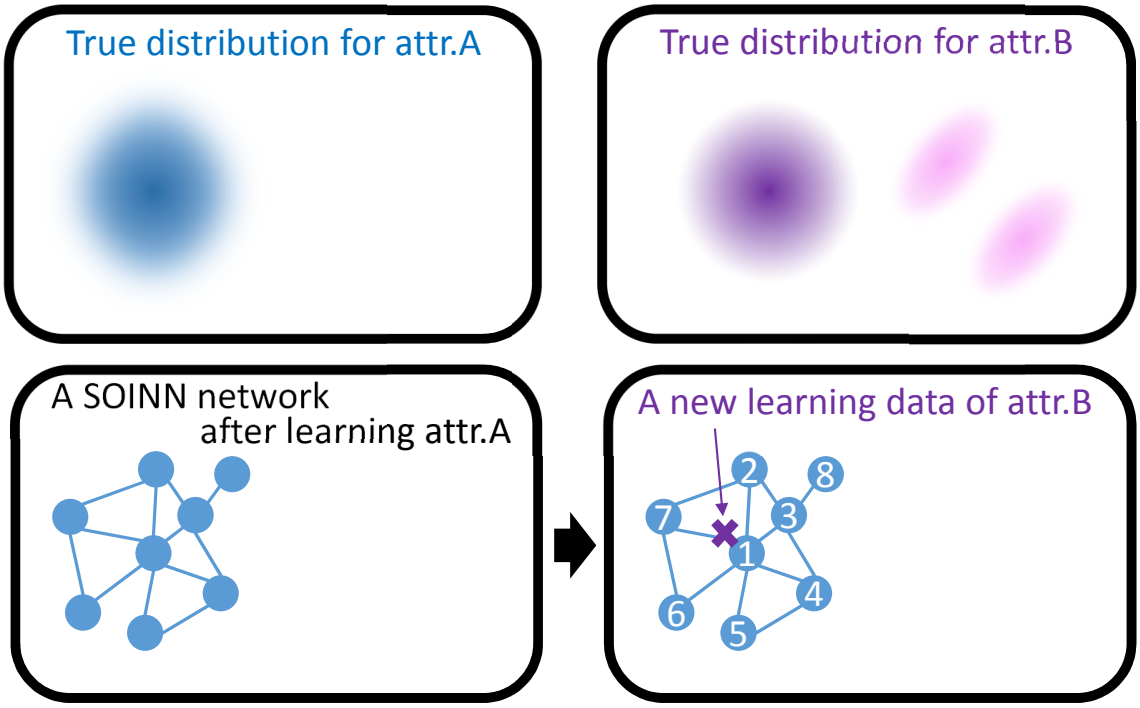

Figure 2.12: ラベル拡散による効果の例
上図：属性に関する真の確率密度（濃淡で強弱）
下図：属性 A を十分学習した後に属性 B を新たに学習した場合の SOINN

現在左下図のようなネットワークを SOINN が保持しているとする．ここで，新たな属性 B について学習するために，右下図のような入力があったとする．すると，まずノード 1 に対して属性 B がポジティブである情報が学習される．その後，ラベル拡散により，ノード 2 から 7 に対しても同じようにポジティブである情報が伝播される．つまり，ノード 2 から 7 の近傍には入力がされていないにも関わらず，学習が可能となる．属性 B の右側の小さな 2 つの分布については，事前にネットワークが形成され

ていないため，一から学習する必要が存在する．このように，ラベル拡散を行うことにより属性の追加に対して早期に対応可能である．なお，その一方で，この効果の副作用として，パターン的に似ている場所に異なった構造（ネットワーク）を持つ知識を学習しにくいという点が存在する．この点が，学習器を重ねた時の難しさであると考えられる．

また，重複した属性の削除も容易に可能である．2 つの属性の重複度を調べるとき，2 つの学習器（AT-SOINN では 4 つのネットワーク：それぞれの pos-SOINN と neg-SOINN）を参照する必要がある．ところが，提案手法では 1 つの SOINN が保有しているラベルの類似度を調べることのみで対応ができ，少ない計算コストで実行可能である．属性を減らすことで，過学習の抑制や認識時間の短縮，情報量の削減などが期待できる．

2.4.4 曖昧性のある属性

提案手法には，従来手法の [17] と比べて属性の定義の仕方に柔軟性がある．具体的には，動物と属性の関連度を連続値で定義可能である．従来手法では，属性に対して “関連あり” と “関連なし” のそれぞれについてネットワークを作成するため，連続値の属性を扱うことが出来ない．一方で提案手法では，連続値で定義された属性をそのまま扱うことが可能である．また，[14] では回帰手法を用いることで対応可能であると述べられている．しかし，回帰処理を用いることで計算時間の増加が懸念され

る．一方で，提案手法では，提案したネットワーク構成でそのまま対応可能である．なお本稿の比較実験では，比較のために，二値を利用した．連続値による属性の定義は，詳しい物体の定義に繋がり，複雑な実世界の物体に対して特に重要であると考えられる．

2.4.5 今後の課題

今後の課題として，SOINN や STAR-SOINN の理論的な解析が挙げられる．本稿では，SOINN が理論的にどのようなネットワークを作成しているのか，作成されたネットワークをどのような理論で活用しているのかまでは言及していない．著者らが最も効果的であると考えるネットワーク構築方法を用いている．そのため，今後 SOINN の理論的な解析が必要であると考えており，それにより本手法の適用可能範囲や更なる改善点などの発見に繋がり，認識率の向上も可能であると考えている．

最後に，提案手法と従来手法が持つ特徴について Table 2.3 にまとめる．

2.5 まとめ

本章では，オンラインで追加学習可能な SOINN に教師情報に関する統計情報を算出する拡張を行い，新しい STAR-SOINN を提案した．次に，STAR-SOINN を用いた属性ベースの軽量な自己転移手法を提案した．この手法を用いることで，従来手法に

Table 2.3: 画像の自己転移に関する各手法の比較

Method	**Proposed**	[17]	[14]
Online learning	○	○	×
Speed of learning and test	○	△	×
Transfer learning	○	×	×
Easy to add and delete attributes	○	×	×
To define a real valued attribute	○	×	△

比べて，非常に高速な学習と認識を実現した．更には，SOINN が持つノイズ耐性により，インターネット上のデータも扱うことができ，人が学習データを準備せずに学習する知的発達の基盤手法を構築した．また，柔軟な属性の追加や削除や，曖昧な属性の定義についても議論した．

Chapter

3

マルチモーダル情報を用いた知識転移

3.1 背景

前章では，画像情報を活用した自己転移に関する手法を提案した．一方で，ロボットは画像以外にも多数のセンサを搭載しており，それらの活用も人と同様の生活環境下で稼働するロボットには必要不可欠なモダリティとなる．すなわち，ロボットが保有する視覚情報や触覚情報などの様々なセンサ情報を活用することで，人と同様の感覚を扱い，物体を総合的に判断する処理基盤も必要となる．

マルチモーダル情報を用いた物体認識に関する研究は，様々な研究機関で実施されている [50–54]．しかし，Haasch らの研究 [50] や Nakamura らの研究 [51] は，既知物

体の認識を実施しており，未知物体については識別が出来ない．Araki らの研究 [52,53] は，マルチモーダル情報を用いたオンラインの識別を実施している．ところが，彼らが用いている ‘未知物体 (Unseen object)’ は，未知物体と同じカテゴリーの物体を学習しているなど，学習物体と非常に似通っている．例えば，ペットボトルで例えると，多数のペットボトルを学習している時に，学習していない種類のペットボトルを推定することは可能であるが，全く学習していない状態から他の物体で獲得した知識を用いてペットボトルという未知物体を推定することは不可能である．更に，Ozasa らの研究 [54] は，学習していない物体（未知物体）をマルチモーダル情報を用いて判断する機構を実現している．しかし，未知物体を “未知物体” として認識するのみに留まっており，ロボットは未知物体が目の前にあることしか認識することができない．

人の生活環境にある物体は，そのほとんどが未知物体であり，それらが持つ「属性」だけでも推定することが出来れば非常に有用であると考えられる.「属性」とは，物体の大まかな特徴や特性などの物体が持つ概念を意味し，例えば “形が直方体である” や “金属でできている”，“壊れやすい” などが挙げられる．前章で述べたとおり，画像を用いた属性情報の転移手法を提案している．ところが，この従来研究では画像のみを対象にしており，様々な種類のセンサからの属性推定の結果を統合するような手法は取り入れていない．

そこで本稿では，実機のロボットからマルチモーダル情報を取得し，自己転移を行い属性を推定可能な手法を提案する．Figure 3.1 に，本研究の概要を示す．また，Table 3.1

Table 3.1: マルチモダール情報の自己転移に関する他手法との比較

Method	Real-world	Unknown object	Multimodal sensor
[50]	○	×	○
[51]	○	×	○
[52]	○	△ (not attribute)	○
[54]	○	△ (only detection)	○
[17]	×	○	×
Sec. 2.2 in this thesis	○	○	×
Proposed method	○	○	○

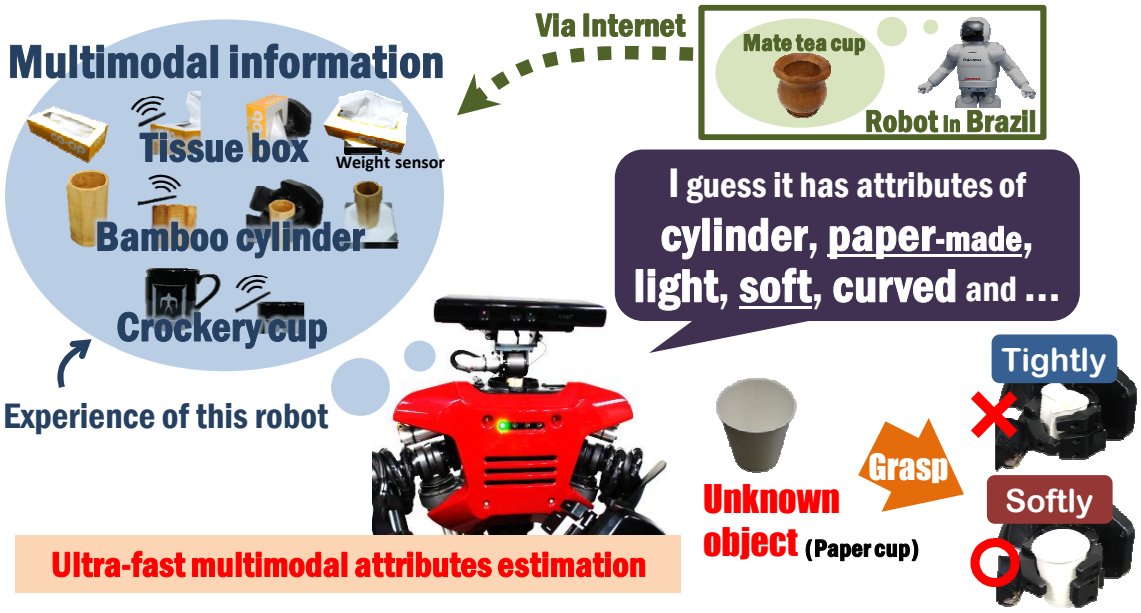

Figure 3.1: マルチモーダル情報を用いた知識転移の概要

に，提案手法と他手法との比較を示す．“Real-world” は実際に実時間で稼働できるのかを示し，“Unknown object” は未知物体の属性を識別できるかを示し，“Multimodal sensor” はマルチモーダル情報を活用できるのかを示している．Figure 3.1 では，未知物体である紙コップを把持する際に，事前に「柔らかい」「紙製」という属性を過去に学習した他の物体の知識より推定している．その結果，紙コップを “そっと”持つことを選択している様子を示している．本研究の手法を用いることで，こうしたロボットの繊細な動作の選択や，未知物体の特定が実行可能となる．また，この研究では，対象としているロボットの経験に限定せず，インターネット等のネットワークを経由することで他のロボットの経験も活用することを想定している．

それでは提案手法を具体的に記述する．本提案手法は，前章で提案した手法に対して，様々なマルチモーダル情報を “総合的に” 活用できる処理機構を提案し導入している．“総合的に”とは，属性ごとにそれぞれのモダリティの特性を把握し，それを考慮した推定を意味する．これは，モダリティにより判別できる属性が大きく異なり，モダリティと属性の関係度が重要であるため導入する．例えば，形の判別には視覚情報が重要であり，聴覚情報は参考になり難いなどが挙げられる．この関連度を自ら算出して，その度合いに基づき，各々のモダリティから得られた属性の度合いの利用度合いを調整する機構を導入する．

次に，実際のロボットからマルチモーダル情報を取得し，それを用いて提案手法の有効性を定量的に評価する．また，実際に未知物体の属性を推定させる実験を行う．最後に，ロボットの基盤技術としての可能性や，インターネットの活用などについて考察する．

本章の構成は，次の通りである．まず 3.2 節で提案手法について述べる．3.3 節で実際のロボットを用いた評価，3.4 節で考察と今後の課題について述べる．最後に，3.5 節で本章のまとめを述べる．

3.2 提案手法

提案手法は，従来の画像を用いた学習手法を，性質の異なる複数のセンサ情報の活用へと拡張した手法である．この時，従来手法と大きく異なる点は，それぞれの属性に対して各モダリティの関連度（判別しやすさ）を学習時に算出する機構と，その度合いを識別時に活用する機構を追加している点である．この関連度を用いて，属性の種類ごとに属性値の判別に利用するモダリティを変更して，複雑で多種多様な特性を持つ属性に対応させる．

Figure 3.2 が提案手法の概略である．本手法（システム）は 3 層からなり，1 層目が特徴量を抽出する入力層，2 層目が属性と特徴パターンの関連性を学習器により記憶するパターン層，3 層目が推定された属性を用いて物体の特定や動作の生成を行うシンボル層である．以下に学習フェーズと，認識フェーズ，物体の特定方法の詳細を記述する．なお，本節の以降に出現する文字の意味を Table 3.2 に記す．

3.2.1 学習フェーズ

まず，ロボットから採取される学習物体の画像や音などの感覚情報や，インターネットから取得する学習物体に関する情報から特徴量を抽出する．次に，抽出された特徴を STAR-SOINN に入力する．この時，学習する物体に関する属性をラベル情報として付加する．ラベル情報 L^m は，属性 m が真の場合には 1，偽の場合には -1，実

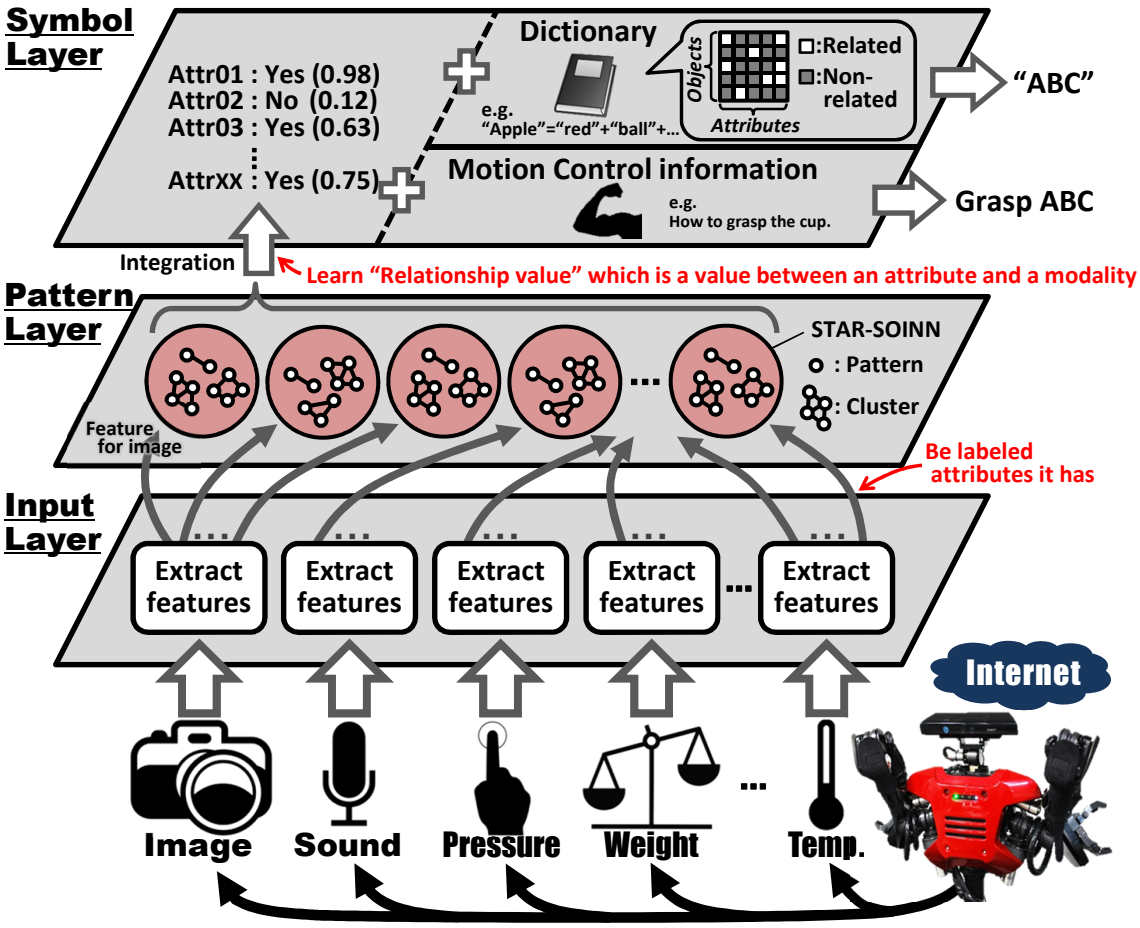

Figure 3.2: マルチモーダル情報を用いた知識転移に関する提案手法の概要図

Table 3.2: 本章での記号の意味

Letter	Meaning
$\mathbb{X}$	All inputed and extracted features
$\mathbb{F}$	All types of feature
$\mathtt{num}\,(\mathbb{A})$	Number of $\mathbb{A}$
$\mathbb{N}_f^k\,(\mathbb{X})$	k-nearest nodes around extracted feature for feature f in *SOINN(=STAR-SOINN)
$L_{f,n}^m$	Label weight of attribute m on node n in *SOINN for feature f
$\mathbb{O}_t$	All training objects
$\mathbb{O}_\alpha$	The objects until object α
$\mathbb{X}_c$	All extracted features for object c
b_c^m	Value of relationship between object c and attribute m (“positive”$= 1$, “negative”$= -1$)
k, α	Parameters of this method
$\mathbb{O}$	All objects
$\mathbb{M}$	All attributes
b_o^m	Value of relationship between object o and attribute m by dictionary

数値（-1 以上 1 以下）の場合はその値を用いる．

そして，定数 α 個目以降の物体の学習時には，後述する認識フェーズの処理を学習前に実施し，その後に物体の学習を行う．この時に記録された認識結果を用いて，属性とモダリティの関連度を算出する．その詳細は，次節で述べる．

3.2.2　認識フェーズ

まず，ロボットから取得された識別物体に関する複数の感覚情報から，それぞれの特徴量を抽出する．それらの特徴 $\mathbb{X}$ と，それぞれの STAR-SOINN 内のノードとの距離を算出し，k 個の近傍ノード $\mathbb{N}_f^k(\mathbb{X})$ を求める（*k-NN*）．それら k 個のノードのラベル情報を用いて，特徴 f に対する属性 m の属性値 $A_f^m(\mathbb{X})$ を Equation (3.2.2) を用いて算出する．なお $A_f^m(\mathbb{X})$ の値域は，$-1 \leq A_f^m(\mathbb{X}) \leq 1$ である．その後，それぞれの特徴より求めた属性値 $A_f^m(\mathbb{X})$ と R_f^m を用いて，総合的な属性値 $A_{all}^m(\mathbb{X})$ を Equation (3.2.1) で算出する．これは，$A_f^m(\mathbb{X})$ の R_f^m による重み付き平均である．R_f^m は，属性とモダリティの関連度を表しており，学習フェーズでの定数 α 個目以降の学習物体に対する認識結果と，Equation (3.2.3)-(3.2.5) を用いて算出する．この値は，学習フェーズ時に認識処理を行った物体に対して，属性の真値（真か偽）を用いて属性の推定値を分類し，真と偽のそれぞれの平均を求め，その差分を意味する．なお，真値が実数値の場合には，その度合いだけ影響を及ぼす．そのため関連度 R_f^m の値が大きい程，用いた特徴 f は属性 m を識別しやすいことを示す．

$$A_{all}^{m}\left(\mathbb{X}\right) = \frac{\sum_{f \in \mathbb{F}} A_{f}^{m}\left(\mathbb{X}\right)\ R_{f}^{m}}{\mathtt{num}\left(\mathbb{F}\right)\ \sum_{f \in \mathbb{F}} R_{f}^{m}} \tag{3.2.1}$$

$$A_{f}^{m}\left(\mathbb{X}\right) = \frac{1}{k} \sum_{n \in \mathbb{N}_{f}^{k}(\mathbb{X})} L_{f,n}^{m} \tag{3.2.2}$$

$$R_{f}^{m} = T_{f}^{m,p} - T_{f}^{m,n} \tag{3.2.3}$$

$$T_{f}^{m,p} = \frac{1}{2 \cdot \mathtt{num}\left(\mathbb{O}_{t} \setminus \mathbb{O}_{\alpha}\right)} \sum_{o \in \mathbb{O}_{t} \setminus \mathbb{O}_{\alpha}} A_{f}^{m}\left(\mathbb{X}_{o}\right) \cdot \left(1 + b_{o}^{m}\right) \tag{3.2.4}$$

$$T_{f}^{m,n} = \frac{1}{2 \cdot \mathtt{num}\left(\mathbb{O}_{t} \setminus \mathbb{O}_{\alpha}\right)} \sum_{o \in \mathbb{O}_{t} \setminus \mathbb{O}_{\alpha}} A_{f}^{m}\left(\mathbb{X}_{o}\right) \cdot \left(1 - b_{o}^{m}\right) \tag{3.2.5}$$

以上が属性値の算出処理であり，その後，属性と物体の関連度を示した辞書を用いた物体の特定や，属性値に合わせたロボットの動作[1]などを行う．本研究においては，物体特定を実施しており，その詳細については後述する．

[1]紙製の中が空洞な筒（紙コップなど）は優しく握る

3.2.3 辞書を用いた物体の特定

既知の物体である場合，その物体が持つ属性の一覧を得ている．また，未学習の物体であっても，その物体概念のみを受け取ることは容易に実行可能である．例えば [16] では，その物体概念をネット上の文書データより自動で獲得する仕組みを構築している．

ここでは，上記の方法より物体と属性の関係性を記述した辞書を得られると仮定して，物体の特定方法を記述する．得られた辞書と，3.2.2 の認識フェーズより算出された属性値 $A^m_{all}(\mathbb{X})$ を用いて Equation (3.2.6) で物体を特定する．j には，特定された物体の番号が代入される．

$$j = \underset{o \in \mathbb{O}}{\operatorname{argmax}} \sum_{m \in \mathbb{M}} A^m_{all}(\mathbb{X}) \cdot (1 + b^m_o) \tag{3.2.6}$$

3.3 実験と評価

提案手法の有効性を定量的に検証する実験と，未知物体の属性を推定する実験を行い，本提案手法を評価する．

3.3.1　有効性の検証

ロボットが持つ様々なモダリティのセンサよりデータを取得し，既知物体に対する認識率と学習及び認識の所要時間を評価対象として定量的に提案手法を評価する．

3.3.1.1　実験環境

本研究では人の生活環境下での稼働を目指しているため，全ての実験データは，日常環境で，人が実際に使用している物体から採取した．実験で使用した 20 種類の物体を Table 3.3 に，使用したロボットや，センサ，物体の外見を Figure 3.3 に示す．Table 3.3 中の属性については，表を見やすくするために種類別に分けて記載したが，手法が種類に関する情報は得ていない．用いたモダリティ情報は，物体を様々な方向から撮影した画像，物体を叩いた時の音，物体表面の立体形状，物体を把持した時の圧力，物体の重量の 5 種類である．具体的には，Figure 3.4 のように物体を机に置きカメラを向け全方向から撮影された画像と，物体を叩いた時にマイクより取得される音，物体を深度センサの正面に掲げた時に取得される立体形状，圧力センサより取得される物体を把持した時に受ける反発力，重さセンサより取得される物体の重量である．それぞれ，1 つの物体につき 60 枚，25 秒，100 枚，30 回，30 回取得した．なお，紙やカッターなど物理的制約によりロボットが把持できない物体の圧力は，指と物体を接触させ，その時の反発力を代用している．特徴量はそれぞれ，物体の輪郭に対し

Table 3.3: 本実験で使用する物体と属性の一覧

Object	Aluminum can, Steel can, Paper cup, Cleaning tool, Bamboo cylinder, Japanese tea cup, Plastic cup, Crockery cup, Cutter, Sponge, Bin, Pet bottle,Paper cylinder, Glass, Paper, Clear plastic cup, Paper dish, Whiteboard marker, Wood, Book	
Attribute	Shape	Cuboid, Triangle pole, Cylinder, Plane, Stick, Spray, Dish
	Material	Metal, Wooden, Crockery, Glass Paper made, Styrofoam, Plastic
	Surface	Clear, Flat, Curved
	etc.	Fragile, Irrefrangible, Handle, Letter, Hard, Soft

Figure 3.3: 使用したロボットや，センサ，物体の外観

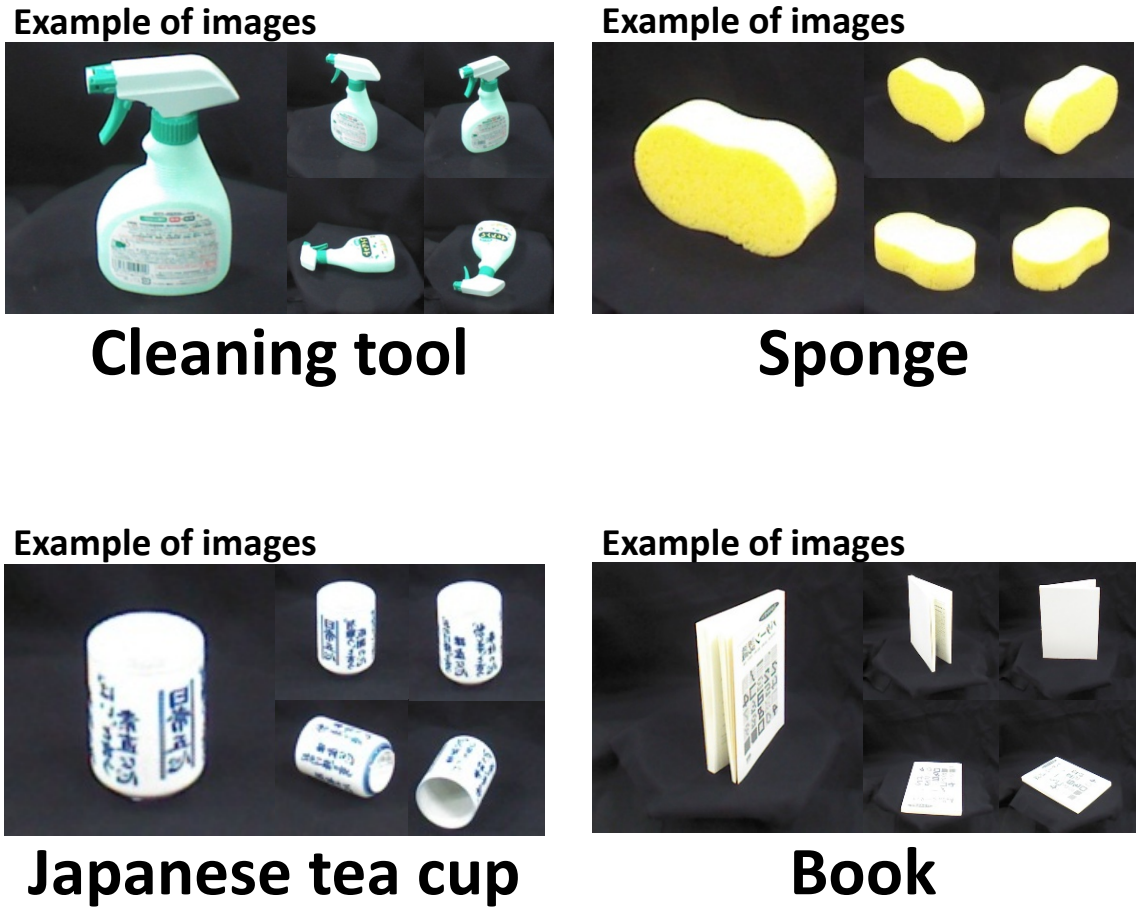

Figure 3.4: 使用した画像情報の一例

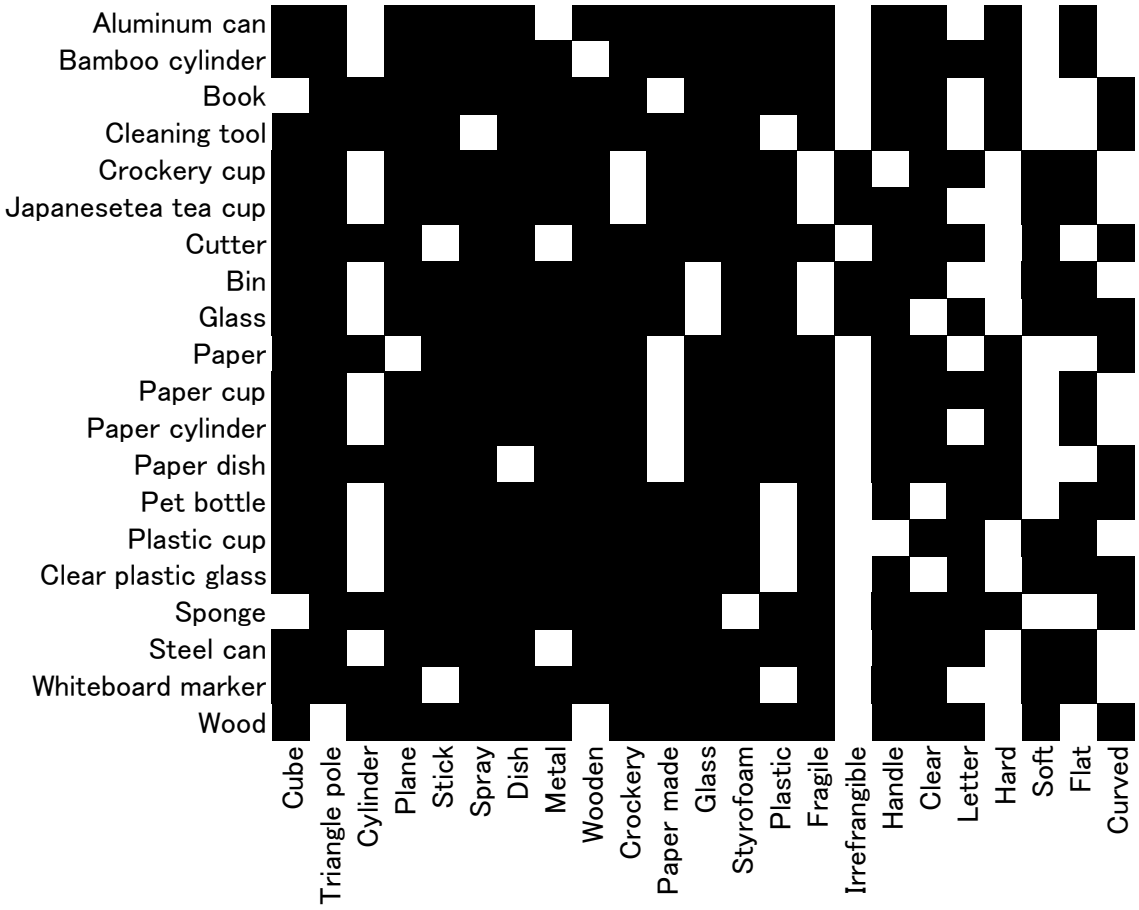

Figure 3.5: 本実験で用いた物体と属性の関連度
「白」は “関連あり”,「黒」は “関連なし”

ての 60 次元の pHOG [43] 特徴量（HOG [55] 特徴量をベースとしている），FFT [56] により求められた 24 次元のパワースペクトル，立体形状の法線ベクトルに対する 60 次元のヒストグラム，反発力の 2 次元の実値，重量の 1 次元の実値を用いた．この時，本研究では実稼働を目指しているため，精度よりも特徴抽出の速度を重視し特徴を選択している．例えば，画像の特徴量については，アフィン変換はもちろん画像の回転に対しても頑健性を持たない特徴を用いている．また，以降の評価は，$K = 5$ のクロスバリデーション（交差検定法）[57] を用いてデータを分け，組み合わせを変えて 20 回実施し，その平均を記述している．

物体の属性は Table 3.3 に示す 23 種類の属性を利用し，実際に当てはまる属性を Figure 3.5 に示す．学習器 STAR-SOINN が持つパラメータ λ，age_{max}，η はそれぞれ 650，100，2 とし，提案手法が持つパラメータ k，α は，3，9 とした．これらは物体数を少なくした小規模な予備実験により決定した．また実験は全て，CPU が 3.00GHz，メモリが 8GB の環境で行なっている．

3.3.1.2　実験結果

Table 3.4 は，手法と使用するモダリティを変更した時の認識率の遷移を示している．ここでの認識率とは，Equation (3.2.6) より得られた j と，正解となる物体の番号の一致率を意味する．モダリティの種類が n 種類の場合，n 個のモダリティより得られた n 個の属性値 $A_f^m(\mathbb{X})$ を用いて物体を識別することを表している．“Naïve adding”

Table 3.4: 各手法及び各モダリティによる認識率の比較

Methods (modalities)	Accuracy
Proposed (5 modalities)	**94.52%**
Naïve adding (5 modalities)	79.12%
Proposed (Image, Sound, Depth)	90.48%
Naïve adding (Image, Sound, Depth)	77.76%
Proposed (Image)	53.15%
Proposed (Sound)	73.63%
Proposed (Depth)	68.40%
Proposed (Pressure)	41.25%
Proposed (Weight)	42.11%

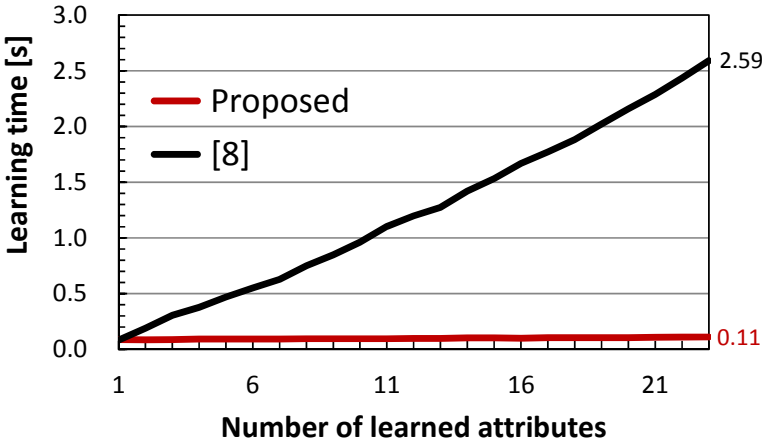

Figure 3.6: 学習時間の増加に関する比較

とは，関連度 R_f^m を全て 1 として，各モダリティから得られた属性値を単純に平均する手法を意味している．この結果より提案手法は，モダリティの増加へ対応し，単純にモダリティを合成する手法 “Naïve adding” よりも高い認識率を持っていることが確認できた．

Table 3.5 は，本実験の学習と認識に所要する時間を示している．この手法は実運用を想定しているため，特徴抽出の時間も考慮している．Figure 3.6 は，属性の増加に対する特徴抽出を除いた学習時間の増加を [17] と比較したグラフを示している．この結果より提案手法の処理速度は，従来手法より高速で，且つロボット上で十分に運用できる程度であると確認できた．

Table 3.5: 提案手法の計算時間
（特徴抽出時間 ＋ 学習時間 もしくは 識別時間）

1 object learning	48 imgs	227 + 32.1 [ms]
	20[s] sounds	206 + 6.84 [ms]
	80 depth-infos	774 + 69.3 [ms]
	24 prs.-infos	0 + 2.56 [ms]
	24 weights	0 + 1.34 [ms]
	Total	**1.32 [s]**
1 time recognition	An image	4.73 + 1.06 [ms]
	A sound	8.27 + 0.73 [ms]
	A depth-info	8.92 + 1.07 [ms]
	A prs.-info	0 + 0.214 [ms]
	A weight	0 + 0.158 [ms]
	Combine	0.173 [ms]
	Total	**26.3 [ms]**

3.3.2　未知物体の属性の推定

次に，未知物体の属性を推定させる実験を実施し，本手法の有用性を確認する．

3.3.2.1　実験環境

3.3.1 節の実験で学習したシステムを用いて，学習していない未知物体の属性を推定させる．未知物体として，マテ茶のカップと，ホワイトボードイレーサー，ティッシュペーパーの箱の 3 種類の物体を用意した．これらの物体よりマルチモーダル情報を収集し，1 つの物体につき 20 組のテストデータセット作成した．このデータセットを用いて，属性の推定を実施する．

3.3.2.2　実験結果

Figure 3.7 は，マテ茶のカップとホワイトボードイレーサーの推定結果の一例を示している．棒グラフ中の色はモダリティの種類，バーの長さは属性値の大きさ，赤枠は正確に推定された属性を表している．マテ茶のカップの推定結果については，妥当な属性について高い属性値を示しており，全て正しく答えている．一方でホワイトボードイレーザーについては，“直方体” という属性を答えられてない．その理由は，“直方体” の学習物体が「スポンジ」と「本」であり，どちらもホワイトボードイレーサーに似ていないことが原因であると考えられる．これより，学習した物体と大きく

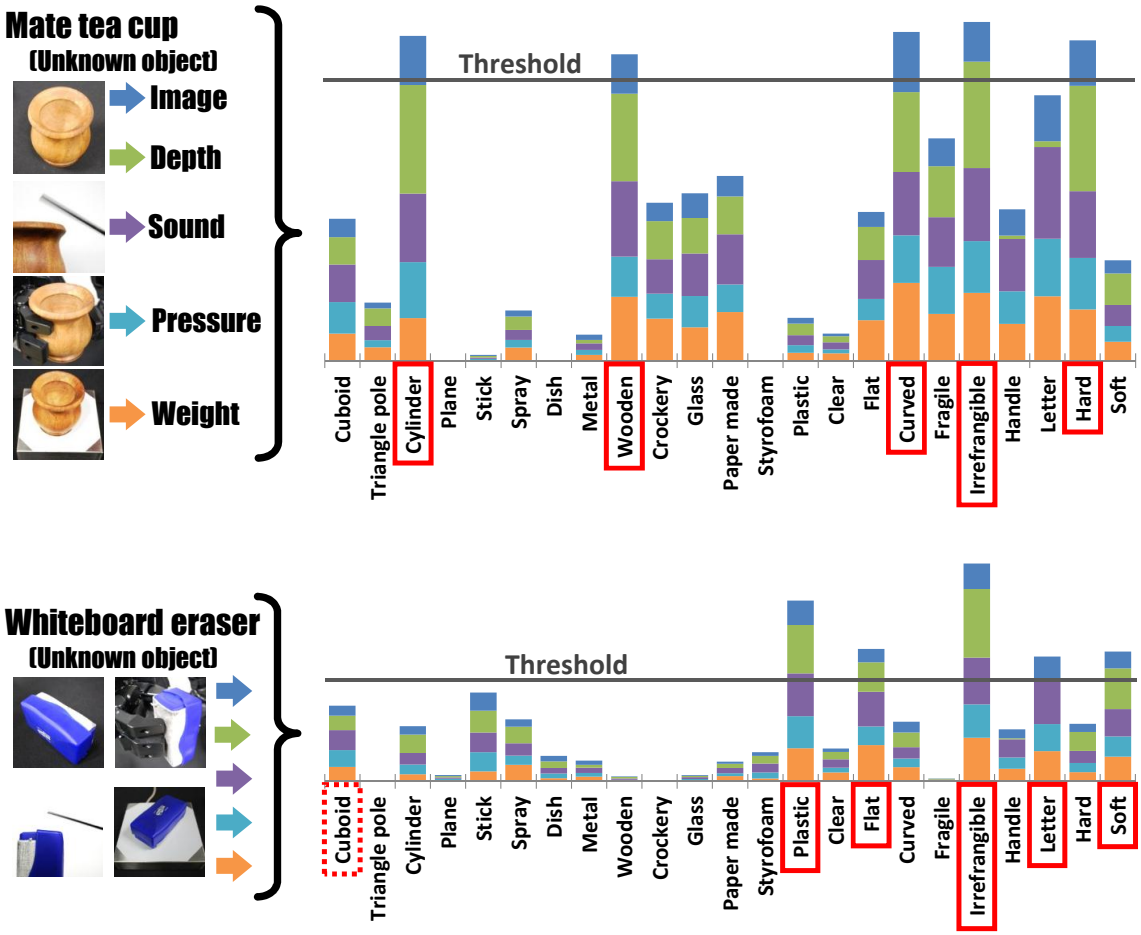

Figure 3.7: 未知物体に関する推定結果

異なった形状や特徴を持っている未知物体については，推定が困難であると考えられる．なおこの時，属性の実数値から真偽値に変換するために，全ての物体における閾値を 1 つ設定している．また，全ての未知物体の属性に対する平均の認識率は 89.3% であり（Precision : 82.9%，Recall : 71.2%），認識にかかる時間は約 25[ms] であった．

3.4　考察

3.4.1　ロボットの基盤技術となる可能性

本手法は，STAR-SOINN のみを学習器として使用している．そのため，使用した学習器が 1 種類であり，使用している技術に統一性や一貫性が生じている．このような特性は，ロボット等の基盤技術として理想的であると考えられる．その理由は，モダリティにあわせて学習器の設計をし直す必要がないためである．すなわち，統一性がない手法に比べて，モダリティの追加が容易に実行可能である．

更に，マルチモーダル情報を活用することで，より人の感覚に近い物体の分類を行うことが可能となる．例えば，陶器のコップと紙コップを持ち上げる時の握る力の大きさは変化している．ロボットが物体の把持動作の前に，物体が持つ属性を推定できることにより，このような繊細な動作を実現することができると考えている．実際に本手法を用いることで，紙コップを掴む前に柔らかいことを認識し，力を調整する機構を実現できた．

3.4.2　インターネットの活用

本提案手法の基盤となっている学習手法は，前章で記述したとおり，インターネット上の画像を検索し，検索より得られたノイジーな画像群から物体形状などを学習し，未知物体が持つ属性を画像から推定する物体認識を実現している．この時，これらの実験が “画像”で実施された理由は，現在ネット上で利用できる学習データが画像しかないためである．ところが，今後のロボット技術の発展を考慮すると，他のモダリティの学習データも後に公開されていくと考えられる．その時，ネット上のノイジーな画像で物体認識を実現した提案手法であれば，これから公開されると考えられるノイジーなマルチモーダル情報にも即座に対応できると考えられる．

3.4.3　今度の課題

未知物体の属性を判別する際，属性の度合いから真偽値に切り替える閾値が必要となる．本稿では，人手で閾値を決定した．閾値の決定方法としては，属性の数を先に決定して属性を選択する方法や，形に関する属性は 1 つしか選ばないなどの属性同士の制約や関連性を用いて選択する方法などが考えられる．閾値に関しては，今後も検討すべき課題であると考えている．

本手法では学習する物体の属性は，事前に人が教える必要がある．一方で，マルチモーダル情報の特徴と属性は非常に密接に関係している．そのため，属性に関す

る教師データを必要とせずに，学習されたパターン情報からクラスタリング手法を用いることにより，属性を発見することができる可能性が非常に高い．例えば中村らは，8 つのカテゴリで，マルチモーダル情報からカテゴリを推定する手法を提案している [51]．このような推定手法と組み合わせることで，ロボット自身がマルチモーダル情報から属性を発見して，その属性を学習し，識別物体に対して属性推定を実施する手法を構築できると考えている．

3.5　まとめ

本章では，マルチモーダル情報に基づく属性を活用した自己転移手法を提案した．実際にロボットへ実装し，実機から得られたモダリティ情報を用いて，定量的に提案手法の有効性を確認した．更に，実際に未知物体をロボットに与えると，その物体が持つ属性を瞬時に推定した．これより，本手法を用いることで，マルチモーダル情報を用いた現実的なロボットの物体認識を実現できると考えている．そして，本手法がロボットの基盤技術になる可能性と，インターネットの活用などについて考察した．

Chapter

4

動作情報を用いた知識転移

4.1 背景

近年のロボット分野の技術発展は活発で，一般家庭へロボットが登場する日も近いと考えられる．ところが，ロボットのための知的発達機構については，未だ研究段階にあり，現在実行できる動作や理解できる事柄は制限されている．ロボットが広く普及すると，様々な企業などから多種多様なスペック（特性）を持つロボットが開発されると考えられる．そのような環境にて，ロボットに行動を学習させるとき，機体の特性がそれぞれ異なっているため，ロボット一つ一つに対して個別に学習させる必要がある．ところが，一般家庭へ導入されているロボットに対して独立に学習させる

ことは非現実的だと考えられる．そこで，ロボット同士で学習した知識を共有出来る機構が必要となる．

一般に，ロボットが保有する視覚センサや聴覚センサなどについては，同様の特性を持つセンサを用意，またはキャリブレーションを行うことで情報共有が可能である．その一方で，ロボットの動作情報については，ロボットの物理的な特性（例えば，腕の長さなど）が重要となり，それらの特性を同一化することは困難となる．そのため，他のロボットから情報を得たとしても，それを自らの特性に合わせる機構が必要となる．そこで本稿では，ロボット間での動作情報の共有について議論する．

ロボットの動作情報の学習に関して近年盛んに研究されている [58–63]．従来，ロボットの動作を学習する際には，ロボットのモデルを取得して，そこから動作を生成する手法が一般的であった．ところが，事前にモデルを測定するなどの準備が必要となるため，近年，経験ベースで学習する手法が増えてきている [58]．また，モデルが変更となった場合や，測定に誤差が生じていた場合には，正確な動作が出来ないなどの問題点もある．更に，人や動物が動作を学習する方法を考えたとき，事前に自己の体のモデルを得ていない．これらの理由から，モデルを事前に用意しない手法のほうが現実的であると考えられる．

そこで本稿では，経験ベースで学習した動作情報に関するロボット間での転移（他機転移）手法を提案する．その後，手法を定量的に評価し，実機を用いて転移について実証実験を行う．なお “転移” とは，転移元（ソースドメイン）のロボット上で獲

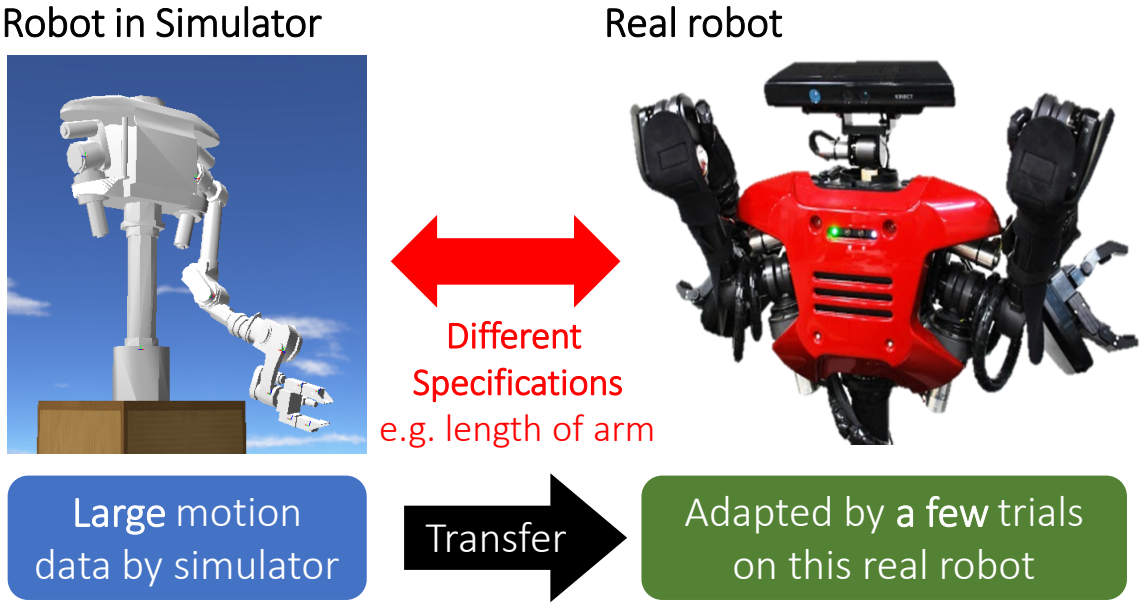

Figure 4.1: 動作情報を用いた知識転移の概念図

得した動作サンプルを，転移先（ターゲットドメイン）のロボットへ共有して，転移先のロボット上で転移元で学習した動作を獲得することを意味している．

そして，更に，転移元のロボットは，汎用的なモデルで構築された耐久性があるロボットとし，転移先のロボットは，一般家庭に設置されたロボットとする．これは，転移元のロボットでは十分な動作の学習を行うことができ，転移先のロボットでは少量の学習しか出来ないことを意味している．このような制約の中，省コストで転移を実現出来る手法を提案する．本稿で扱う問題の概念図を Figure 4.1 に示す．

従来研究として，人がロボットに手を取りながら教える様々な研究 [59] や，人の動作をロボットへ転移させる研究 [60–62] などが挙げられるが，どれもロボット間の

Algorithm 1 SOM の更新アルゴリズム

Require: **W**:weight vectors of all nodes, **I**:all input vectors, t:current iteration, $\alpha(t)$:learning rate by t, $\theta(t, x, y)$:learning rate by t and the distance between x and y

1: $t \leftarrow 0$
2: **for all** $input \in$ **I do**
3: $w_{nearest} \leftarrow$ vector of the nearest weight from $input$
4: **for all** $w \in$ **W do**
5: $w \leftarrow w + \alpha(t)\, \theta(t, w, w_{nearest})\, (input - w)$
6: **end for**
7: $t \leftarrow t + 1$
8: **end for**

転移を実現していない．また，Bocsi らは，同様にロボット間での知識転移を実現しているが，転移に関しては線形変換を実施しており，更に特に主眼を置いているのは，PCA 等の次元圧縮による関節数の違いの克服である [63].

さて，競合学習 (Competitive learning) の一種として，ネットワーク構造を用いて，入力データに対して非線形に変形を行う手法である自己組織化マップ (Self Organizing Maps: SOM) が Kohonen らによって提案されている [26,27,64]．SOM には様々な応用や活用方法が存在しているが，基本的な SOM は，作成したニューロンを入力に従ってトポロジーを変形させる機構を持つ．その時に活用される SOM の更新式は，入力に対して近傍ニューロンを更新する構造を持っており，今回の知識転移において活用できる可能性があると考えられる．具体的には，転移元で構成した SOM のネットワー

クを，転移先で獲得した動作情報を活用して更新することでネットワークの構造を非線形に変形する．ここで SOM が持つ更新アルゴリズムを Algorithm 1 に示す．

4.2 提案手法

4.2.1 仮定

本章では，次の条件下にある転移元と転移先の 2 種類のロボット間での動作情報の転移手法を提案する．

(1) ロボットが持つ関節の数は共に等しい

(2) 関節数以外の物理特性は未知

(3) 転移元は，シミュレータ上や研究施設等にあり，大量の動作サンプルを取得可能

(4) 転移先は，実機等とし，少量の動作サンプルのみ取得可能

本章でのロボットの “動作” とは，腕のエンドエフェクタ（end-effector: 人の手に当たる部分）を任意の場所へ移動させることを意味している．一般にロボットの物理特性が既知の場合，逆運動学（Inverse Kinematics）[65, 66] を用いることにより任意の場所に移動させる事が可能である．ところが，本手法では物理特性を得ていないため，経験ベースの Inverse Kinematics を実施する．

4.2.2 経験ベースの逆運動学

ここでは，ロボットが獲得するデータと，そのデータを用いた逆運動学（Inverse Kinematics）の方法について記述する．

4.2.2.1 経験から獲得するデータ

まず，対象とするロボットの腕が持つ全ての関節角を表した n 種類の関節値データセット $\mathbf{J} = \{\vec{j}_1, \vec{j}_2, ..., \vec{j}_n \,|\vec{j}_i \ni \text{each joint value}\}$ を作成する．m 軸のロボットの場合には，$\vec{j}_i$ は m 次元ベクトルとなる．そして，関節値 $\vec{j}_i$ を用いてロボットを稼働させた後，ロボットの各種センサから腕のエンドエフェクタの絶対座標値を求め，それをまとめたデータセット $\mathbf{X} = \{\vec{x}_1, \vec{x}_2, ..., \vec{x}_n\}$（$\vec{x}_i$ は，XY 平面や，XYZ 空間等）を作成する．シミュレータ上のロボットなど，物理特性が既知の場合には，Forward Kinematics [65] を用いることで，座標値取得の所要時間を短縮する．なお，エンドエフェクタの座標値は，ロボットの関節値より一意に決定される．これらが経験から獲得するデータであり，これらをまとめたデータセット $\mathbf{D}$ は，Equation (4.2.1)(4.2.2) のよう表される．

$$\mathbf{D} = \langle \mathbf{J}, \mathbf{X} \rangle \ni \langle \vec{j}_i, \vec{x}_i \rangle \tag{4.2.1}$$

$$\textbf{each } \vec{x}_i = f(\vec{j}_i) \tag{4.2.2}$$

4.2.2.2 経験に基づく関節値の推定

ここでは，腕のエンドエフェクタを任意の絶対座標値 $\vec{x}_{goal}$ へ移動させる関節値 $\vec{j}_{ik}$ を，獲得したデータセット $\mathbf{D}$ から求める手法について記述する．大まかな方針としては，$\vec{x}_{goal}$ に対する k 個の近傍データを座標値データセット $\mathbf{X}$ の中から探し出し，それらに対応する関節値の逆距離加重平均 (Inverse Distance Weighting [67]) を $\vec{j}_{ik}$ とする．ところが，指定された場所に腕を動かす時に，複数の関節値の組み合わせが存在する場合がある [1]．そのため，似通った座標値であっても，関節値が非常に異なる場合が存在する．その場合，座標値空間だけの近傍を用いて平均を算出すると，全く異なった関節値同士の平均を用いてしまう可能性もある．

そこで，座標値空間で複数の近傍サンプルを求めた後，関節値の空間で再度絞り込みを行うことで，関節値空間内での距離も考慮した $\vec{j}_{ik}$ の算出アルゴリズムを提案する．具体的には，$\mathbf{X}$ の中から $\vec{x}_{goal}$ の最近傍サンプル $\vec{x}_{one}$ と，複数の近傍サンプル $\mathbf{X}_{knn}$ を探し出し，それぞれの関節値 $\vec{j}_{one}$ と $\mathbf{J}_{knn}$ を求める．その後，$\mathbf{J}_{knn}$ の中から $\vec{j}_{one}$ の近傍関節値 $\mathbf{J}_{selected}$ を探し出し，それらの逆距離加重平均を $\vec{j}_{ik}$ とした．これらの具体的なアルゴリズムを Algorithm 2 に示す．なお本稿では，パラメータ k を 3 とした．

[1] 特に，関節値の次元数が，座標値の次元数以上の場合など

Algorithm 2 目的の関節値を生成するアルゴリズム

Require: $\mathbf{D} = \langle \mathbf{J}, \mathbf{X} \rangle$:the dataset of motions, $\vec{j}_{ik}$:joints value for result, $\vec{x}_{goal}$:goal position of end-effector, knn$(\mathbf{A}, \vec{x}, k)$:choose k-nearest neighbor vectors for $\vec{x}$ from $\mathbf{A}$, max(a, b):return larger value, k:parameter for a number of interpolating, $\epsilon = 2^{-52}$

1: $\mathbf{X}_{knn} \leftarrow$ knn$(\mathbf{X},\ \vec{x}_{goal},\ 3\,k)$
2: $\vec{x}_{one} \leftarrow$ knn$(\mathbf{X},\ \vec{x}_{goal},\ 1)$
3: $\mathbf{J}_{knn} \leftarrow$ find each joints value of $\mathbf{X}_{knn}$ from $\mathbf{D}$
4: $\vec{j}_{one} \leftarrow$ find joints value for $\vec{x}_{one}$ from $\mathbf{D}$
5: $\mathbf{J}_{selected} \leftarrow$ knn$(\mathbf{J}_{knn},\ \vec{j}_{one},\ k)$
6: $\mathbf{X}_{selected} \leftarrow$ find each end-effector of $\mathbf{J}_{selected}$ from $\mathbf{D}$
7: $\vec{j}_{ik} \leftarrow 0,\ \vec{w}_{sum} \leftarrow 0$
8: **for all** $\langle \vec{j}_i, \vec{x}_i \rangle \in \langle \mathbf{J}_{selected}, \mathbf{X}_{selected} \rangle$ **do**
9: $\quad \vec{w} \leftarrow 1/\mathbf{max}(\|\vec{j}_{one} - \vec{j}_i\| + \|\vec{x}_{goal} - \vec{x}_i\|, \epsilon)$
10: $\quad \vec{w}_{sum} \leftarrow \vec{w}_{sum} + \vec{w}$
11: $\quad \vec{j}_{ik} \leftarrow \vec{j}_{ik} + \vec{j}_i / \vec{w}$
12: **end for**
13: $\vec{j}_{ik} \leftarrow \vec{j}_{ik} / \vec{w}_{sum}$

4.2.3 問題設定

ここでは，本稿で提案する動作情報の転移に関する問題設定を記述する．

動作の転移とは，転移元のロボット上で作成された動作の経験データベース $\mathbf{D}_{src}$ から，転移先のロボットの作業空間にあわせた経験データベース $\hat{\mathbf{D}}_{tgt}$ を作成することを意味している．この時，$\hat{\mathbf{D}}_{tgt}$ は，転移先のロボット上にて少量のみ取得した動作の

経験データセット $\mathbf{D}_{tgt}$ によって $\mathbf{D}_{src}$ を補正することで作成する．なお，この時，$\mathbf{D}_{src}$ などの経験データベースは，単純に関節値とエンドエフェクタの座標値のペアの集合であれば良く，SOM を改良させた SOINN [24] などを持ちいて情報を圧縮することも可能である．

4.2.4 提案手法

本稿では，SOM 変形法と，それを改良した誤差伝播法を提案する．前者は，座標値空間を SOM の更新式と似た式で更新する手法であり，後者は，転移元と転移先の座標値の誤差（差分）を用いて更新する手法である．

4.2.4.1 SOM 変形法 (Transfer by SOM-algorithm)

本手法は，転移先のロボットから得られた座標値を用いて，SOM [26] を参考にして作成した更新式で，データセット $\mathbf{X}_{src}$ を変形させる手法である．アルゴリズムを Algorithm 3 に記述する．

なお本手法は，SOM の更新式から着想されたアルゴリズムであるが，問題設定に合わせて更新率の変動方法を変更している．従来の SOM は，更新する空間内の近傍ノードとの距離で更新率の決定している．一方で，本問題設定では，関節値空間と座標空間の 2 つの空間が存在している．更新する空間は，座標空間が適当と考えられるが，関節値空間内での距離も重要となっている．その理由は，4.2.2.2 項で取り上げた

Algorithm 3 SOM 変形法のアルゴリズム

Require: $\mathbf{D}_{src} = \langle \mathbf{J}_{src}, \mathbf{X}_{src} \rangle$:the large dataset generated by source robot, $\mathbf{D}_{tgt} = \langle \mathbf{J}_{tgt}, \mathbf{X}_{tgt} \rangle$:the small dataset generated by target robot, num($\mathbf{A}$):number of $\mathbf{A}$, $sgm(a, x)$:sigmoid function (a =gain, x =value), $\{a_s, b_s\}$:parameter

1: $t \leftarrow 1$
2: $\hat{\mathbf{D}}_{tgt} \leftarrow \mathbf{D}_{src}$
3: **for all** $\langle \vec{j}_{tgt}, \vec{x}_{tgt} \rangle \in \mathbf{D}_{tgt}$ **do**
4: $\quad a \leftarrow a_s + 9a_s\, t/\text{num}(\mathbf{D}_{tgt})$
5: $\quad b \leftarrow b_s + 9b_s\, t/\text{num}(\mathbf{D}_{tgt})$
6: $\quad \vec{j}_{nearest} \leftarrow$ nearest data to $\vec{j}_{tgt}$ in $\hat{\mathbf{J}}_{tgt}$
7: $\quad \vec{x}_{nearest} \leftarrow$ search pair with $\vec{j}_{nearest}$ in $\hat{\mathbf{D}}_{tgt}$
8: $\quad d_{nearest} \leftarrow \|\vec{x}_{tgt} - \vec{x}_{nearest}\|$
9: $\quad$ **for all** $\langle \vec{j}_i, \vec{x}_i \rangle \in \hat{\mathbf{D}}_{tgt}$ **do**
10: $\quad\quad d_j \leftarrow \|\vec{j}_{tgt} - \vec{j}_i\|$
11: $\quad\quad \Delta\vec{x}_i \leftarrow \vec{x}_{tgt} - \vec{x}_i$
12: $\quad\quad \vec{x}_i \leftarrow \vec{x}_i + 4\, sgm(a, -d_j)\, sgm(b, -d_{nearest})\, \Delta\vec{x}_i$
13: $\quad$ **end for**
14: $\quad t \leftarrow t + 1$
15: **end for**

問題点と同様である．そこで，関節値の距離も更新率に反映するように変更している．

4.2.4.2 誤差伝播法 (Transfer by Error propagation)

本手法は，転移元と転移先のロボットより得られた値の差分を基に変形させる手法である．具体的には，転移先より得られたデータセット $\langle \vec{j}_{tgt}, \vec{x}_{tgt} \rangle \in \mathbf{D}_{tgt}$ を用いて，

Algorithm 4 誤差伝播法のアルゴリズム

Require: $\mathbf{D}_{src} = \langle \mathbf{J}_{src}, \mathbf{X}_{src} \rangle$:the large dataset generated by source robot, $\mathbf{D}_{tgt} = \langle \mathbf{J}_{tgt}, \mathbf{X}_{tgt} \rangle$:the small dataset generated by target robot, num($\mathbf{A}$):number of $\mathbf{A}$, $sgm(a, x)$:sigmoid function (a =gain, x =value), a_e:parameter

1: $t \leftarrow 1$
2: $\hat{\mathbf{D}}_{tgt} \leftarrow \mathbf{D}_{src}$
3: **for all** $\langle \vec{j}_{tgt}, \vec{x}_{tgt} \rangle \in \mathbf{D}_{tgt}$ **do**
4: $\quad a \leftarrow a_e + 9a_e\, t/\mathrm{num}(\mathbf{D}_{tgt})$
5: $\quad \vec{j}_{nearest} \leftarrow$ nearest data to $\vec{j}_{tgt}$ in $\hat{\mathbf{J}}_{tgt}$
6: $\quad \vec{x}_{nearest} \leftarrow$ search pair with $\vec{j}_{nearest}$ in $\hat{\mathbf{D}}_{tgt}$
7: $\quad \Delta\vec{x}_{nearest} \leftarrow \vec{x}_{tgt} - \vec{x}_{nearest}$
8: $\quad$ **for all** $\langle \vec{j}_i, \vec{x}_i \rangle \in \hat{\mathbf{D}}_{tgt}$ **do**
9: $\quad\quad d_j \leftarrow \|\vec{j}_{tgt} - \vec{j}_i\|$
10: $\quad\quad \vec{x}_i \leftarrow \vec{x}_i + 2\, sgm(a, -d_j)\, \Delta\vec{x}_{nearest}$
11: $\quad$ **end for**
12: $\quad t \leftarrow t + 1$
13: **end for**

転移元のロボット上で $\vec{j}_{tgt}$ を実行した時の誤差を $\mathbf{D}_{src}$ より算出して，その値を周辺のサンプルデータへと伝播させて変形させる．アルゴリズムを Algorithm 4 に記述する．本手法は，そのほとんどが SOM 変形法と類似しているが，Algorithm 3 の 12 行目，Algorithm 4 の 10 行目の更新時に用いる値が異なっている．前者は，更新するニューロンとの差分を，後者は，最近傍ニューロンとの差分を用いている．

Table 4.1: 2 軸のロボットに対する腕の長さ

	Arm 1	**Arm 2**
Robot Src	0.8	1.2
Robot A	0.9	1.1
Robot B	1.0	1.0
Robot C	1.5	1.0

4.3 実験

本稿では，2 種類のロボットで実験を行う．2 次元上を操作する 2 軸のロボットと，3 次元上を操作する 6 軸のロボットである．主に前者で定量的な評価を，後者で有効性の検証を実施する．

4.3.1 2 軸のロボット上での実験

4.3.1.1 実験環境

腕の長さが異なる 2 軸の片腕ロボットを 4 種類シミュレータ上に作成し，それを用いて動作転移を実施して，転移先のロボット上での位置誤差を測定する．それぞれのロボットの腕の長さを Table 4.1 に，各々のロボットに対して関節をそれぞれ段階的に動かした時のエンドエフェクタの到達地点（Working space）を Figure 4.2 に示す．

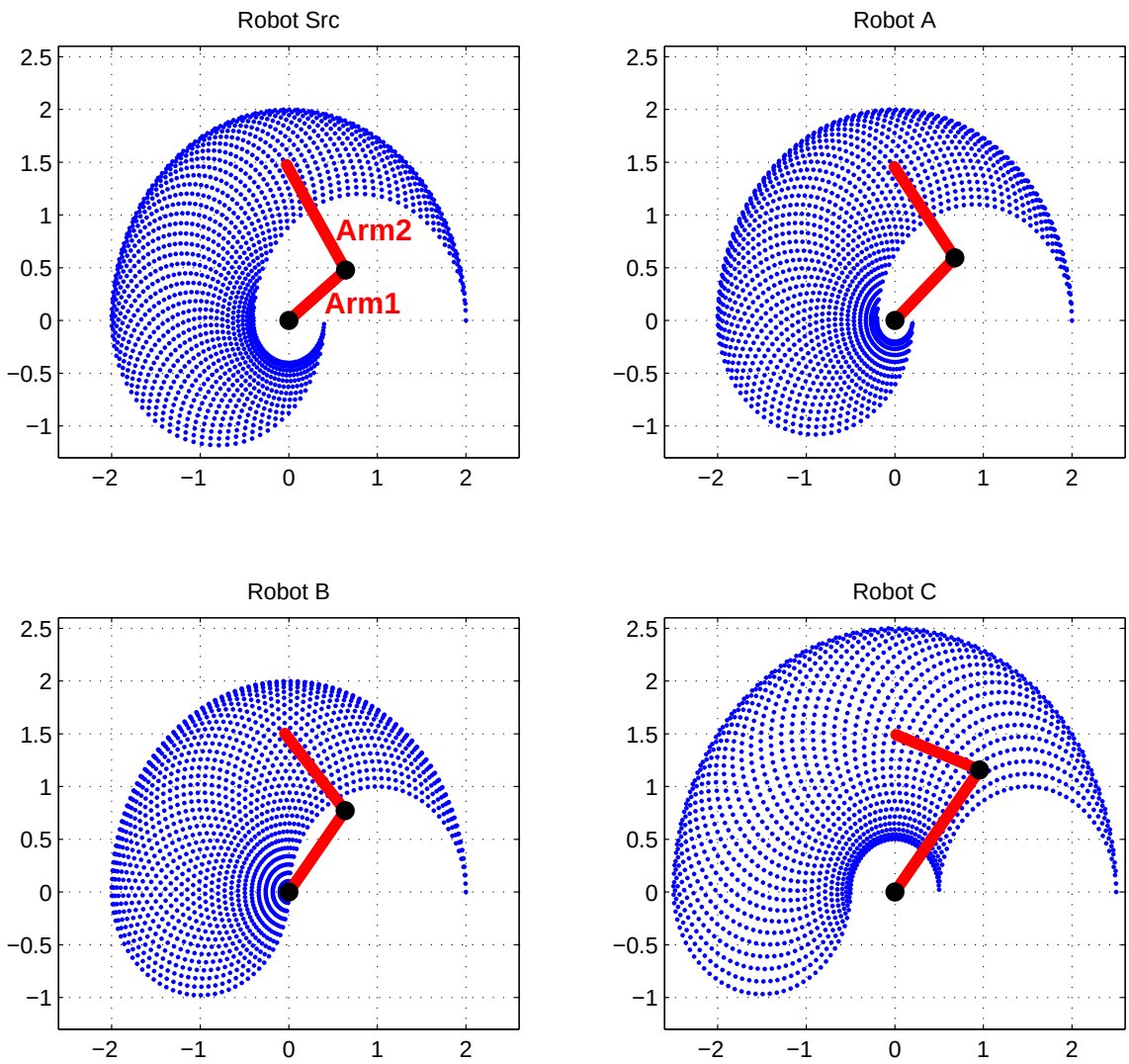

Figure 4.2: それぞれのロボットに対するエンドエフェクタの可動領域
赤い線：ロボットの腕，黒い点：駆動点，青い点：エンドエフェクトの到達点

実験は，ロボット Src にて獲得した動作情報をロボット A や，B，C へ転移させて実施する．その時の位置誤差 $err = \|f_{tgt}(\vec{j}_{ik}) - \vec{x}_{goal}\|$ を計測する．この時，それぞれの関節の可動範囲を，0 から 180 度に限定した場合（以降 “関節値に制限あり” と呼び，Figure 4.2 は制限ありである）と，0 から 360 度にした場合（“関節値に制限なし” と呼ぶ）の 2 つの条件で実験を行う．後者の場合，任意の場所にエンドエフェクタを持って行く時に，複数の関節値の候補が存在するため，動作が一意に定まらず誤差が大きくなると考えられる [2]．転移元のロボット Src にて獲得したデータベースは，それぞれの関節の値を $0.0314 \fallingdotseq \pi/100[rad]$ ずつ動かした時に得られるグリッド上の関節値データと，それを実行した時の座標値（XY 平面）である．そのため，$\mathbf{D}_{src}$ のサンプル数は，関節値に制限ありで 10000 サンプル，制限なしで 40000 サンプルとなった．転移先のロボットで獲得した $\mathbf{D}_{tgt}$ は，無作為に設定された 50 種類の関節値とその座標値とした．

評価は，無作為に設定された 200 箇所の開始地点から $\vec{x}_{goal}$ に向けて腕を動かす実験を，$\vec{x}_{goal}$ を無作為に変更して 100 セット実施し，その時の位置誤差の平均値と標準偏差を用いた．また，比較対象は，以下の 6 種類のデータセットを用いた場合の動作である．

[2] Algorithm 2 により，誤差が少なくなるように対応している

LSrc 転移元でロボットで得たサンプル $\mathbf{D}_{src}$ を利用した場合

STgt 転移先でロボットで得た少数サンプル $\mathbf{D}_{tgt}$ を利用した場合

TSom SOM 変形法を用いて転移した場合

TErr 誤差伝播法を用いて転移した場合

LTgt 転移先にて大量サンプルを得た場合[3]

なお，転移先のロボットで取得した大量のサンプルデータ（LTgt）は，今回の実験における理論的な最小誤差であり，現実では取得困難なサンプルデータとなる．また，SOM 変形法のパラメータは $a_s = 1$，$b_s = 0.001$ として，誤差伝播法のパラメータは $a_e = 0.1$ とした．これらのパラメータは，一番転移が困難だと考えられる関節値に制限なしのロボット C への転移で，ステップ数を $0.314 \fallingdotseq \pi/10$ とした場合の実験（小規模実験）を行いて算出した．試行した値は，$\{10^n | n = -3, -2, ..., 1\}$ であり，それぞれ最良値を採用している．

4.3.1.2 結果

関節値に制限ありの場合の位置誤差の箱ひげ図を Figure 4.3 に示し，制限なしの箱ひげ図を Figure 4.4 に示す．表示の関係で，これらの箱ひげ図は，1 セットごとに平均を取って作図している．また，それぞれの平均値，標準偏差を Table 4.2 に記す．

[3]手法の理論的な最小誤差であり，現実では取得不可能

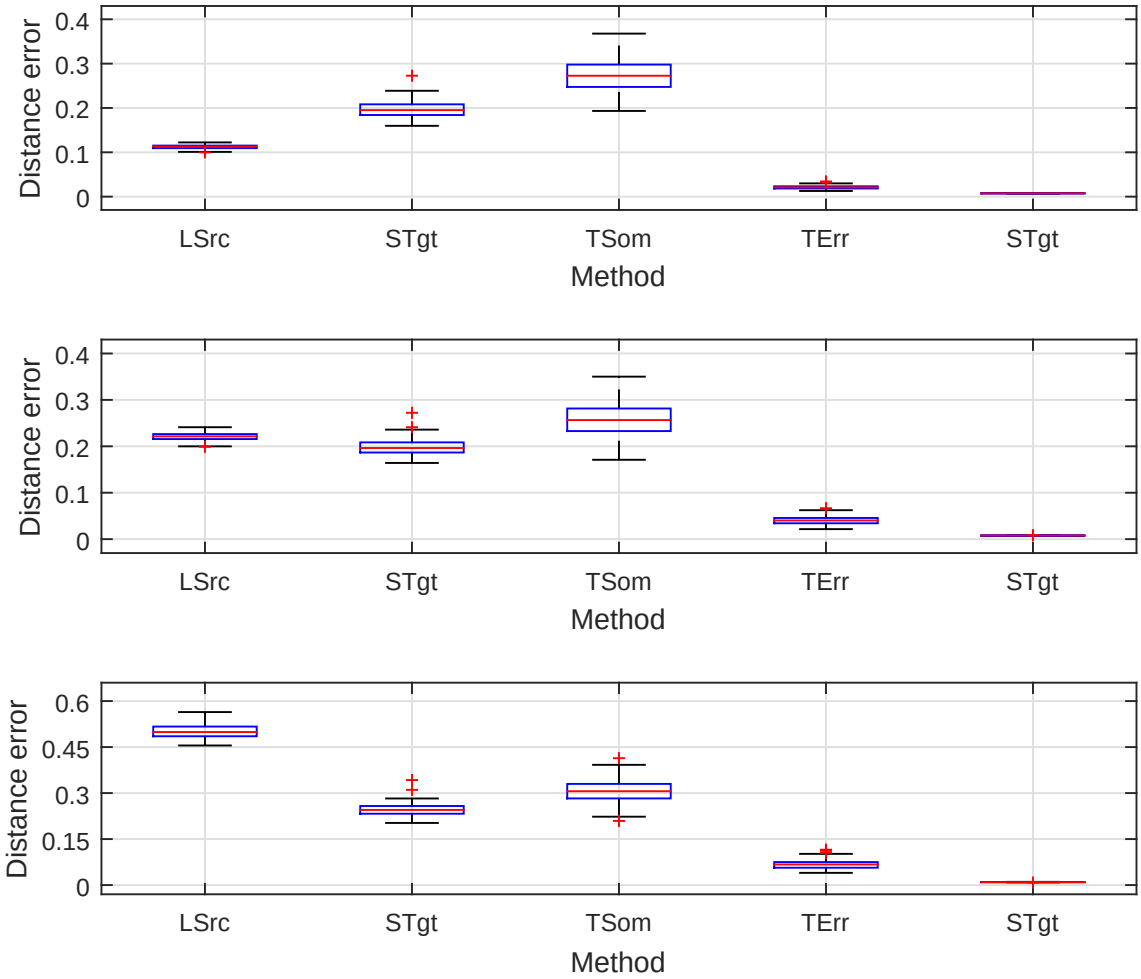

Figure 4.3: 関節値に制限ありの場合の駆動誤差
上：ロボット A への転移，中：ロボット B への転移，下：ロボット C への転移

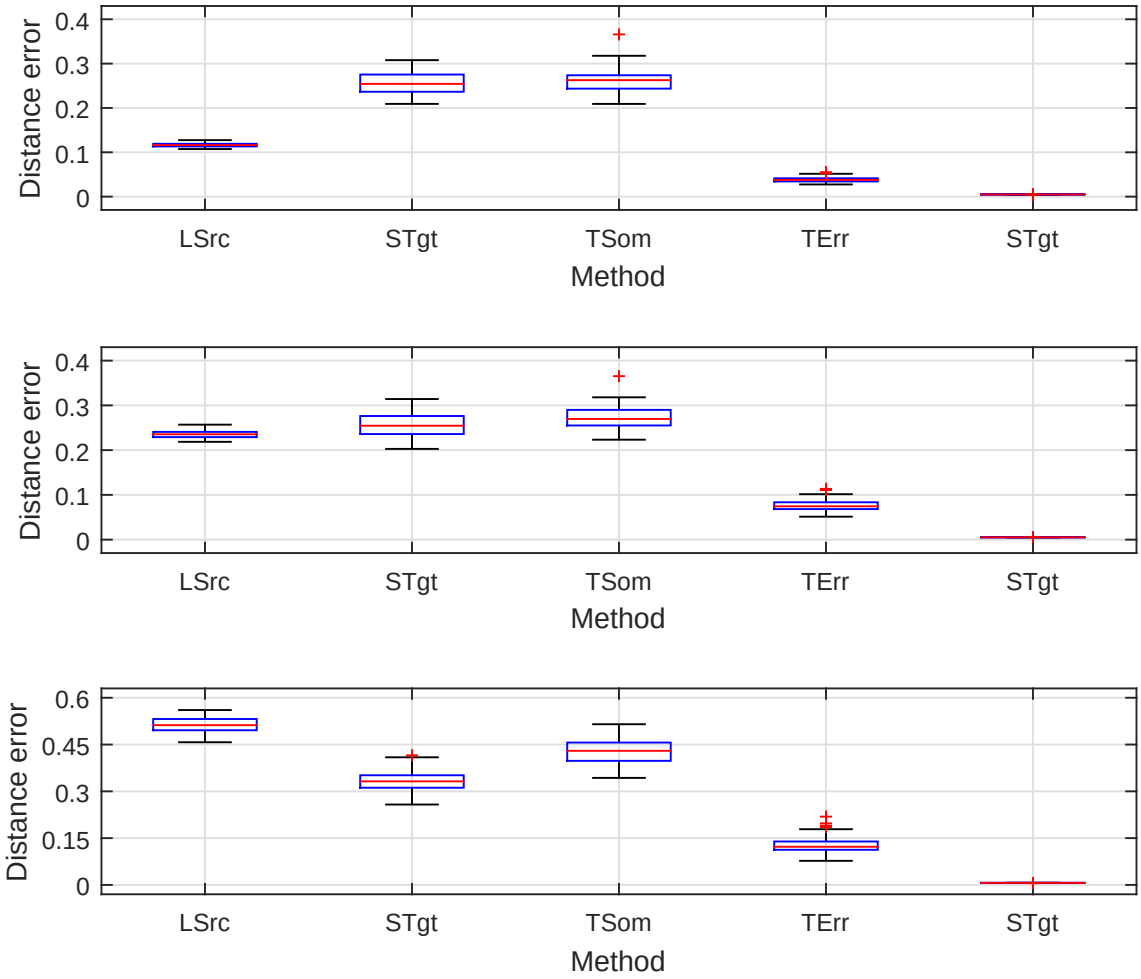

Figure 4.4: 関節値に制限なしの場合の駆動誤差
上：ロボット A への転移，中：ロボット B への転移，下：ロボット C への転移

Table 4.2: それぞれの手法による駆動誤差
L：関節値に制限あり，N：関節値に制限なし

		Average ± Standard Deviation		
		Robot A	Robot B	Robot C
L	LSrc	0.11 ± 0.06	0.22 ± 0.12	0.50 ± 0.31
	STgt	0.20 ± 0.15	0.20 ± 0.15	0.25 ± 0.18
	TSom	0.28 ± 0.19	0.26 ± 0.18	0.31 ± 0.21
	TErr	**0.02 ± 0.02**	**0.04 ± 0.03**	**0.07 ± 0.06**
	LTgt	0.01 ± 0.00	0.01 ± 0.00	0.01 ± 0.01
N	LSrc	0.12 ± 0.06	0.24 ± 0.12	0.51 ± 0.32
	STgt	0.26 ± 0.19	0.26 ± 0.19	0.33 ± 0.24
	TSom	0.26 ± 0.17	0.27 ± 0.17	0.43 ± 0.24
	TErr	**0.04 ± 0.03**	**0.08 ± 0.07**	**0.13 ± 0.12**
	LTgt	0.01 ± 0.00	0.01 ± 0.00	0.01 ± 0.00

なお，経験ベースの逆運動学を用いているため，転移先での大量サンプル（LTgt）を用いても誤差はゼロにならない．

これらの結果より，誤差伝播法 (TErr) を用いることで，転移先のロボットから大量のサンプルデータを得た状態 (LTgt) に近い位置誤差まで抑えることができた．また，ロボットのスペック差が大きい時や，関節値が一意に定まらない時（関節値に制限なし）に精度が低下することが分かった．一方で，誤差伝播法を用いても未だ位置

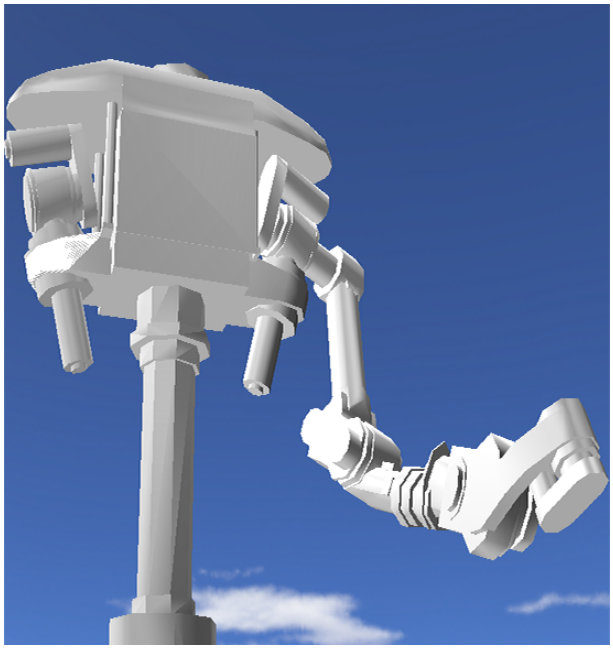

Figure 4.5: 使用したシミュレータ上のロボット
（この図はシミュレータ α ）

誤差は存在しており，実機のロボットで実際に動作できるか不明である．

4.3.2 実機のロボット上での実験

4.3.2.1 実験環境

実機のロボットが持つ 6 軸の左腕と，その腕に似たスペックを持つロボットをシミュレータ上に 2 種類作成した．作成したシミュレータ上のロボットの外観を Figure 4.5 に，実機のロボットの外観を Figure 4.6 に，それぞれのロボットの 2 つの腕と手首の長

Figure 4.6: 使用した実機のロボット

さを Table 4.3 に示す．また，ロボットの関節の場所，及びその回転方向を Figure 4.7 に示す．なお，今回の実験ではロボットの左腕を動かしており，関節 9 から関節 14 までが対象となる．なお，関節 9 と関節 10，及び，関節 12 と関節 13 は同じ場所に設置されている．

本実験では，シミュレータ上のロボットから実機のロボットに対して動作情報の転移を実施する．シミュレータ上で獲得したデータセット $\mathbf{D}_{src}$ は，それぞれの関節を $0.0157 \fallingdotseq \pi/200[rad]$ ずつ動かしたものとした．そのため，$\mathbf{D}_{src}$ のサンプル数は，約 70 万サンプルとなった．実機のロボットで得た $\mathbf{D}_{tgt}$ は，100 種類の動作サンプルとした．

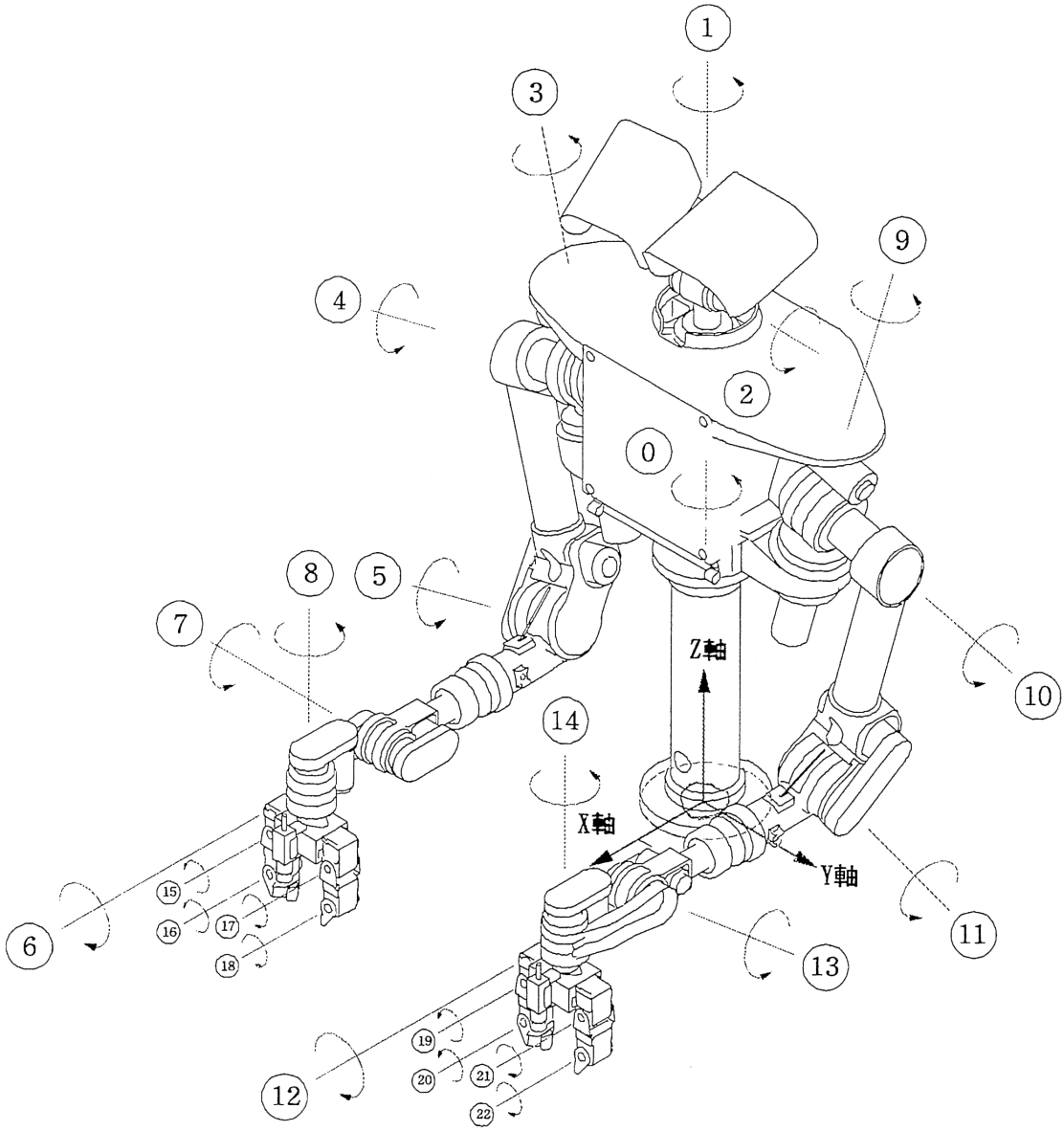

Figure 4.7: ロボットの関節の場所と回転方向

Table 4.3: 6 軸のロボットに対する腕の長さ

	Arm 1 (b/w jnt_{10} and jnt_{11})	Arm 2 (b/w jnt_{11} and jnt_{13})	Wrist (b/w jnt_{13} and jnt_{14})
Smlt-α	20 cm	15 cm	10 cm
Smlt-β	20 cm	20 cm	10 cm
Real	25 cm	22 cm	9 cm

また本実験では，位置誤差と計算時間による定量的な評価と，実機を用いた有効性の評価を行う．この時，実機上での実験では，指定された色を持つ物体の場所をロボットが自ら特定して，その場所まで転移結果を用いて手を伸ばし，置かれた物体を把持するまでを動作とした．物体位置の絶対座標値は，ロボットに搭載されたカメラと深度センサを用いることで特定させる．なお，把持の動作には物体への近接パスも重要となり，それにより動作の過程で対象物体と手が衝突する場合も考えられる．ところが，この問題の解決は，本稿の目的ではないため，ロボットの関節値に制限をかけることで回避している．

4.3.2.2 結果

4.3.1 節と同様，エンドエフェクタの位置誤差の箱ひげ図を Figure 4.8 に示す．また，それぞれの平均値，標準偏差を Table 4.4 に記す．Figure 4.8 及び Table 4.4 より，

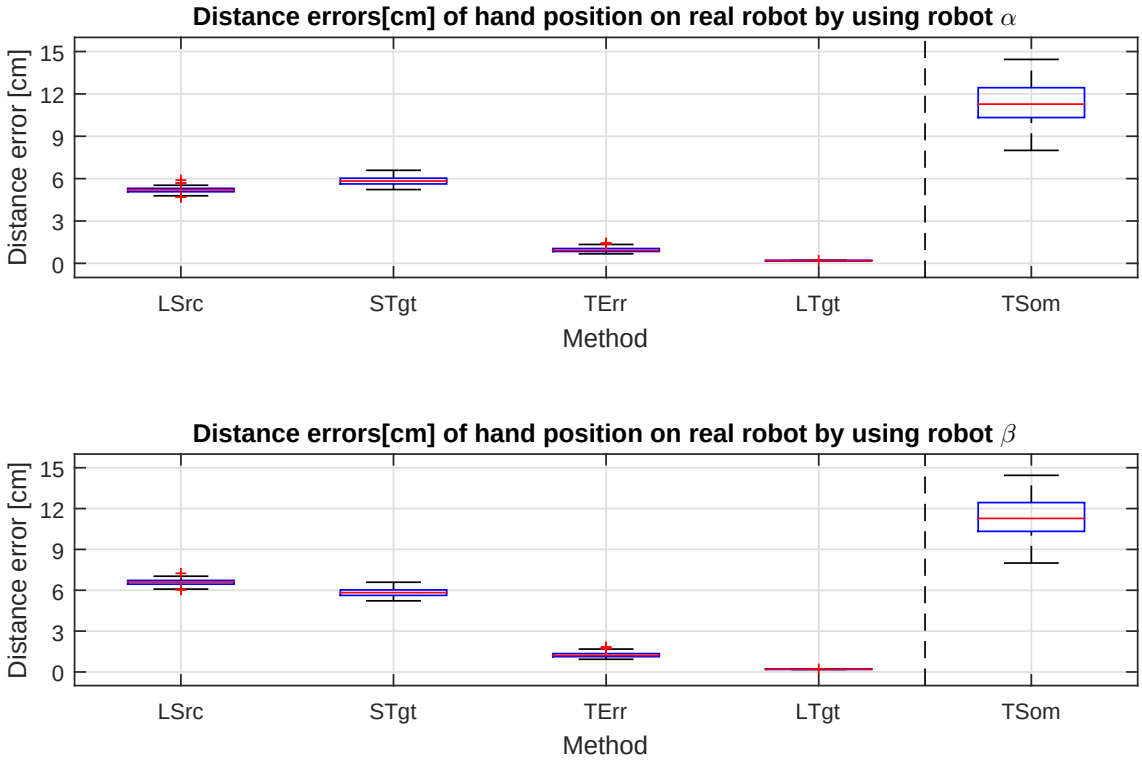

Figure 4.8: 実機のロボットを用いた駆動誤差
上：シミュレータ α からの転移，下：シミュレータ β からの転移

4.3.1 節の実験結果と同様，誤差伝播法を用いることで位置誤差を削減することができた．また，誤差伝播法を用いた 1 回の転移（Algorithm 4 を実行）に所用する時間は，3.30GHz の CPU を用いて 56.8 ± 0.3[sec] であり，実用に耐えうる時間であると言える．

次に，シミュレータ β からの転移結果を用いて，任意の場所に置かれた物体を把持する実験を実施した．その結果，実際に物体を把持できることを確認した [4]．

[4] デモビデオ（STgt → LSrc → TSom）：`http://youtu.be/vCexL1vpwHk`

Table 4.4: 実機のロボットを用いたそれぞれの手法による駆動誤差

	Average ± Standard Deviation [cm]	
	By robot α on smlt.	By robot β on smlt.
LSrc	5.19 ± 2.43	6.58 ± 2.62
STgt	5.85 ± 2.92	5.85 ± 2.92
TErr	**0.96 ± 0.53**	**1.26 ± 0.66**
LTgt	0.20 ± 0.07	0.20 ± 0.07
TSom	11.31 ± 5.02	11.31 ± 5.02

4.4 考察

4.4.1 SOM 変形法と誤差伝播法の違い

2 軸のロボット Src から A への転移を，それぞれの変形法で実施した結果の座標空間を Figure 4.9 と Figure 4.10 に記す．図中の青い点は転移後の絶対座標 $\hat{\mathbf{X}}_{tgt}$ を，青い線は移動の軌跡を[5]，赤い点は転移先のロボット上で獲得した $\mathbf{X}_{tgt}$ を，黒い小さな点は転移先のロボットでグリッド上の大量のデータを取得できると仮定した場合の座標値群を表している．なお，表示のために，関節に制限をかけて，更に移動間隔を $0.157 \fallingdotseq \pi/20[rad]$ としている．図中の全ての青い点が黒い点と重なっている状態の

[5]すなわち，線の根本は $\mathbf{X}_{src}$ を表している

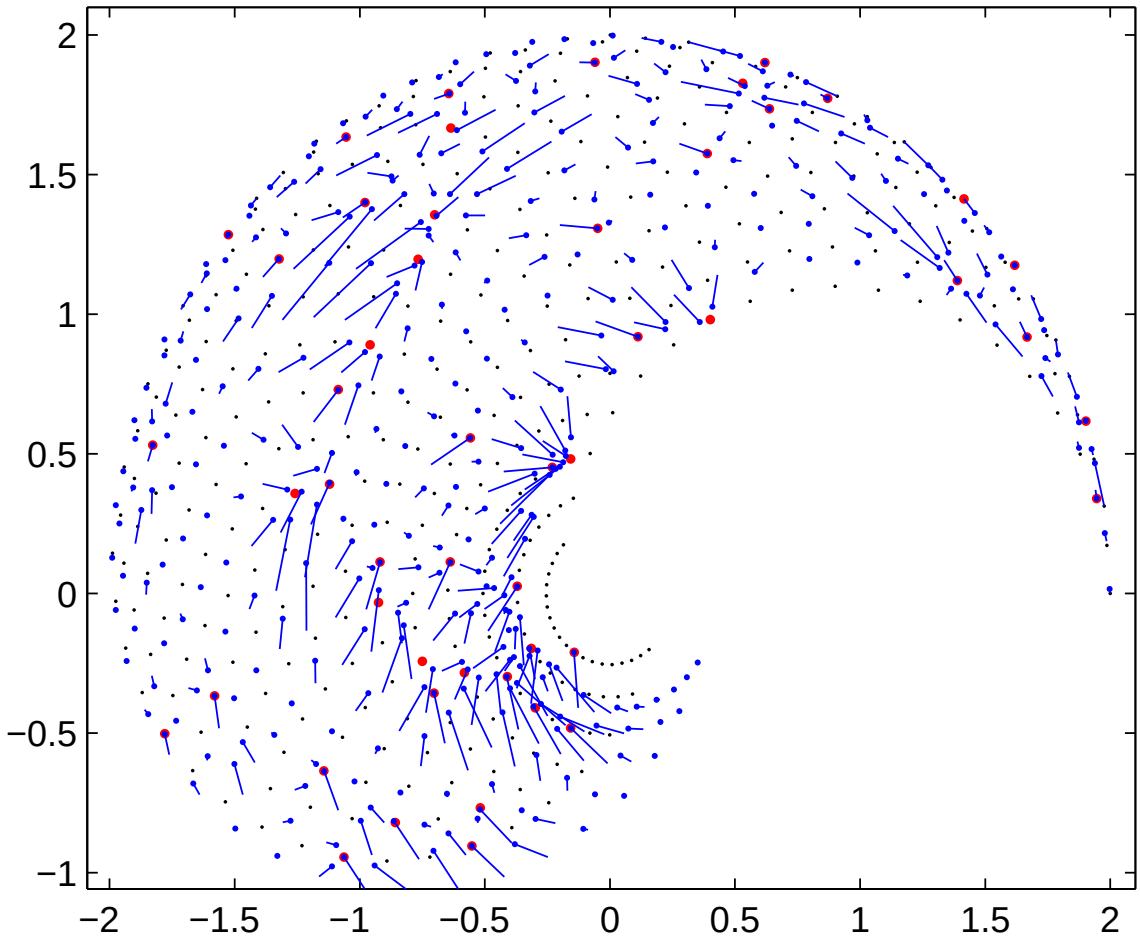

Figure 4.9: SOM 変形法を用いた転移後の可動領域

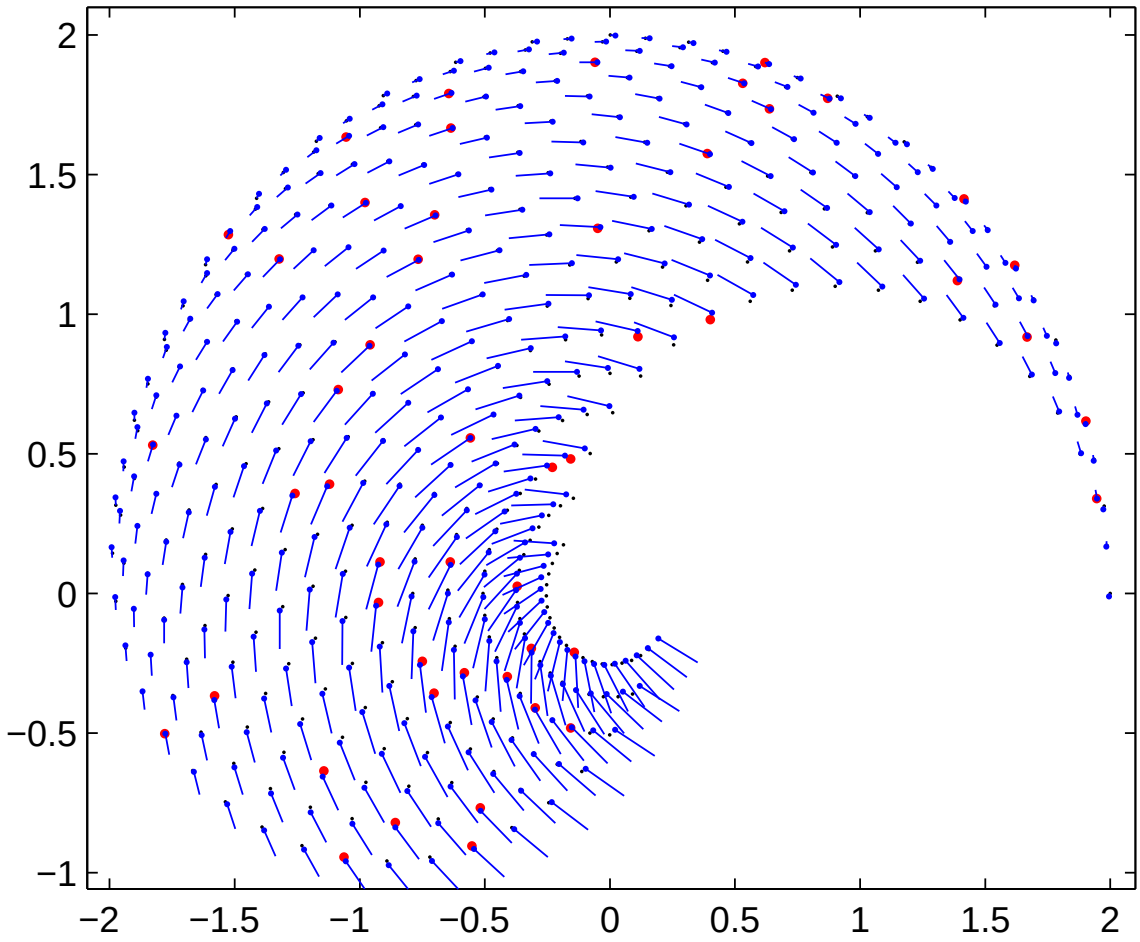

Figure 4.10: 誤差伝播法を用いた転移後の可動領域

時，最も位置誤差が小さくなる．

SOM 変形法では，入力された座標点（赤い点）に向かうように青い点が移動している．一方で誤差伝播法では，入力された座標点と転移元のロボット上での座標点との差，つまりは図中の青い線を周辺に伝播している．そのため，SOM 変形法よりも誤差伝播法のほうが，黒い点と重なった状態になっている．

SOM のアルゴリズムは，元々，未知の対象に対してネットワークを変形する手法として提案されたものである．一方で，今回の問題設定では事前に似た環境の情報を得ている状態からの “転移” を目的としている．そのため，新しい環境と学習済みの環境との誤差が重要となり，誤差伝播法を用いることで，より効果的な転移を実現できた．このような，ある環境から似た環境へ学習データを転移させる場合，本手法のような誤差伝播法を使うことが有効であると考えられる．

4.4.2 転移先のロボットから取得するサンプルの変動

先の実験では，転移先のロボットから取得するサンプルを固定して実験を行った．ここでは 2 軸のロボットを用いて転移先のロボットからの取得サンプル数を変化させた時の誤差の変化について考察する．

Figure 4.11，Figure 4.12，Figure 4.13 は，それぞれロボット A，B，C への転移について関節値に制限ありで転移先のサンプル数を変動させた時の誤差の変化を示している．青い箱ひげ図が，転移を行わず得られた転移先のサンプルのみを用いて動作し

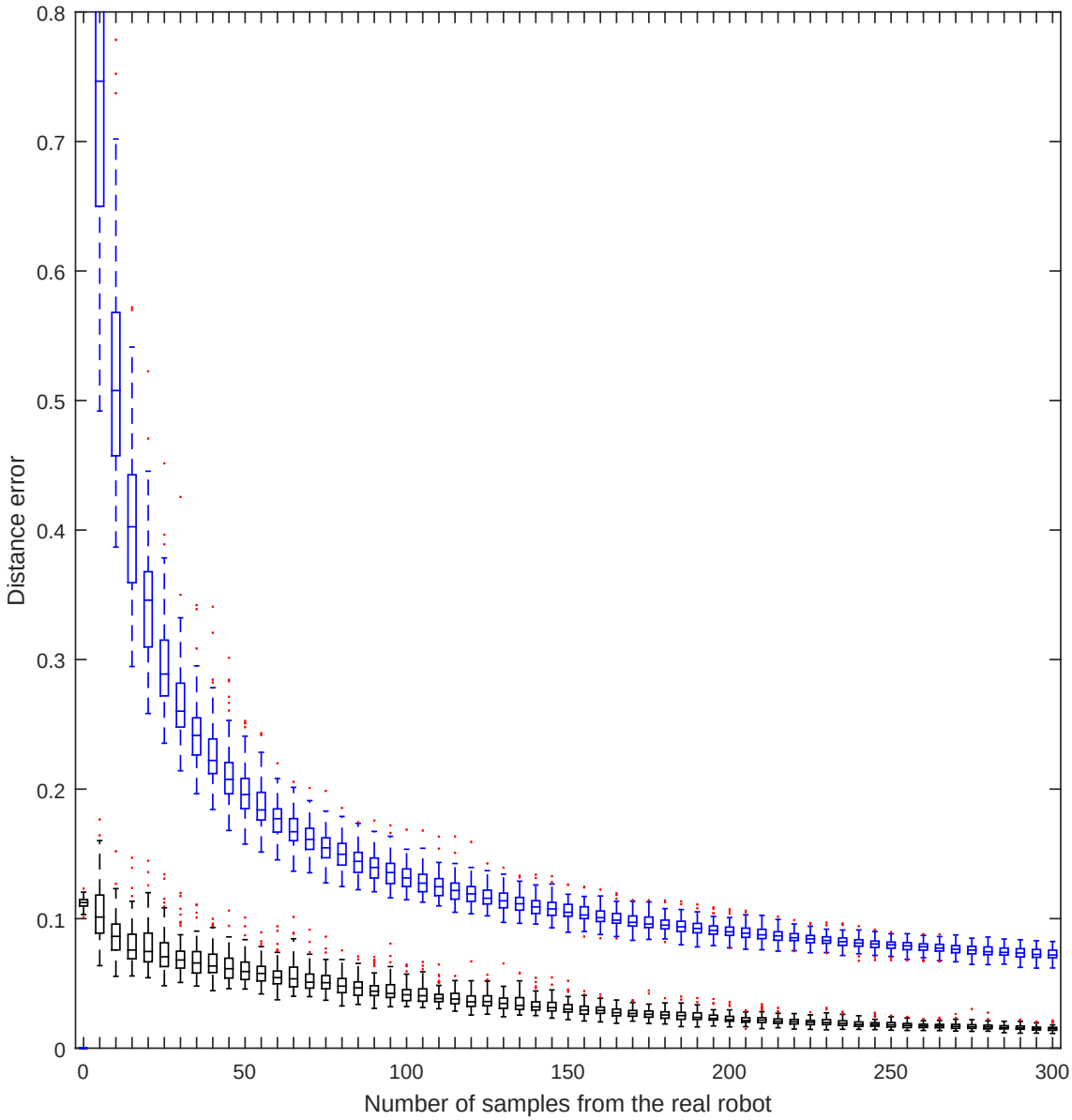

Figure 4.11: 転移先のサンプル数を変動させた時の誤差（ロボット A への転移）

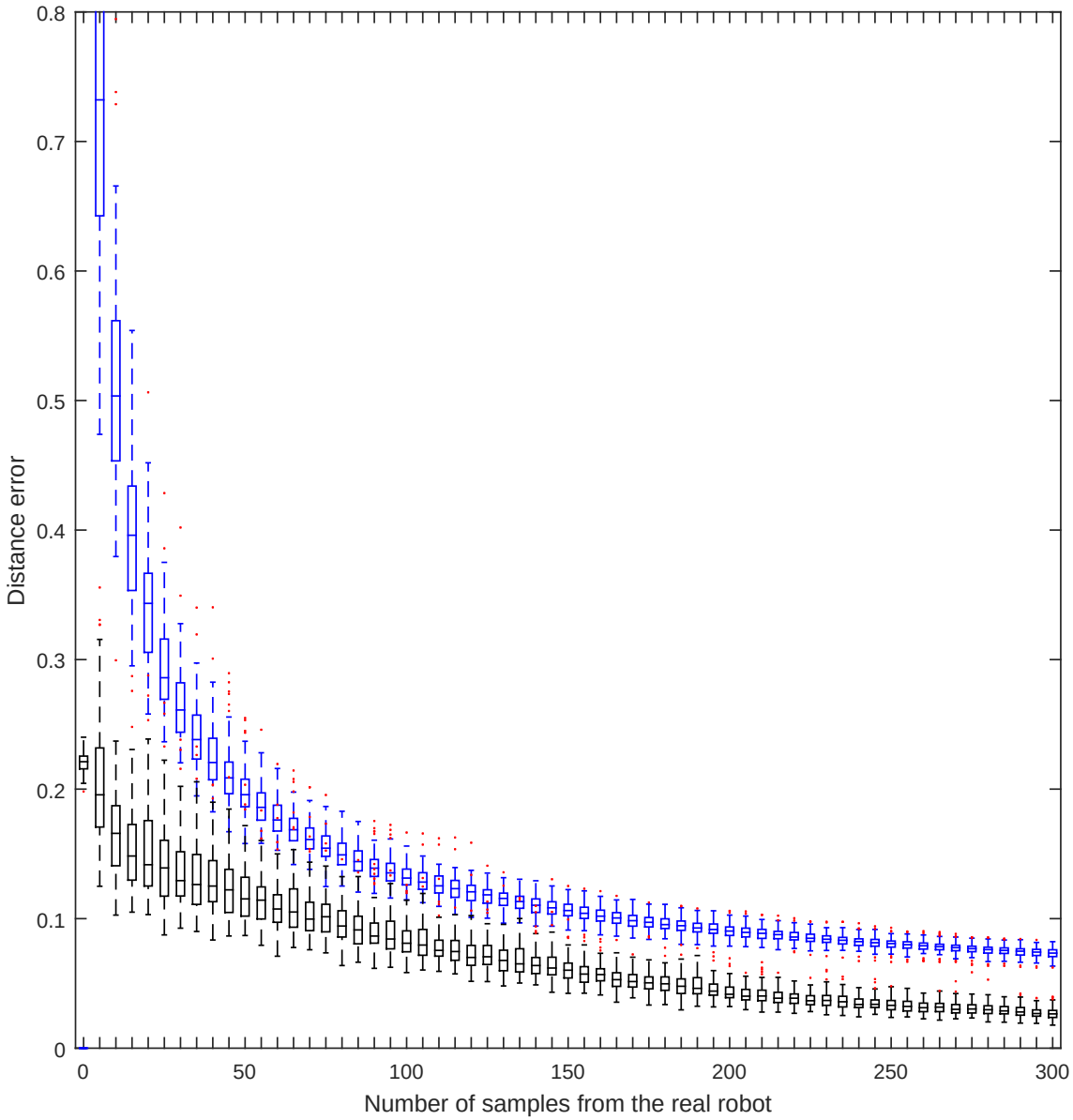

Figure 4.12: 転移先のサンプル数を変動させた時の誤差（ロボット B への転移）

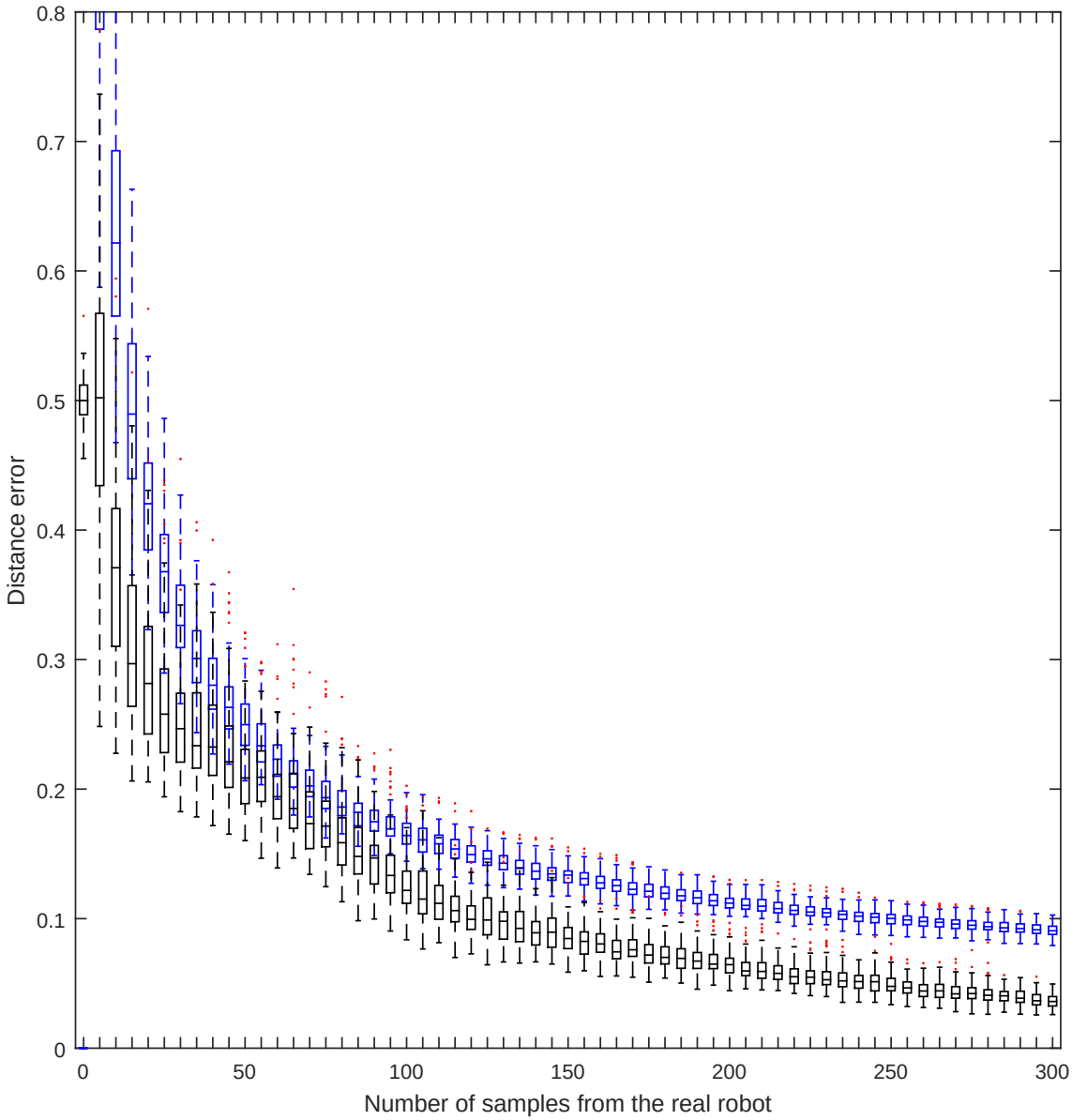

Figure 4.13: 転移先のサンプル数を変動させた時の誤差（ロボット C への転移）

た時の誤差を表し，黒い箱ひげ図が，転移を行った時の誤差の変化を表している．なお，先のサンプル数を固定した時の実験とは結果が多少ずれているが，それは更新率を調整するサンプル数 $\mathrm{num}(\mathbf{D}_{tgt})$ が本実験では全実験の最大値 300 となっているためである．全ロボットへの転移において，サンプル数が増えることにより徐々に誤差が小さくなっていることが確認できている．また，誤差は常に転移を用いたほうが少なくなっている．一方で，ロボット C への転移の内，50 サンプル辺りにおいて転移効果が低下している．これは，ロボット C とロボット Src のエンドエフェクトの作業領域を大きく変形しており，知識を転移するためには他に比べて多くのサンプルが必要であったためであると考えられる．

4.4.3 提案手法の可能性

本提案手法では，ロボットが広く普及した時を想定して，ロボット間の知識転移を実現した．具体的には，ある典型的なロボットが，研究施設等で十分に（本実験では数十万サンプル）学習されていれば，異なったスペックを持ったロボット上においても，多少訓練することで，瞬時に学習された知識を扱うことができるようになる．つまり，ロボット 1 台 1 台に対して動作の学習を行う手間を無くすことができ，このような基盤技術がロボットの普及のために必要であると考えられる．将来は，ロボット同士が動作情報も含めた知識を相互に教え合う技術の基盤になり得ると考えられる．

4.4.4 今後の課題

まず，実機間での転移の実現が挙げられる．本稿では物理的に実機のロボットが 1 台しかなかったため，シミュレータからの転移しか実験できなかった．今後，複数台の実機を用いてロボット同士が知識を共有する実験を行っていきたい．

次に，関節数が異なるロボット間での転移の実現が挙げられる．従来研究 [63] では，その対処について言及されており，この手法と組み合わせることにより，関節数が異なったロボット間でも転移を実現できると考えている．

4.5 まとめ

本章では，異なるスペックを持つロボット間での動作情報の知識転移を実施する手法を提案した．そして，提案手法を位置誤差を用いて定量的に評価した．その後，実機に搭載して実際に動作情報の知識転移を実証した．

Chapter

5

結論

5.1 自己転移と他機転移

本論文では，画像のみを用いた自己転移，及びマルチモーダル情報を用いた自己転移，動作情報を用いた他機転移に関する手法を提案した．ところが，これらは独立した手法ではないと考えられる．自己転移に関する手法は，自己増殖型ニューラルネットワークに改良を加えて学習器を提案しているが，その自己増殖型ニューラルネットワークの基となっているアルゴリズムは，SOM と GNG のアルゴリズムである．実際に SOM と同様に入力データに従ってネットワークを変形する仕組みを保持している．他機転移の手法の基盤となっている手法も，SOM であり，入力データに従って

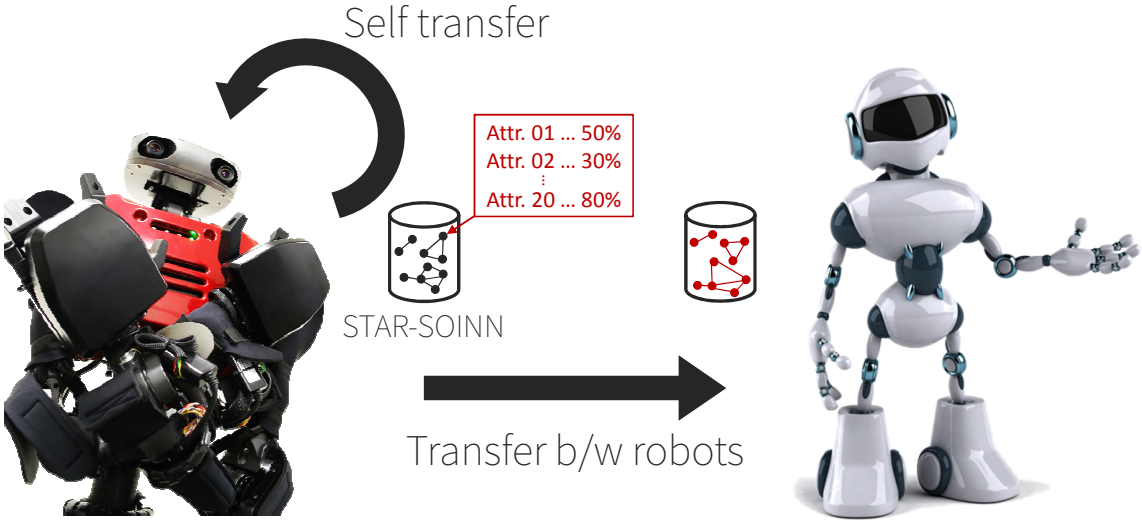

Figure 5.1: 本論文の手法を用いた自己転移と他機転移の統合

ネットワークを変形する仕組みを用いている．

これらの２つの転移手法を組み合わせることにより，新しい知識を学習してネットワークを形成するときには，自己転移を実施して少ないサンプルで効率的に学習し，新たな機体や自分の環境が変更された場合には，他機転移を実施して新たな環境に適応させる手法を実現可能である．すなわち，与えられた知識を STAR-SOINN のアルゴリズムからネットワークを構成して学習し，新たな環境への転移時には誤差伝播法を用いてネットワークを変形する．この統合された手法を用いることにより，本研究の目的である，“多種多様な知識” を効率的に学習し，“様々な特性を持つロボット” に対して対応できるような手法を実現することができると考えている．Figure 5.1 に，自己転移及び他機転移を共に実現した手法の概略を示す．

5.2 本論文のまとめ

本論文では，ロボットが社会へ幅広く進出した時に “多種多様な知識” を “様々な特性を持つロボット” に対して学習させる必要が存在し，それを解決するために知識転移を活用した知的発達手法を構築した．その構築にあたり，自らに転移を行う自己転移と，ロボット間で転移を行う他機転移について，それぞれ手法を提案した．更に以下に示す 3 種類の実験を実施し，手法の有効性や有用性を確認した．

画像情報を用いた知識転移

画像情報を自己転移する手法を提案した．そのために，まず，自己増殖型ニューラルネットワークを改良して，軽量な学習器 STAR-SOINN を構築した．その STAR-SOINN を基盤として，画像の転移手法を実現した．その後，従来手法と評価実験を行い，提案手法は非常に高速であり，更に転移学習を実現している手法であることを確認した．更には，インターネット上の画像を用いた転移手法を実現し，手法の安定性なども確認した．

マルチモーダル情報を用いた知識転移

STAR-SOINN を活用して，画像情報以外も含めた，様々なセンサ情報を活用可能な転移手法を提案した．そのために，モダリティと属性の関係度合いを計算する仕組みを取り入れ，特性の異なる様々なセンサから得られた属性情報を統

合する手法を実現した．その後，比較手法との評価実験を行い，提案手法が高精度であることを確認した．更に，ロボットに手法を実装し，実世界の未知の物体に対する属性を推定させる評価実験を実施し，有用性を確認した．

動作情報を用いた知識転移

他のロボットに対して腕の動作情報を転移する他機転移手法を提案した．そのために，ネットワーク構造を変形する自己組織化マップを改良して，実行時の誤差を伝播する手法を構築した．本手法により，転移先のロボット上で少量の動作情報を獲得するだけで，転移元での動作を活用可能な手法を実現した．その後，手法をロボットに実装し，比較手法との評価実験を行い，転移後に動作誤差が少なくなっていることを確認した．更に，実際にロボットに対して把持動作を行わせ，知識の転移を確認した．

一方で，各章末で述べたように，各手法にはそれぞれ課題が存在する．また，これらの実験の中で対象としていない，自然言語や自己位置情報もロボットの重要な知識と言え，今後検証の必要があると考えている．

本論文により，ロボットが社会に進出した時の重要な要素となる知識の転移手法が実現できたと，私は考えている．その一方で，本手法は，多くの問題の一つの解決策であると考えており，ロボットの知的発達に関して更なる研究が実施されることを期待している．

REFERENCES

[1] 総務省統計局国勢統計課：“平成 22 年度 国勢調査 最終報告書”, 国勢調査 (2011).

[2] 経済産業省：“産業構造ビジョン 2010”, 産業構造審議会産業競争力部会報告書 (2010).

[3] R. Brooks: “A robust layered control system for a mobile robot”, *Robotics and Automation, IEEE Journal of*, **2**, 1, pp. 14–23 (1986).

[4] 本田技研工業株式会社：“さらなる進化を遂げた「新型 asimo」と、ロボティクス研究および応用製品の総称「honda robotics」を発表”, 本田技研工業株式会社 プレスリリース (2011).

[5] 産業技術総合研究所：“人間に近い外観と動作性能を備えたロボットの開発に成功”, 産業技術総合研究所 プレスリリース (2009).

[6] S. Mobile and Aldebaran: “Softbank mobile and aldebaran unveil “pepper” - the world’s first personal robot that reads emotions”, *SoftBank Mobile Corp. Press Releases* (2014).

[7] D. Robotics: "3d robotics introduces solo, the world's first smart drone", *3D Robotics press release* (2015).

[8] J. Kober, J. A. Bagnell and J. Peters: "Reinforcement learning in robotics: A survey", *The International Journal of Robotics Research*, p. 0278364913495721 (2013).

[9] M. Ito, K. Noda, Y. Hoshino and J. Tani: "Dynamic and interactive generation of object handling behaviors by a small humanoid robot using a dynamic neural network model", *Neural Networks*, **19**, 3, pp. 323 – 337 (2006). The Brain Mechanisms of Imitation Learning.

[10] N. Michael, S. Shen, K. Mohta, Y. Mulgaonkar, V. Kumar, K. Nagatani, Y. Okada, S. Kiribayashi, K. Otake, K. Yoshida, K. Ohno, E. Takeuchi and S. Tadokoro: "Collaborative mapping of an earthquake-damaged building via ground and aerial robots", *Journal of Field Robotics*, **29**, 5, pp. 832–841 (2012).

[11] J. Engel, J. Sturm and D. Cremers: "Scale-aware navigation of a low-cost quadrocopter with a monocular camera", *Robotics and Autonomous Systems*, **62**, 11, pp. 1646 – 1656 (2014). Special Issue on Visual Control of Mobile Robots.

[12] D. Scaramuzza, M. Achtelik, L. Doitsidis, F. Friedrich, E. Kosmatopoulos, A. Martinelli, M. Achtelik, M. Chli, S. Chatzichristofis, L. Kneip, D. Gurdan, L. Heng, G. H.

Lee, S. Lynen, M. Pollefeys, A. Renzaglia, R. Siegwart, J. Stumpf, P. Tanskanen, C. Troiani, S. Weiss and L. Meier: "Vision-controlled micro flying robots: From system design to autonomous navigation and mapping in gps-denied environments", *Robotics Automation Magazine, IEEE*, **21**, 3, pp. 26–40 (2014).

[13] S. J. Pan and Q. Yang: "A survey on transfer learning", *Knowledge and Data Engineering, IEEE Transactions on*, **22**, 10, pp. 1345–1359 (2010).

[14] C. H. Lampert, H. Nickisch and S. Harmeling: "Learning to detect unseen object classes by between-class attribute transfer", *IEEE Conference on Computer Vision and Pattern Recognition* (2009).

[15] N. Kumar, A. C. Berg, P. N. Belhumeur and S. K. Nayar: "Attribute and simile classifiers for face verification", *IEEE International Conference on Computer Vision (ICCV)* (2009).

[16] M. Rohrbach, M. Stark, G. Szarvas, I. Gurevych and B. Schiele: "What helps where – and why? semantic relatedness for knowledge transfer", *IEEE Conference on Computer Vision and Pattern Recognition* (2010).

[17] A. Kawewong and O. Hasegawa: "Fast and incremental attribute transferring and classifying system for detecting unseen object classes", in K. Diamantaras, W. Duch

and L. Iliadis Eds., *Proceedings of the 20th international conference on Artificial neural networks: Part III*, Vol. 6354 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 563–568 (2010).

[18] R. Fergus, P. Perona and A. Zisserman: "Object class recognition by unsupervised scale-invariant learning", *Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on*, Vol. 2, pp. II–264–II–271 vol.2 (2003).

[19] H. Schneiderman and T. Kanade: "A statistical method for 3d object detection applied to faces and cars", *Computer Vision and Pattern Recognition, 2000. Proceedings. IEEE Conference on*, Vol. 1, pp. 746–751 vol.1 (2000).

[20] P. Viola and M. Jones: "Rapid object detection using a boosted cascade of simple features", *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, Vol. 1, pp. I–511–I–518 vol.1 (2001).

[21] M. Weber, M. Welling and P. Perona: "Unsupervised learning of models for recognition", *Computer Vision - ECCV 2000*, Vol. 1842 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 18–32 (2000).

[22] I. Biederman: "Recognition-by-components: A theory of human image understanding", *Psychological Review*, **94**, pp. 115–147 (1987).

[23] C. Cortes and V. Vapnik: "Support-vector networks", *Mach. Learn.*, **20**, 3, pp. 273–297 (1995).

[24] F. Shen and O. Hasegawa: "A fast nearest neighbor classifier based on self-organizing incremental neural network", *Neural Networks*, **21**, 10, pp. 1537 – 1547 (2008).

[25] L. Bottou and O. Bousquet: "The tradeoffs of large scale learning", *IN: ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS 20*, pp. 161–168 (2008).

[26] T. Kohonen: "Self-organized formation of topologically correct feature maps", *Biological cybernetics*, **43**, 1, pp. 59–69 (1982).

[27] T. Kohonen: "Neurocomputing: Foundations of research", MIT Press, Cambridge, MA, USA, chapter Self-organized Formation of Topologically Correct Feature Maps, pp. 509–521 (1988).

[28] B. Fritzke: "A growing neural gas network learns topologies", *Advances in Neural Information Processing Systems 7*, MIT Press, pp. 625–632 (1995).

[29] S. Furao and O. Hasegawa: "An incremental network for on-line unsupervised classification and topology learning", *Neural Networks*, **19**, 1, pp. 90 – 106 (2006).

[30] S. Furao, T. Ogura and O. Hasegawa: "An enhanced self-organizing incremental neural network for online unsupervised learning", *Neural Networks*, **20**, 8, pp. 893 – 903 (2007).

[31] F. Shen and O. Hasegawa: "Self-organizing incremental neural network and its application", **6354**, pp. 535–540 (2010).

[32] F. Shen, H. Yu, K. Sakurai and O. Hasegawa: "An incremental online semi-supervised active learning algorithm based on self-organizing incremental neural network", *Neural Computing and Applications*, **20**, 7, pp. 1061–1074 (2011).

[33] A. Sudo, A. Sato and O. Hasegawa: "Associative memory for online learning in noisy environments using self-organizing incremental neural network", *Neural Networks, IEEE Transactions on*, **20**, 6, pp. 964–972 (2009).

[34] X. He, R. Kojima and O. Hasegawa: "Developmental word grounding through a growing neural network with a humanoid robot", *IEEE Trans. Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, **37**, 2, pp. 451–462 (2007).

[35] X. He, T. Ogura, A. Satou and O. Hasegawa: "Developmental word acquisition and grammar learning by humanoid robots through a self-organizing incremental neural

network", *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, **37**, 5, pp. 1357–1372 (2007).

[36] S. Okada and T. Nishida: "Incremental clustering of gesture patterns based on a self organizing incremental neural network", *Neural Networks, 2009. IJCNN 2009. International Joint Conference on*IEEE, pp. 2316–2322 (2009).

[37] S. Furao, A. Sudo and O. Hasegawa: "An online incremental learning pattern-based reasoning system", *Neural Networks*, **23**, 1, pp. 135 – 143 (2010).

[38] N. Makibuchi, F. Shen and O. Hasegawa: "Online knowledge acquisition and general problem solving in a real world by humanoid robots", in K. Diamantaras, W. Duch and L. Iliadis Eds., *Artificial Neural Networks ICANN 2010*, Vol. 6354 of *Lecture Notes in Computer Science*, Springer Berlin Heidelberg, pp. 551–556 (2010).

[39] A. Kawewong, R. Pimup and O. Hasegawa: "Incremental learning framework for indoor scene recognition", *Twenty-Seventh AAAI Conference on Artificial Intelligence* (2013).

[40] A. K. Jain, R. P. W. Duin and J. Mao: "Statistical pattern recognition: A review", *IEEE Trans. Pattern Anal. Mach. Intell.*, **22**, 1, pp. 4–37 (2000).

[41] D. Lowe: "Object recognition from local scale-invariant features", *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, Vol. 2, pp. 1150–1157 vol.2 (1999).

[42] H. Bay, A. Ess, T. Tuytelaars and L. V. Gool: "Speeded-up robust features (surf)", *Computer Vision and Image Understanding*, **110**, 3, pp. 346 – 359 (2008). Similarity Matching in Computer Vision and Multimedia.

[43] A. Bosch, A. Zisserman and X. Munoz: "Representing shape with a spatial pyramid kernel", *Proceedings of the 6th ACM International Conference on Image and Video Retrieval*, CIVR '07, New York, NY, USA, ACM, pp. 401–408 (2007).

[44] K. van de Sande, T. Gevers and C. Snoek: "Evaluating color descriptors for object and scene recognition", *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, **32**, 9, pp. 1582–1596 (2010).

[45] E. Shechtman and M. Irani: "Matching local self-similarities across images and videos", *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, pp. 1–8 (2007).

[46] G. Csurka, C. Dance, L. Fan, J. Willamowski and C. Bray: "Visual categorization with bags of keypoints", *Workshop on statistical learning in computer vision, ECCV*,

Vol. 1Prague, pp. 1–2 (2004).

[47] D. Kimura and O. Hasegawa: "Ultra-fast attribute-based transfer learning using images on the internet", *IEEE Conference on Computer Vision and Pattern Recognition*, Columbus, OH (2014).

[48] D. Kimura, K. Pichai and O. Hasegawa: "Unknown object recognition by images on the internet using ultra-fast and online incremental transfer learning", *Meeting on Image Recognition and Understanding*, Fukuoka, Japan (2012). [Japanese].

[49] D. Kimura, K. Pichai and O. Hasegawa: "Ultra-fast and online incremental transfer learning with the internet", *The 18th Symposium on Sensing via Image Information*, Kanagawa, Japan (2012). [Japanese].

[50] A. Haasch, N. Hofemann, J. Fritsch and G. Sagerer: "A multi-modal object attention system for a mobile robot", *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, Edmonton, Alberta, Canada, IEEE, pp. 1499–1504 (2005).

[51] T. Nakamura, T. Nagai and N. Iwahashi: "Multimodal object categorization by a robot", *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2415–2420 (2007).

[52] T. Araki, T. Nakamura, T. Nagai, K. Funakoshi, M. Nakano and N. Iwahashi: "Autonomous acquisition of multimodal information for online object concept formation by robots", *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 1540 –1547 (2011).

[53] T. Araki, T. Nakamura, T. Nagai, K. Funakoshi, M. Nakano and N. Iwahashi: "Online object categorization using multimodal information autonomously acquired by a mobile robot.", *Advanced Robotics*, **26**, 17, pp. 1995–2020 (2012).

[54] Y. Ozasa, N. Iwahashi, T. Takiguchi, Y. Ariki and M. Nakano: "Integrated multimodal information for detection of unknown objects and unknown names", *2012 RISP International Workshop on Nonlinear Circuits, Communications and Signal Processing* (2012).

[55] N. Dalal and B. Triggs: "Histograms of oriented gradients for human detection", *IEEE Conference on Computer Vision and Pattern Recognition*, Vol. 1, pp. 886 –893 vol. 1 (2005).

[56] J. W. Cooley and J. W. Tukey: "An algorithm for the machine calculation of complex fourier series", *Mathematics of computation*, **19**, 90, pp. 297–301 (1965).

[57] R. Kohavi, et al.: "A study of cross-validation and bootstrap for accuracy estimation and model selection", *International Joint Conferences on Artificial Intelligence*, Vol. 14, pp. 1137–1145 (1995).

[58] D. Nguyen-Tuong and J. Peters: "Model learning for robot control: a survey", *Cognitive Processing*, **12**, 4, pp. 319–340 (2011).

[59] B. D. Argall, S. Chernova, M. Veloso and B. Browning: "A survey of robot learning from demonstration", *Robotics and Autonomous Systems*, **57**, 5, pp. 469 – 483 (2009).

[60] J. Koenemann, F. Burget and M. Bennewitz: "Real-time imitation of human whole-body motions by humanoids", *Robotics and Automation (ICRA), 2014 IEEE International Conference on*IEEE, pp. 2806–2812 (2014).

[61] N. Naksuk, C. G. Lee and S. Rietdyk: "Whole-body human-to-humanoid motion transfer", *Humanoid Robots, 2005 5th IEEE-RAS International Conference on*IEEE, pp. 104–109 (2005).

[62] H. Asada and S. Liu: "Transfer of human skills to neural net robot controllers", *Robotics and Automation, 1991. Proceedings., 1991 IEEE International Conference on*, pp. 2442–2448 vol.3 (1991).

[63] B. Bocsi, L. Csató and J. Peters: "Alignment-based transfer learning for robot models", *Neural Networks (IJCNN), The 2013 International Joint Conference on*IEEE, pp. 1–7 (2013).

[64] D. J. Willshaw and C. V. D. Malsburg: "How patterned neural connections can be set up by self-organization", *Proceedings of the Royal Society of London B: Biological Sciences*, **194**, 1117, pp. 431–445 (1976).

[65] R. P. Paul: "Robot Manipulators: Mathematics, Programming, and Control", MIT Press, Cambridge, MA, USA, 1st edition (1982).

[66] S. Kucuk and Z. Bingul: "Robot kinematics: forward and inverse kinematics", INTECH Open Access Publisher (2006).

[67] D. Shepard: "A two-dimensional interpolation function for irregularly-spaced data", *Proceedings of the 1968 23rd ACM National Conference*, ACM '68, New York, NY, USA, ACM, pp. 517–524 (1968).

PUBLICATIONS

学術論文誌

[J1] 木村大毅, Kankuekul Pichai, 長谷川修, “属性知識の高速オンライン学習と転移による未知クラス推定”, 知能と情報, V26, No.5, pp.830-843, 2014.

国際会議

[IC1] Daiki KIMURA, R. NISHIMURA, A. OGURO, and O. HASEGAWA, “Ultra-fast Multimodal and Online Transfer learning on Humanoid Robots”, Proceedings on ACM/IEEE International Conference on Human-Robot Interaction (HRI), Tokyo, Japan, March 2013.

[IC2] O. HASEGAWA, and Daiki KIMURA (presenter), “Home Robots, Learn by Themselves”, Proceedings on International Conference on Human-Computer Interaction (HCI), Las Vegas, Nevada, the USA, July 2013.

[IC3] Daiki KIMURA, and O. HASEGAWA, “An Artificial Brain for Intelligent Robots”, Proceedings on International Workshop on Advanced Computational Intelligence and Intelligent Informatics, Shanghai, China, October 2013

[IC4] Daiki KIMURA, and O. HASEGAWA, “Ultra-Fast Attribute-Based Transfer Learning Using Images on the Internet”, IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), Ohio, the USA, June 2014.

[IC5] Daiki KIMURA, and O. HASEGAWA, “Object Recognition for Pedagogical Robots”, Proceedings on International Symposium on Pedagogical Machines, Tokyo, Japan, March 2015.

[IC6] Daiki KIMURA, and O. HASEGAWA, “Estimating Multimodal Attributes for Unknown Objects”, Proceedings on International Joint Conference on Neural Networks (IJCNN), Kilarney, Ireland, July 2015.

国内会議

[DC1] 木村大毅, Kankuekul Pichai, Kawewong Aram, 長谷川修, “ 超高速オンライン転移学習 ”, 画像センシングシンポジウム, 神奈川県, June 2011. 【優秀学術賞】

PUBLICATIONS

[DC2] 木村大毅, Kankuekul Pichai, 長谷川修, ”インターネットを活用した超高速オンライン転移学習”, 画像センシングシンポジウム, 神奈川県, June 2012.

[DC3] 木村大毅, Kankuekul Pichai, 長谷川修,“ネット画像を用いた超高速オンライン転移学習による未知物体推定”, 画像の認識･理解シンポジウム, 福岡県, August 2012. 【ベストデモ賞】

[DC4] 木村大毅, 西村竜太朗, 小黒陽弘, 長谷川修,“ネットから学習し未知物体を推定する超高速マルチモーダル・オンライン転移学習”, ロボティクスシンポジア, 山形県, March 2013. 【若手奨励賞】

[DC5] 木村大毅, 長谷川修,“超高速マルチモーダル属性転移による実世界状況把握”, 画像センシングシンポジウム, 神奈川県, June 2013.

[DC6] 木村大毅, 長谷川修,“マルチモーダル情報を用いた実世界物体認識”, 画像の認識・理解シンポジウム, 東京都, July 2013.

[DC7] 木村大毅, 水野俊一郎, 長谷川修,“対話的能動学習による超高速ビッグデータ解析”, 画像センシングシンポジウム, 神奈川県, June 2014.

[DC8] 木村大毅, 水野俊一郎, 長谷川修,“能動学習を用いた効率的なビッグデータ解析”, 画像の認識・理解シンポジウム, 岡山県, July 2014.

PUBLICATIONS

[DC9] P. KANKUEKUL, 木村大毅, K. ARAM, T. SIRINART, 長谷川修, “ オンライン物体認識に関する一考察 ”, 情報処理学会 全国大会, 東京都, March 2011.

特許

[P1] 長谷川修, 木村大毅, カーンクアクーン ピシャイ, カーウィーウォン アラム, “ 属性の学習及び転移システム，認識器生成装置，認識器生成方法及び認識装置 ”, 2012.

[P2] 長谷川修, 木村大毅, “動作の転移装置，動作の転移方法及びプログラム”, 2014.

ACKNOWLEDGMENTS

本研究において，終始変わらぬ御指導を頂きました長谷川修准教授に心から感謝すると共に，厚く御礼申し上げます．また東京工業大学 大学院総合理工学研究科 知能システム科学専攻の新田克己教授，渡邊澄夫教授，小野功准教授，石井秀明准教授には，本研究に関して多くの有益なコメントを頂きました．ここに御礼申し上げます．

また日頃からお世話になりました長谷川研究室の皆様に感謝いたします．特に，本研究に対して多くの助言を下さった卒業生の Kawewong Aram さん，巻渕有哉さん，Kankuekul Pichai さん，西村竜太朗さん，小黒陽弘さん，Sermsawatsri Panjamapong さんに感謝致します．

なお，本研究の実施にあたり国立研究開発法人 新エネルギー・産業技術総合開発機構 産業技術研究助成事業，及び科学技術振興機構 戦略的創造研究推進事業 CREST，日本学術振興会 科学研究費助成事業 特別研究員奨励費から御支援を頂きました．記して感謝致します．

最後に，温かく見守ってくれた両親や友人，そして，これまでの学生生活で関わったすべての人たちに感謝致します．